Development of Intelligent Pattern Recognition Algorithms for Assessment of Quality of Edible Oils

THESIS

Submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

by

SRINATH K
(2013PHXF0105P)

Under the Supervision of

PROF. SUREKHA BHANOT

And Under the Co-Supervision of

Dr. P C PANCHARIYA

BITS Pilani
Pilani | Dubai | Goa | Hyderabad

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI

2021

Dedication

To my parents and my brother

&

My wife for their unconditional support

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI – 333031 (RAJASTHAN), INDIA

CERTIFICATE

This is to certify that the thesis entitled **"Development of Intelligent Pattern Recognition Algorithms for Assessment of Quality of Edible Oils"** submitted by **Srinath K,** ID No. **2013PHXF0105P** for the award of Ph.D. of the institute embodies original work done by him under our supervision.

Signature of the Supervisor Name: Prof Surekha Bhanot Designation: Professor Date:	Signature of Co-Supervisor Name: Dr. P C Panchariya Designation: Director Institute: CSIR-CEERI, Pilani Date:

Acknowledgments

First, I would like to thank Almighty for the support of whom I have reached this stage of my Ph.D. I would like to thank the following people for their support and affection during the entire duration of my Ph.D., professionally or personally, without whom I would not have been able to complete my research. I may have forgotten to acknowledge all the people, but such mistakes may be forgotten.

I would like to take this opportunity to convey my gratitude and appreciation to my supervisors for all their support and guidance during my complete research tenure. At the outset, I wish to express my deepest gratitude to Prof. Surekha Bhanot (Professor), Department of Electrical and Electronics Engineering, BITS-Pilani, and Dr. P C Panchariya, Director, CSIR-CEERI, Pilani, for their guidance, suggestions, moral support throughout the research period.

I express my appreciation to BITS-Pilani for providing all the necessary facilities to conduct the research. My special thanks to Prof. R. N. Saha, Acting Vice-Chancellor of the university, Prof. S. K. Barai, Director, BITS-Pilani, Pilani campus, allow me to pursue my research successfully. I would like to thank Prof. M. B. Srinivas, Dean, and Prof. Jitendra Panwar, Associate Dean, Academic–Graduate Studies & Research Division, for valuable support throughout the program. I would like to express my gratitude to Prof. K. K. Gupta, Associate Professor, Department of Electrical and Electronics Engineering and Prof. R. K. Gupta, Professor, Department of Physics, Members of Doctoral Advisory Committee (DAC), for motivation and moral support technically as well as personally. I would like to thank Prof. Hitesh Dutt Mathur, Head of Department, Electrical and Electronics Engineering, for their motivation during the entire period of Ph.D.

I would like to thank my friend Mr. Punit Khatri for his support throughout the research work, and without him, I wouldn’t have completed this work. I would like to thank all my friends Dr. Harsha Raj Koundinya, Mr. Kailash Nayak, Mr. Surendra Singh Patel, Mr. Amit Singh, Mr. Krishna Veer Singh, and Mr. Rahul Sharma for their support. I also thank Dr. A. H. Kiranmayee, Mr. Ravindra Singh Chauhan, Mr. Navjot Kumar, Mr. Rishi Ranjan and all other whose name is not here and have helped me

during my Ph.D. Thanks to all faculty members of BITS-Pilani, Pilani Campus, for helping me out at various times.

Finally, I would like to thank my father-Late Shri. Kandala Laxmi Narsimha Chary and my mother Smt. Kandala Manjula for their sacrifice to make me for what I am today. I also thank my brother-Mr. Kandala Ramakanth and my wife-Ms. Kandala Archana, for their care, affection, and support during my research journey.

Srinath. K

Abstract

Edible oils are important components of the human diet because they provide nutritional value and trans fatty acids. To address concerns of consumers and manufacturers regarding the quality of edible oils, several quality standards have been proposed to maintain their quality. Unfortunately, some traders and manufacturing units fail to meet standards for obtaining profits at the expense of compromising the quality of edible oils by, adding low-quality ingredients, or removing several vital components which results in adulterated edible oils. The focus of this research work has been on the analysis of adulteration in edible oils due to their importance in a daily diet and health effects in case of adulteration. Human senses have been used for assessing the quality of food items, but the assessment processes are not so reliable as they are subjective in nature, depends on individual perception, and complex adulterations cannot be identified with human senses. Alternatively, using analytical instruments is an excellent solution to problems associated with human senses, such as individual variability, impossibility of online monitoring, subjectivity, adaptation, infections, harmful exposure to hazardous components. These analytical instruments are biomimetic systems inspired from the human senses like taste, smell, and vision.

An Electronic tongue is a biomimetic inspiration of human gustation (perception of taste). An array of non-specific/semi-specific sensors linked to a pattern recognition system can determine the fingerprint of a liquid sample using the electronic tongue. The human olfaction system has been the inspiration for the development of an electronic system for distinguishing different odors called an electronic nose (e-nose). An electronic nose comprises an array of non-specific/semi-specific gas sensors coupled to a pattern recognition system. Spectroscopic methods are an inspiration from human vision that use a light source and detector to study the absorption and transmission parameters which are the characteristic to the sample under test combined with pattern recognition algorithms.

Traditional chemical methods based on American Oil and Chemist Society (AOCS) have been used for the measurement of physicochemical parameters of edible oils and thereby the authenticity of edible oil confirmed. The research work in this thesis deals with the investigation of electrochemical and spectroscopy methods coupled with chemometric and artificial pattern recognition algorithms for the discrimination of edible oils. Based on the investigation results a simple yet accurate method (MIR spectroscopy based on ATR sampling) has been identified and used further in detection

of adulteration in edible oils. Lab designed adulterated samples have been used in the experimentation. Proposed analytical methods produce a huge amount of data (involving thousands of variables at times), hence some data reduction techniques to reduce the data while keeping the necessary pertinent information intact have been used. The thesis aims to develop Artificial Intelligent (AI) based chemometric models for the discrimination of edible oils and detection of adulterations in edible oils. Statistical and soft computing algorithms for classification and adulteration detection have been developed. Regression models based on partial least squares and artificial neural networks have been developed. These models are simpler and more accurate at the same time, making them easily implementable in embedded systems. The models were designed to be deployed in stand-alone, online, and cost-effective systems, so that data analysis and inferences can be made instantly and accurately.

Performances of classification models have been analyzed with metrics like sensitivity, specificity, precision, accuracy, and F_1-score. The regression model's performance has been assessed with root mean square error and coefficient of regression.

The results obtained using the Linear Support Vector Machines (L-SVM) and Convolutional Neural Networks (1D CNN and 2D CNN) have been highly accurate in discrimination of edible oils and detection of adulterations in edible oils. Furthermore, the Artificial Neural Networks (ANN) based regression results are accurate in predicting the percentage of adulteration. The inference models have been implemented on ARM-based embedded processor, and the results are compared with the algorithms implemented in the standard MATLAB and Python. In short, the work presented in this thesis can be useful in developing portable and reliable intelligent instrumentation systems for assessment of the quality of edible oils.

Table of Contents

List of Figures

List of Tables

Abbreviations

AI	Artificial Intelligence
ANN	Artificial neural networks
AOCS	American Oils and Chemist Society
ATR	Attenuated Total Reflection
CNN	Convolutional Neural Networks
CSV	Comma Separated Values
CSE	Conventional Solvent Extraction
CV	Cyclic Voltammetry
DA	Discriminative Analysis
EIS	Electrochemical Impedance Spectroscopy
FSSAI	Food Safety Standards Authority of India
GC	Gas Chromatography
HCA	Hierarchal Clustering Analysis
HCL	Hydrochloric Acid
HPLC	High Pressure Liquid Chromatography
LDA	Linear Discriminant Analysis
ME	Mechanical Extraction
MIR	Mid infrared
ML	Machine Learning
MUFA	Monounsaturated Fatty Acids
NIR	Near infrared
PCA	Principal Component Analysis
PCA-DA	Principal Component Analysis-Discriminant Analysis
PLS-DA	Partial Least Square-Discriminant Analysis
PLSR	Partial Least Square Regression
PUFA	Ploy Unsaturated Fatty Acids
RMSE	Root Mean Squared Error
RMSEV	Root Mean Squared Error for Validation
SVM	Support Vector Machine
SIMCA	Soft Independent Modeling of Class Analogy
TAG	Triacylglycerol
TFA	Trans Fatty Acids
WHO	World Health Organization

Chapter 1

Introduction

Food is one of the basic needs of every living being. It is composed of essential nutrients such as carbohydrates, water, fats, proteins, vitamins, and minerals, etc. Food can be taken in liquid or solid form by animals, humans for nutrition or pleasure. It is usually of plant, animal origin, ingested by an organism and absorbed by the organism's cells to give energy, maintain life, and support growth. One must consume safe and nutritious food to remain healthy. Milk, rice, edible oils, fruit juices are essential dietary components in the daily life of human beings. Every food and its components will have characteristic attributes such as aroma, texture, taste, and color by which food quality is assessed. Among the food quality attributes, color is an appearance factor for an indication of ripeness or spoilage and the endpoint of cooking processes. The texture of food refers to the sensation a person feels with their fingers, tongue, the teeth. Flavor refers to the taste sensations perceived by the tongue — sweet, salty, sour, umami, and bitter — and, to a lesser extent, smells as perceived by the nose. Human senses have always been used to assess food quality. The senses of sight, taste, smell, and touch are utilized daily in all aspects of human life. These sensory systems of Homo sapiens are the results of millions of years of evaluations which is advantageous to our survival and growth [1]. Attributes of edible items such as taste, aroma, and color can be studied analytically using sensory methods and integrated to provide the overall sensory evaluation of food [2], [3].

The quality of food is the primary concern to the manufacturing industries and consumers, the "quality" encompassing many different meanings, including nutritional value, safety, the composition of its constituent and their physical, chemical properties, the proportions of microbiological and toxic contaminants, processing methods, packaging, and storage. The quality aspect of food is an essential need for the consumer.

Initially, the human sensory assessment was the primary method for estimating the quality of any food or food ingredient. Taste panels comprised of professionals or groups of experts assessing the standard of edible foods are normal practice. This evaluation process will not be always reliable and consistent, because it is an entirely subjective assessment and depends on human perception. The results are dependent

on the taster's state of mind, and the assessment results are not transferable to numerical values. The presence of invisible and unidentifiable food components may not be detected with human senses. There is a legitimate need for finding a method to authenticate the food quality in a reliable and precise manner. In recent years, people are progressively paying attention to food quality and safety, which are significantly relating to health. Food adulteration is the intentional degrading of food quality through the admixture or substitution of inferior substances or the removal of a valuable ingredient. Such adulteration of food is a major concern to the consumers, and it is a worldwide issue relating to food quality and safety.

The World Health Organization (WHO) defines safe food as "food that may not harm the consumer when prepared or consumed as intended". Pollutants in the environment, as well as the indiscriminate use of chemical fertilizers and pesticides, are sources of food contamination. Aside from environmental pollutants, food is contaminated by the intentional or unintentional addition of toxicants. According to the regulations of the Food Adulteration Act of 1956, the following are considered as adulteration:

- The item sold by a vendor is not of nature claimed, for example, using hydrogenated oils instead of ghee to make sweets.
- Any inferior or cheap substance has been substituted wholly or in a part for the item; for example, starch powder has been mixed in milk powder or milk.
- It contains any prohibited preservatives or an excessive amount of permitted ones.
- It does not satisfy the prescribed standards laid down by the authorities, which makes the item injurious to consumer health.

According to the Food Safety and Standards Authority of India (FSSAI), adulteration is the process of mixing inferior and sometimes harmful components or removing some rich nutritional substances which make the material unfit for consumption [4]. FSSAI has been constituted for laying down science-based standards for food products and to regulate rules regarding their manufacture, storage, distribution, sale, and import, thus ensuring the availability of safe and wholesome food for human consumption. Unfortunately, some traders and manufacturing agencies fail to comply with standards for their profits. They do it at the cost of compromising the quality of food, with the addition of low-quality ingredients, or by removal of several vital components. This is the result of adulterated or counterfeit products in the food industry. It is also a fact that as demand for food increases,

adulteration in the food substances also increases. Gaps between production, consumption, and the inability of regulatory authorities to check adulteration are reasons/opportunities for fraudsters to make easy money through adulteration at the cost of nutrition and consumer's health [5]. Such adulteration in food not only reduces its nutritional quality but may affect its safety also, which may, in turn, affect the health of the consumers adversely. Food adulteration is a big threat to food quality and food safety worldwide.

The problem of food adulteration is likely to be as old as the food processing and production systems themselves [6]. The first scientific attempt to expose food adulteration was made by German analytical chemist Frederick Accum who completed a treatise on the adulteration of food and culinary poisons published in 1820 [7]. Later it was carried out by several scientists to expose such adulterations. According to the World health organization (WHO), the China milk crisis in May 2007 is the largest food adulteration fraud in the world which involved melamine adulteration in milk and pet food, causing 30000 infants to become its victims in China. Spanish toxic oil syndrome in 1981, epidemic dropsy in India during 1989 after consuming adulterated mustard oil with argemone oil are some examples of severe consequences of adulteration [8].

Generally, the food products that are more prone to get adulterated include olive oil, fish, honey, milk and dairy products, edible oils, meat products, grain-based foods, fruit juices, wine and alcoholic beverages, organic foods, spices, coffee, and tea, and some highly processed foods. Adulteration in the above-mentioned food components may be in the form of addition, substitution, dilution, counterfeiting, unauthorized enhancements, mislabeling [9], [10]. For example, the adulteration of cheap vegetable oils in virgin olive oil, palm oil addition in groundnut oil [11]–[13], adulteration of low-quality rice in a superior type of rice [14], the addition of water, skim milk powder, cane sugar (sucrose), starch, fat, ammonium sulfate, melamine [15], etc. In addition to preservatives such as hydrogen peroxide, benzoic acid, salicylic acid, carbonates, bicarbonates, formalin, caustic soda, and antibiotics are used to increase shelf life. Additives such as urea, vanaspati are also used to make milk look natural. Mung beans are added to pistachios [16]. Food adulteration, especially to cheat consumers, is a constant struggle between the "science of deception" and the "science of detection" [15]. Many adulterants might prove to be harmful to health, especially when consumed over a long period like milk adulteration causes severe abdominal diseases, edible oil adulteration causes cardiac diseases [17].

As previously stated, the primary approach for evaluating food quality is human sensory evaluation, in which the assessment is based on individual perception of food attributes and is incapable of accurately and definitively detecting adulterations using human senses. Analytical instrumentation techniques combined with signal or data processing algorithms are an alternative to humans in the study of adulteration. For this research work, we focused on adulteration in edible oils, because of their importance in everyday diets and the health consequences of adulteration.

1.1. Background

Edible oils extracted from plant seeds are a vital component of a routine diet for human consumption. Oils are chemically composed of major triacylglycerols (TAGs) and minor (sterols, carotenoids, and tocopherols) components [18]. These edible oils offer nutrients that are essential to human health as they are the primary source of mono and poly-unsaturated fats. Besides health benefits, these oils also have economic benefits. For example, Extra Virgin Olive Oil (EVO) is more expensive as it offers a delicious flavor along with high content of vitamins and antioxidants. Moreover, it plays a big role in the economy of some countries like Tunisia, where olive farms covering around 1.7 million hectares of the area contribute about 4% of the olive oil produced in the world [19], [20]. Sesame oil (SES) is extracted from a plant called Sesamum Indicum L, which is one of the highly valuable edible oils due to its special characteristic flavors that makes it a food oil in many Asian countries, especially India. It is used as a flavor enhancer. Studies have been conducted to investigate the health-benefit effects of sesame and its effectiveness against various diseases, including atherosclerosis and hypertension, and also its role as an anti-aging effect [21], [22]. Similarly, there are multiple varieties of most frequently used edible oils like groundnut oil, canola oil, soybean oil, rice bran oil, palm oil, sunflower oil, and coconut oil available in India. Each of them has flavor enhancement, as well as health and economic benefits.

Generally, the cost of edible oil depends on the extraction process from seeds and their nutritional values. There are two major types of extraction methods, namely the cold-pressed method and solvent extraction method. Oils extracted from plant seeds by the cold-pressed methods are proved to be healthier as they can retain antioxidants and nutritional compounds [23]. Because of this reason, cold-pressed oils are expensive than other refined oils. The solvent extraction method uses various chemicals like hexane.

From a survey report, the total production of edible oil in India was 25.3 million tonnes in 2015-16, and the total area of cultivation under edible oils was 26.13 million hectares. The maximum production of edible oils reported in 2013-14 was 32.75 million tonnes from an area of 28.05 million hectares. India imported 148.2 lakh tonnes of edible oils in 2015-16, and net domestic availability was 86.37 lakh tonnes (ICAR, 2016) [24]. Similarly, the world production of soybean oil, rapeseed oil, palm oil, sunflower oil, coconut oil, and olive oil was 56.51, 27.3, 72.27, 21.2, 3.62, and 3.12 million metric tons respectively in 2019–2020 [25]. Greater demand in the national and international market for edible oils creates the problem of adulteration, in which the cheap oils are mixed with cold-pressed edible oils. In some cases, adulteration of argemone mexicana seed oil in mustard oils has been reported as cause of epidemic dropsy.

Edible oils are made of approximately 98% triacylglycerols with variations in length, substitution order, and degree of saturation of fatty acid chains. The remaining 2% consists of sugars, sterols, carotenes, phospholipids, and lipid soluble vitamins [2]. Soybean, olive, sunflower, palm, coconut, and rapeseed oils are the most frequently used edible oils these days. Physical properties of edible oils include density, specific gravity, viscosity, and surface tension. Chemical properties of edible oil include peroxide value, refractive index, melting point, freezing point, free fatty acids, unsaturated and mono-saturated fats, etc. All these properties are characteristic of edible oil, and the purity of oils is ascertained using these properties. Most of the adulteration detection methods are chemical based, so there is a need for a robust, yet simple analytical instrumental method combined with signal processing algorithms.

The present research work focuses on three important issues.

i Discrimination of different types of edible oils based on the data acquired from instrumental methods.
ii Detection and quantification of adulterations in edible oils by developing Artificial Intelligence(AI) based algorithms.
iii Realization of developed discrimination and adulteration detection algorithms on embedded systems.

Adulteration in edible oils is a major concern for the government as well as for society. Investigation of analytical instruments, development of AI based models for edible oil analysis, and their realization on the embedded platform may lead to

making real-time portable handheld prototypes for detection and calibration of adulteration in edible oils.

1.2. Research motivation

Motivation to work on detection of adulteration in edible oils has come from a variety of sources, including media, news, and market studies, such as a consumer study in 2016 that revealed the percentage of adulteration in edible oils. The report states that the maximum percentage of adulteration is in edible oils [26]. Taking the high failure rate, i.e., the high percentage of adulteration in edible oils, motivated us to work in this area.

Table 1.1 Report on edible oil adulteration in India

Edible oil	**Total samples tested**	**Failed samples**	**Failed samples in %**
Coconut oil	149	126	84.56
Cottonseed oil	54	40	74.07
Sesame oil	50	37	71.77
Mustard oil	124	89	71.77
Groundnut oil	150	90	59.21
Palmolive oil	50	16	32
Soyabean oil	230	46	20
Sunflower Oil	206	34	16.5

Source: Consumer Voice 2016.

1.3. Objectives of proposed research

After doing background study and extensive literature review following objectives are laid down for research work.

1. Investigation of different instrumental methods like Electrochemical (Voltammetry, Electrochemical Impedance Spectroscopy (EIS)) and Spectroscopy methods (Near Infrared (NIR) and Mid Infrared with Attenuated Total Reflection (MIR-ATR) sampling for rapid analysis of edible oils.
2. Development of Artificial Intelligence (AI) based chemometric classification models for identification of edible oils.
3. Development of AI based chemometric models for detection and quantification of adulteration of edible oils.
4. Implementation of developed AI based inference models on an embedded platform.

1.4. Thesis organization

The organization of the thesis is as follows:

Chapter 1 discusses background of the research topic, properties of edible oils, motivation of research work, objectives of the research work to be carried out in thesis.

Chapter 2 presents the extensive literature survey covering the perspective of existing, state-of-art experimental methodologies and data processing algorithms for detection, classification, and quantification of adulterations in edible oils.

Chapter 3 discusses the experimental methods for analyzing edible oils, such as electrochemical methods and spectroscopic approaches.

Chapter 4 describes the development of artificial pattern recognition algorithms like Convolution Neural Networks (1D-CNN,2D-CNN), Linear Support Vector Machine (SVM), Partial Least Square Discriminant Analysis (PLS-DA), Soft Independent Modeling of Class Analogy (SIMCA), and Principal Component Analysis Discriminant Analysis (PCA-DA) for the classification of edible oils.

Chapter 5 describes the regression analysis based on a Successive Projection Algorithm (SPA) coupled with Multiple Linear Regression (MLR), Partial Least Square Regression (PLSR), PCR, and ANN-based prediction models for quantification of adulteration in edible oils.

Chapter 6 explains the AI-based pattern recognition algorithm implementation on an ARM-based embedded platform to make a portable system for the detection of adulteration.

Chapter 7 discusses the results, conclusion, and future scope of work in this area.

Bibliography

[1] H. Smyth and D. Cozzolino, "Instrumental methods (Spectroscopy, Electronic Nose, and Tongue) as tools to predict taste and aroma in beverages: Advantages and limitations," *Chem. Rev.*, vol. 113, no. 3, pp. 1429–1440, 2013, doi: 10.1021/cr300076c.

[2] M. W. Heaven and D. Nash, "Recent analyses using solid phase microextraction in industries related to food made into or from liquids," *Food Control*, vol. 27, no. 1. Elsevier, pp. 214–227, Sep. 01, 2012, doi: 10.1016/j.foodcont.2012.03.018.

[3] Y. Huang, Y. Lan, and R. E. Lacey, "Artificial senses for characterization of food quality," *J. Bionic Eng.*, vol. 1, no. 4, pp. 159–173, Dec. 2004, doi: 10.1007/bf03399472.

[4] http://www.old.fssai.gov.in/Portals/0/Pdf/pfa acts and rules.pdf (accessed Feb. 10, 2021).

[5] D. P. Attrey, "Detection of food adulterants/contaminants," in *Food Safety in the 21st Century: Public Health Perspective*, Elsevier Inc., 2017, pp. 129–143.

[6] P. Shears, "Food fraud - a current issue but an old problem," *Br. Food J.*, vol. 112, no. 2, pp. 198–213, 2010, doi: 10.1108/00070701011018879.

[7] "A Treatise on Adulteration of Food and Culinary Poisons (1820) – The Public Domain Review." https://publicdomainreview.org/collection/a-treatise-on-adulteration-of-food-and-culinary-poisons-1820 (accessed Feb. 12, 2021).

[8] R. Jamwal *et al.*, "Attenuated total Reflectance–Fourier transform infrared (ATR–FTIR) spectroscopy coupled with chemometrics for rapid detection of argemone oil adulteration in mustard oil," *LWT*, vol. 120, p. 108945, Feb. 2020, doi: 10.1016/j.lwt.2019.108945.

[9] S. Sumar and H. Ismail, "Adulteration of foods -past and present," *Nutr. Food Sci.*, vol. 95, no. 4, pp. 11–15, Aug. 1995, doi: 10.1108/00346659510088663.

[10] G. P. Danezis, A. S. Tsagkaris, V. Brusic, and C. A. Georgiou, "Food authentication: state of the art and prospects," *Current Opinion in Food Science*, vol. 10. Elsevier Ltd, pp. 22–31, Aug. 01, 2016, doi: 10.1016/j.cofs.2016.07.003.

[11] P. F. Filoda *et al.*, "Fast Methodology for Identification of Olive Oil Adulterated with a Mix of Different Vegetable Oils," *Food Anal. Methods*, vol. 12, no. 1, pp. 293–304, Jan. 2019, doi: 10.1007/s12161-018-1360-5.

[12] J. H. Kalivas, C. A. Georgiou, M. Moira, I. Tsafaras, E. A. Petrakis, and G. A. Mousdis, "Food adulteration analysis without laboratory prepared or determined reference food adulterant values," *Food Chem.*, vol. 148, pp. 289–

293, Apr. 2014, doi: 10.1016/j.foodchem.2013.10.065.

[13] S. Mildner-Szkudlarz and H. H. Jeleń, "The potential of different techniques for volatile compounds analysis coupled with PCA for the detection of the adulteration of olive oil with hazelnut oil," *Food Chem.*, vol. 110, no. 3, pp. 751–761, Oct. 2008, doi: 10.1016/j.foodchem.2008.02.053.

[14] K. Timsorn, Y. Lorjaroenphon, and C. Wongchoosuk, "Identification of adulteration in uncooked Jasmine rice by a portable low-cost artificial olfactory system," *Meas. J. Int. Meas. Confed.*, vol. 108, pp. 67–76, Oct. 2017, doi: 10.1016/j.measurement.2017.05.035.

[15] B. Sezer, S. Durna, G. Bilge, A. Berkkan, A. Yetisemiyen, and I. H. Boyaci, "Identification of milk fraud using laser-induced breakdown spectroscopy (LIBS)," *Int. Dairy J.*, vol. 81, pp. 1–7, Jun. 2018, doi: 10.1016/j.idairyj.2017.12.005.

[16] B. Sezer, H. Apaydin, G. Bilge, and I. H. Boyaci, "Detection of Pistacia vera adulteration by using laser induced breakdown spectroscopy," *J. Sci. Food Agric.*, vol. 99, no. 5, pp. 2236–2242, Mar. 2019, doi: 10.1002/jsfa.9418.

[17] M. P. Iqbal, "Trans fatty acids - A risk factor for cardiovascular disease," *Pakistan Journal of Medical Sciences*, vol. 30, no. 1. Professional Medical Publications, pp. 194–197, 2014, doi: 10.12669/pjms.301.4525.

[18] S. Azadmard-Damirchi and M. Torbati, "Adulterations in Some Edible Oils and Fats and Their Detection Methods," 2015. Accessed: Feb. 10, 2021.

[19] H. Jabeur, A. Zribi, J. Makni, A. Rebai, R. Abdelhedi, and M. Bouaziz, "Detection of chemlali extra-virgin olive oil adulteration mixed with soybean oil, corn oil, and sunflower oil by using GC and HPLC," *J. Agric. Food Chem.*, vol. 62, no. 21, pp. 4893–4904, May 2014, doi: 10.1021/jf500571n.

[20] W. A. Salah and M. Nofal, "Review of some adulteration detection techniques of edible oils," *Journal of the Science of Food and Agriculture*, vol. 101, no. 3. John Wiley and Sons Ltd, pp. 811–819, Feb. 24, 2021, doi: 10.1002/jsfa.10750.

[21] M. H. Kang, Y. Kawai, M. Naito, and T. Osawa, "Dietary defatted sesame flour decreases susceptibility to oxidative stress in hypercholesterolemic rabbits," *J. Nutr.*, vol. 129, no. 10, pp. 1885–1890, 1999, doi: 10.1093/jn/129.10.1885.

[22] Y. Fukuda, T. Osawa, M. Namiki, and T. Ozaki, "Studies on antioxidative substances in sesame seed," *Agric. Biol. Chem.*, vol. 49, no. 2, pp. 301–306, 1985, doi: 10.1080/00021369.1985.10866739.

[23] a. Siger, m. Nogala-kalucka, and e. Lampart-szczapa, "the content and antioxidant activity of phenolic compounds in cold-pressed plant oils," *J. Food Lipids*, vol. 15, no. 2, pp. 137–149, May 2008, doi: 10.1111/j.1745-

4522.2007.00107.x.

[24] R. Jamwal *et al.*, "Recent trends in the use of FTIR spectroscopy integrated with chemometrics for the detection of edible oil adulteration," *Vibrational Spectroscopy*, vol. 113. Elsevier B.V., Mar. 01, 2021, doi: 10.1016/j.vibspec.2021.103222.

[25] " Statista - The Statistics Portal for Market Data, Market Research and Market Studies." https://www.statista.com.

[26] "Shubham Yadav. Edible oil adulterations: Current issues, detection techniques, and health hazards. Int J Chem Stud 2018;6(2):1393-1397.).

Chapter 2

Literature Review

2.1. Preamble

Edible oils are fats derived from organic plants or plant seeds such as groundnut, soybean, sunflower, rapeseed, mustard, castor, cottonseed, and a variety of tree fruits such as coconut, palm, and olive. They are plant-derived biological mixtures that contain ester mixtures and a chain of fatty acids [1]. Edible oils are an important component of human diets, not only for their sensory (taste and smell) qualities but also for providing the necessary nutritional sources. Concern about the quality of edible oils is increasing these days, as it impacts consumer health. As a result, determining the quality of edible oil is important.

The current chapter provides an in-depth review of the literature on edible oil constituents, adulteration causes and potential impacts, historical perspectives in edible oil adulteration detection, state-of-art analytical, experimental methods, and algorithms for classification of edible oils, detection and quantification of adulteration in edible oils. This chapter also includes a review of the literature on researchers efforts to implement AI and machine learning inference algorithms on embedded processors.

2.2. Edible oil

As the name suggests, “edible oil” is safe for human consumption, and it provides health, nutritional, and taste requirements. Oilseeds and various tree fruits are the primary sources of edible oils. Oilseeds include sunflower, rapeseed, mustard, castor, cottonseed, groundnut, and soybean, etc., while tree fruits include coconut, palm, and olive. Vegetable oil refers to oil derived from vegetable sources. However, not all vegetable oils are suitable for human consumption, and some are used in industrial applications. The commercial value of certain oils is better for industrial usages, like castor oil. Edible oils contain a variety of essential fatty acids, which can be classified into three categories: saturated, mono-unsaturated (MUFA), and polyunsaturated (PUFA). The unsaturated fatty acids are divided into omega series. The human body needs omega-3 and omega-6 fatty acids for improving heart health and balancing blood pressure. Human beings cannot synthesize them because they lack the

desaturase enzymes required for their production, so they must be obtained through edible oils in their diet [2]. The oil quality will depend, among others, on oilseed type, soil, and environmental conditions around the resource oil-bearing plant, pretreatment procedure, and the particular extraction method(s) used [3].

Extraction is the process of separating triglycerides from oilseeds. Mechanical extraction (ME) and conventional solvent extraction (CSE) are the two main types of extraction methods. Mechanical extraction is the oldest method of producing seed oil, in which the oilseeds are placed between two barriers, with pressure on the oil seeds forcing the oil out of the seed. The quality of the oil produced by the ME method will be good because the nutritional values will be preserved, but the yield will be very low. The solvent extraction method dissolves the oil in solvents such as hexane and acetone before extracting it in the next stage. This method is used in commercial production, but the chemicals used in this process are hazardous and may harm the environment. Non-conventional extraction techniques such as Enzyme Assisted Extraction (EAE), Supercritical Fluid Extraction (SFE), and extraction assisted by ultrasound (UAE) and microwaves (MAE) are also discussed in the literature [4]–[6].

The quality of the oil extracted using any of the above-discussed methods depends on its physicochemical properties and its composition. Physicochemical properties of edible oils are peroxide value, iodine value, unsaponifiable matter, saponification value, refractive index, viscosity, density, temperature, melting point, etc. The compositions of edible oils are mainly triacylglycerols (commonly referred to as triglycerides), accompanied by lower levels of diacylglycerols (diglycerides), monoacylglycerols (monoglycerides), free fatty acids, and some other minor components like sterols, carotenoids, and tocopherols [7]. Although oils are the sources of dietary lipids, they are also an important source of other essential dietary requirements. These minor components include phospholipids, phytosterols, tocols (tocopherols and tocotrienols, including vitamin E), and hydrocarbons.

The major edible oils consumed in India are mustard oil, sunflower oil, canola oil, sesame oil etc. The world production of soybean oil, rapeseed oil, palm oil, sunflower oil, coconut oil, and olive oil was 56.51, 27.3, 72.27, 21.2, 3.62, and 3.12 million metric tons respectively in 2019–2020 [8]. As the demand for edible oil increases, unfortunately, due to commercial interests, the quality of oils decreases. Some standards have been established to monitor and maintain the quality of the edible item. The Food Safety Standards Authority of India (FSSAI) has been established to

define science-based standards for food products as well as to regulate rules governing their manufacture, storage, distribution, sale, and import, thereby ensuring the availability of safe and wholesome food for human consumption. Unfortunately, some traders and manufacturing agencies fail to comply with standards for their profits. They do it at the cost of compromising with the quality of food and related articles with the addition of low-quality, low-cost ingredients or by removal of several vital high-cost components resulting in adulterated or counterfeit products in the food industry. Gaps between production, consumption, and the inability of regulatory authorities to check adulteration are big opportunities for fraudsters to make easy money through adulteration at the cost of nutrition and the health of consumers. Such adulteration in food not only reduces its nutritional quality but may affect its safety also, which may, in turn, affect the health of the consumers adversely. Food adulteration is a big threat to food quality and food safety worldwide [9].

In a developing country like India, the use of unpacked edible oils is still a common practice. A survey conducted by FSSAI by collecting samples from Delhi, Haryana, Uttar Pradesh, Gujarat, West Bengal, Bihar, Jharkhand, Maharashtra, Kerala, Tamil Nadu, Andhra Pradesh, Telangana, Karnataka, Rajasthan, and Madhya Pradesh, according to the FSSAI findings, 74.1% of cottonseed oil samples, 74% of sesame oil and 72% of mustard oil samples found adulterated. Out of the 230 soybean oil samples, 46 failed, and in the case of sunflower oil, 34 out of the 206 samples were adulterated [10].

2.3. Methods for analysis of edible oils and detection of adulteration

There are two methods for evaluation of the quality of edible oils, Subjective and Objective. Subjective methods are based on a human assessment of the quality characteristics of oils [11]–[13]. Human senses can assess the quality of an edible item by the attributes like color, taste, and aroma. Taste panels consisting of a highly trained group of professionals evaluate the quality. Still, this process is time-consuming, susceptible to large sources of variation, subjective to day-to-day variation, and depends on individual perception. Human sensory evaluation cannot detect adulteration very accurately and reliably in edible oils as the adulteration process is becoming sophisticated.

Objective methods for assessing quality are based on instrumental analysis techniques for data collection using multiple sensors, which are then combined with artificial pattern recognition algorithms on acquired data. They are immensely

beneficial because they are reliable, repeatable, non-subjective, and reproducible [13]–[15]. There are considerable benefits to developing instrumental methods to describe edible oil quality, which must be cost-effective and provide rapid, reproducible results while operating continuously. Due to their high cost and slower processing times, existing analytical techniques for assessing edible oil composition and adulteration detection are inadequate. Factors like promptness and low cost of analysis, minimal sample preparation, and environmentally friendly are important in selecting instrumental methods [16].

Gas chromatography, mass spectrometry, electrochemical methods such as voltammetry, electrochemical impedance spectroscopy, volatile compound analysis, electronic nose, and optical methods like NIR spectroscopy, ATR-based spectroscopy are among the analytical methods for edible oil analysis and detecting adulteration in edible oils. The following sections cover the extensive literature review on artificial senses, signal processing methods, and the research outcomes of the aforementioned analytical methods.

2.3.1. Chromatography methods for the analysis of edible oils

Gas chromatography is an analytical method for qualitative and quantitative analysis of samples in liquid and gaseous phases. Xing et al. [17] discussed detection of adulteration in sesame oil with rapeseed seed samples, soybean seed samples, sunflower oil, and maize oil by measuring fatty acid composition. Principal Component Analysis (PCA) and Discriminant Analysis (DA) algorithms on the GC data applied a minimum of 5% rapeseed oil or soybean oil adulteration in Sesame oil, a minimum of 10% adulteration of sunflower oil or maize oil identified. Using GC and applying PLSR, quantification of the proportion of adulterated oils in sesame oils is presented, and the Root means square errors of cross-validation (RMSECV), root mean square error of prediction (RMSEP), and adjusted R^2 is presented as 2.3732, 0.9808, 2.3228, and 0.9812 respectively [17].

C. Imai et al. [18] have discussed cottonseed oil adulteration with vegetable oils like soybean, rapeseed oil. Oils were analyzed for sterols, fatty acids, and triglycerides using gas chromatography. Sterol analysis profile and fatty acid profile used as an indication for adulteration are presented.

Luisito Cercaci et al. [19] has discussed the detection of adulteration of hazelnut oil in olive oil and the classification of olive oils from different origins by the

sensitive and precise determination of esterified sterols using GC. The solid-phase extraction (SPE) method was applied to an admixture with 10% of hazelnut oil and a screening of 11 oils of husk oil, virgin, and refined olive oils type for different regions.

Miloudi Hilali et al. [20] has presented a methodology for organ oil adulteration detection using GC. Determination of the campesterol level in adulterated oil samples (hazelnut, and apricot oil, sunflower oil) using the GC method was carried out. It is observed that the campesterol level in adulterated samples is higher than pure argan oils. It is observed a good correlation between measured and actual campesterol levels which suggests that GC campesterol level determination is 95% precise, reliable for organ oil adulteration detection.

Cristina Ruiz-Samblás et al. [21] has discussed the identification and quantification of the adulteration of olive oils blending with sunflower oil, corn seed oil, sesame oil, and soya oils. In this publication, the determination of triacylglycerol profiles of pure and adulterated edible oils using high-temperature GC was presented. A Soft Independent Modeling of Class Analogy (SIMCA) based classification model was successful in identifying the blending, and PLS based calibration model showed R^2 value between 0.95 to 0.98.

M. Hajimahmoodi et al. [22] used a PLS-based chemometric algorithm with the GC method for the determination of free fatty acid fingerprint of oils and oil mixtures. This method is claimed to be suitable for the determination of oils and oil mixtures fatty acid fingerprint measurement and thereby detection of adulteration. The relative standard error is less than 10% in each oil mixture.

Dang. Peng et al. [23] has presented a hierarchical approach for detecting and quantifying adulteration of sesame oil with vegetable oils using gas chromatography (GC) coupled with a chemometric algorithm. Ensemble support vector machine (SVM) models presented for classification of adulterated to non-adulterated sesame oil and identification of the type of adulteration followed by PLSR model for calibration of adulteration percentage. The prediction results from this work showed the detection limit low as 5% in mixing ratio and the RMSE for prediction range from 1.19% to 4.29%, stating that this approach is efficient to detect and quantify the adulteration in sesame oil.

Tanyaradzwa E. Mungure et al. [24] has discussed the use of High Pressure Liquid Chromatography (HPLC) in the detection of adulteration in cold-pressed oil. Cold pressing is a technique known to retain the nutritional and essential fatty acids naturally during extraction. The triacylglycerol (TAG) data is an indicator for quantitative assessment of the quality, adulteration in cold-pressed seed oils. Maria Fasciottiet al.[25] has presented adulteration of virgin olive oil with soybean oil using TAG as an indicator. TAG profile of soybean, olive oil, and a blend of both oils was characterized by HPLC-Atmospheric Pressure Chemical Ionization (HPLC-APCI) and HPLC-APCI-Mass Spectroscopy (HPLC-APCI-MS). PCA treatment of mass spectral data discriminates between Argentinean and European extra virgin olive oils and their adulteration with soybean oil [25].

H. Jabeur et al. [26] has studied extra virgin olive oil (EVOO) adulteration. A GC analysis for Trans Fatty Acid (TFA) composition permits the detection of a wider range of adulteration of 2% to 10%. Application of Linear Discriminant Analysis (LDA) is used for cheap and faster analysis to detect and quantify the possible adulterations of extra virgin olive oils (EVOO) by refined vegetable oils. H. Jabeur has also discussed the detection of extra virgin olive oil adulteration with soya seed oil, corn oil, and sunflower. HPLC method based on the fatty acid profile of edible oils is used for the detection of adulteration with a minimum detection level of 2%. LDA is used as a chemometric tool for adulteration detection.

2.3.2. Electrochemical methods for the analysis of edible oil

Electrochemical methods are also successful in edible oil analysis and adulteration detection. An electronic tongue or e-tongue consists of an array of non-specific, nonselective chemical sensors combined with appropriate data acquisition systems and chemometric algorithms for data processing [27]. Apetreia et al. [29] and [30], has discussed the detection of extra virgin olive oil adulteration detection using an electronic tongue based on square wave voltammetry. Chemicals like KCL, and HCL, Graphite were used in the carbon paste electrode preparation. Carbon Paste Electrode (CPE) was used as the working electrode, Platinum (Pt) wire was used as the counter electrode, and Ag/AgCl/KCl (saturated) as the reference electrode. Chemometric methods like Partial Least Square-Discriminant Analysis (PLS-DA) and Partial Least Square Regression (PLSR) algorithms on voltammetric data resulted in a successful classification of adulterated virgin olive oil with refined sunflower and soybean oils at a level of 10%, and the coefficient of regression R^2 is observed as 0.98.

Paolo Oliveri et al. [30] has discussed the development of voltammetric electronic tongue with platinum microelectrode along with the chemometrics algorithms on the voltammogram of oil and Room Temperature Ionic Liquids (RTIL) mixtures for discrimination of edible oils. Data processing using k- nearest neighbor algorithms and PCA showed a clear separation of olive oils of different origins. Hong men et al. [31] proposed data fusion of electronic tongue based on cyclic voltammetry and electronic nose for detecting the content of the frying oil mixing in edible oils. The application of PCA and PLSR on the fusion data showed the feasibility of detection of frying oils in edible oils.

Zhang Hang et al. [32] has presented a method for rapid detection of camellia seed oil adulteration with palm oil by using a voltammetry-based electronic tongue. Determination of adulteration is carried out using principal component analysis (PCA), discriminant factor analysis (DFA), and similarity analysis. From the results, it is observed that electronic tongue could be a good and quick method to detect the adulteration of camellia seed oil.

F. Tsopelas et al. [33] discussed voltammetric fingerprinting of edible oils combined with chemometrics for identification and calibration of virgin olive oil adulteration with sunflower, corn oils. Dichloromethane and $LiClO_4$ in absolute ethanol were used in sample preparation. Chemometric algorithms like PCA, PLS-DA, SIMCA showed clear detection of adulteration up to a minimum level of 2%. SIMCA results were a bit inferior to the results of PLS-DA.

Madiha Bougrin et al. [34] has presented detection of sunflower oil adulteration in argan oil by the voltammetric electronic tongue. The configuration had seven working electrodes, platinum (Pt), gold (Au), glassy carbon (GC), silver (Ag), nickel (Ni), palladium (Pd), and copper (Cu). Ag/Agcl was used as a reference electrode and platinum as the auxiliary electrode. Supervised multivariate data analysis methods such as PCA, DFA, and SVM were applied on sensor fused data, and the success rate of detection of adulteration was stated as 100% using an SVM classifier.

Haddi et al. [35] has presented a combination of voltammetric electronic tongue and electronic nose for olive oil classification. Tetra-butyl-ammonium tetra-fluoro-borate was used as a supporting electrolyte. Cyclic voltammetry (CV) experiments with potentials −700 to +1700 mV with a scan rate of 100 mV/s are conducted, resulting in a perfect identification of olive oils by PCA, CA, and SVM algorithms on e-tongue data.

Emre Ordukaya et al. [36] presented electronic nose as an alternative tool for the chemical analysis methods for olive oil adulteration detection. Data analysis is carried out with and without dimensionality reduction. PCA was used for dimensionality reduction, classification models like Bayesian, k-nearest neighbors (k-NN), Linear Discriminate Analysis (LDA), Decision Tree, Artificial Neural Networks (ANN), and Support Vector Machine (SVM) applied, and the classification efficiencies are compared, the best success rate was found with Bayes classifier (70.83% accuracy).

A study on the potential of volatile compounds analysis for detection of hazelnut oil adulteration in olive oil was presented by Sylwia Mildner-Szkudlarz et al. [37]. PCA was used for data analysis. Concepción Cerrato Oliveros et al. [38] presented an electronic nose containing 9 metal oxide semiconductor sensors for detection of adulteration in virgin olive oils with sunflower and olive-pomace oils. LDA, QDA, and Artificial neural network (ANN) based algorithms were successful in the detection of adulteration.

Xiaobao Wei et al. [39] proposed a rapid detection of adulteration in expensive peony oil with soybean oil, corn oil, sunflower oil, and rapeseed oil. Fatty acid composition and iodine value of peony oil soybean oil, corn oil, sunflower oil, and rapeseed oil were measured using GC and e-nose. E-nose was very fast in identifying adulteration with the application of PCA and LDA, a minimum detection of 10% of adulteration was reported.

Diego L et al. [40] proposed a metal oxide sensor-based electronic nose with mathematical modeling of the sensor data using a canonical equation for authenticating the quality of virgin olive oils using PCA and a developed model. All the extra virgin olive oils were correctly classified (100% accuracy). Peng Qi et al. [41] presented a geometrical region-based classification of camellia seed oils using an electronic nose. Application of PCA and PLS-DA on e-nose data demonstrated electronic nose fingerprinting for the authenticity of camellia oil.

Tomasz Majchrzak et al. [42] presented an exhaustive review on electronic nose application in classification and adulteration detection in edible oils supported by chemometric algorithms. Wojciech Wojnowskic et al. [43] proposed a portable, electronic nose system for food quality assessment. This prototype was able to detect the adulteration in olive oil and rapeseed oil with an accuracy of 82% using the SVM algorithm.

Table 2.1 shows some more literature on chemical sensors used in the electronic nose, electronic tongue, the combination of both, algorithms used, and the outcome in edible oil analysis and adulteration detection.

2.3.3. Spectroscopy methods for edible oil analysis

Spectroscopy method like infrared spectroscopy has the advantages of being rapid, green, chemical-free, non-destructive for the quality analysis of oilseeds and edible oils. Near infrared (NIR) can detect the compositions in oilseeds and edible oils in combination with advanced chemometrics. Basic knowledge of NIR spectroscopy and its applications in the quality of oilseeds and edible oils was emphasized by Xue li et al. [44].

K. N. Basri et al. [45] presented a portable NIR spectrometer for the detection of lard adulteration in palm oil. Transmittance and reflectance spectra were recorded. Classification of pure and adulterated oil samples with SIMCA algorithm showed an accuracy of 95%. Additionally, PLSR with a coefficient of determination (R^2) of 0.9987 and 0.9994 with root mean square error of calibration (RMSEC) of 0.5931 and 0.6703 respectively reported.

Recently Zhe Yuan et al. in [46] proposed a target detection approach for the detection of adulterations in flaxseed oils. A variable selection method was developed to reduce the number of variables for computational easiness. One-class partial least square (OCPLS) was built that could identify adulteration with blends of five different adulterants cottonseed oil, soybean oil, rapeseed oil, maize oil, and sunflower oil at a 5% level.

S. Farres et al. [47] discussed a combination of UV and NIR spectroscopy data coupled with a chemometric method like PLS-DA was used to detect adulteration detection. NIR Spectroscopy of 1 S duration was able to detect the argan oil from a 0.35% level of adulteration with two cheap oils. PLSR model was able to predict the concentration of adulterants with R^2 of 0.90, RMSEP, RMSEV 4.67, and 4.57, respectively.

Cleiton A.Nunes et al. [48] described NIR, Raman, and mainly MIR spectroscopy as a rapid and simple method to detect adulteration and thereby assess the authenticity in a variety of edible oils. Spectroscopy methods along with classification algorithms

like PCA, LDA SVM, PLSDA have been used, and their performances compared for NIR, MIR, and Raman spectroscopy.

Mutia Nurulhusna Hussain et al. [49] proposed a NIR spectroscopy using chemometrics for detecting canola oil adulteration with palm oil. LDA classification algorithm using open-source R software has shown classification accuracy of 100 % with a minimum detection level of 3.23 %.

Table 2.1 Application of electronic nose and electronic tongues in edible oil analysis

Reference	**Type of study**	**Chemical Sensors**	**Algorithms**	**Outcome**
Mildner-Szkudlarz and Jeleń (2008)[37]	Detection of adulteration with hazelnut oil	MOS sensors	PCA	Minimum detection level up to 5%
Mildner-Szkudlarz and Jeleń (2010)[50]	Detection of adulteration with rapeseed and sunflower oils	MOS sensors and SPME-MS	PCA, PLS	Minimum detection level up to 5%
Hai and Wang (2006a)[51]	Detection of adulteration Sesame oil	10 MOS sensors	LDA,PNN,BP-ANN,GRNN	LDA results were good than PNN. GRNN didn't work for detection
Marina et al. (2010)[52]	Detection of adulteration Virgin coconut oil	zNoseTM (SAW)	PCA	PCA with 91% variance R^2 above 0.97
Apetrei et al. (2005) [53]	Discrimination between olive oils of different qualities and discrimination between different vegetable oils	CPE with modified edible oils	PCA, kernel variable reduction	Clear discrimination among oils.
Apetrei et al. (2007) [28]	Discrimination between samples of different bitterness degrees. Prediction of sensorial bitterness degree obtained by a panel of experts. Prediction of chemical parameters (bitterness index, peroxide index, K indexes, and stability)	CPEs modified with 9 olive oils	PCA, PLS-DA, PLS, Kernell variable reduction	discrimination of the nine virgin olive oils according to their degree of bitterness.
Apetrei and Apetrei (2014) [29]	Discrimination between pure and adulterated oils. Prediction of the composition of seed oils and extra virgin olive oil mixtures	CPE modified with each edible oil studied	PCA, PLS-DA, PLS, kernel variable reduction	Correct classification. PCA and PLS-DA.
RodríguezMéndez et al. (2008b)[54]	Discrimination of samples according to their phenolic content and bitterness index. Correlation with the polyphenol content, the bitterness index (analyzed by chemical methods), and the bitterness degree	5 CPEs modified with lanthanide bisphthalocyanines, 6 polypyrrole conducting polymers, and 1 unmodified CP	PCA, PLS-DA, PLS, kernel variable reduction	PLS-DA, Correct classification. Calculation of bitterness index

	(determined by a panel of experts			
Apetrei (2012) [55]	Discrimination between samples of different bitterness degrees. Prediction of sensorial bitterness degree obtained by a panel of experts. Prediction of chemical parameters (bitterness index, free acidity, peroxide index, and K indexes)	6 polypyrrole based electrodes	PCA, PLS-DA, PLS, kernel variable reduction	Good correlations between multisensor system and the bitterness
Apetrei and Apetrei (2013[27]	Discrimination between samples of different total polyphenolic content. Prediction of total polyphenolic content. Discrimination between samples with different individual polyphenolic in extra virgin olive	6 polypyrrole based electrodes with different doping agents	PCA, PLS-DA, SIMCA, PLS, kernel variable reduction	Correct classification A high correlation between e tongue data polyphenyl content.
Dias et al. (2014) [56]	Discrimination between olive cultivars	2 units of 20 all-solid-state electrodes	LDA-SA	LDA-SA correctly classified
Veloso et al. (2016)[57]	Discrimination between intensity sensory perception levels	2 units of 20 all-solid-state electrodes with different pre-established mass combinations of 4 lipidic, 5 weight polymer	LDA-SA	Correct classification

Betül Öztürk et al. [58] presented a fast and rapid adulteration detection with NIR spectroscopy and chemometric algorithms. A genetic algorithm-based variable selection algorithm, coupled with an inverse least squares multivariate calibration method (GILS) was used. The correlation coefficient R^2 of 0.98 was observed.

Hui Chen et al. [59] described a fast and rapid adulteration detection in sesame oil using NIR and chemometrics methods like competitive adaptive reweighted sampling (CARS), elastic component regression (ECR), and partial least squares (PLS). PLS results with the complete spectrum are compared with PLS on selected variables obtained from the CARS and ECR algorithms. PLS model using only 10 variables from CARS and ECR showed similar performances, and they are better than complete spectrum PLS results.

Hormoz Azizian et al. [60] proposed a Fourier Transform Near-infrared (FT-NIR) methodology for rapid, accurate detection of adulteration in virgin olive oils with canola, sunflower, soyabean, peanut, palm. Using FT-NIR data, a PLS model was developed for free fatty acid content calculation in olive. Four linear regression

equation calculations of the fatty acid profile corresponding to each adulteration were developed.

X.Sun et al. [61] discussed the detection of virgin olive oil adulteration with soyabean, corn, camellia, and sunflower oil. Supervised locally linear embedding (SLLE) was used for dimensionality reduction followed by nearest centroid classification and PLS regression methods for classification and quantification.R^2 value using PLSR obtained from 0.9757 to 0.9988.

Amit et al. [62] applied ATR-FTIR spectroscopy for paraffin oil adulteration in virgin coconut oil, The model performances for PCR and PLSR for two different spectral windows (3000–2800cm^{-1}, and 1800–700cm^{-1}) were presented. Results showed clear detection of adulterations. R^2 value from the prediction model was 0.98. It is also reported in utilizing ATR-FTIR for the study of mustard oil adulteration in virgin coconut oil. It was observed from the study that PCA and LDA classified adulterated samples and PLSR resulted in R^2 (0.999), RMSEC (0.123), and RMSEP (0.125) values with the lowest detection level of 1%.

Rahul et al. [63] presented a rapid and non-destructive methodology of FTIR-spectroscopy for the detection of mixing fried mustard oil in pure mustard oil. Chemometric methods like PCA and LDA successfully separated and classified fried mustard oil adulterants using the PLSR model, R^2 (0.999), RMSE of 0.53%, and residual error (RE) of 3.37% values obtained.

Table 2.2 shows in tabular form details of the type of study, spectroscopy type, classification algorithms used in that study, and outcomes reported in the literature.

2.3.4. Chemometric methods for the analysis of edible oils

In edible oil analysis using different analytical instruments, chemometrics plays an important role. The term chemometrics describes the mathematical and statistical methods used to extract useful information from chemical data [64]. Nattane Luíza da Costa et al. [65] presented a review of chemometric methods and machine learning algorithms in food analysis. He presented chemometrics as a collection of several statistical algorithms to describe the sensory data and to predict inferences from the data, such as the pre-processing methods of variable selection, variable extraction, dimensionality reduction, and data modeling methods (clustering, classification, and regression).

Granato et al. [66] presented the unsupervised and supervised chemometric tools for qualitative and quantitative analysis of experimental data. PCA and HCA methods are unsupervised methods to evaluate whether clustering exists in a dataset without using class membership information. The principles of PCA and HCA and their applications in food analysis are presented.

Table 2.2 Application of spectroscopy methods for edible oil analysis

Reference	**Type of study**	**Spectroscopy**	**Algorithms**	**Outcome**
Shuifang Li et al. 2012[67]	Authentication of pure camellia oil	Near-Infrared	PCA,HCA,DA, RFBNN	Correct classification-98.3% accuracy
Chen et al. [68]	Classification of Soybean oil, palm oil, sesame oil, peanut oil	2D- Near Infrared	-	Correct classification
Ozeb et al.[69]	detect the adulteration of hazelnut oil with sunflower and olive oil	Fourier transform infrared spectroscopy (FT-IR)	PLS-DA PLSR	Correct detection Sunflower 2% Olive oil 25% minimum detection of hazelnut.
Feng et al.(2014) [70]	Adulteration of walnut oil with sunflower oil	synchronous fluorescence spectra	PLSR	Correct detection, minimum detection level 0.3%
YuanpengLi et al.2018 [71]	Detection and quantification of olive oil adulteration with waste cooking oil	Raman Spectroscopy	iPLS SiPLS	Correct Detection and correlation coefficient 0.98
Yang li et al. (2019) [72]	Quantitative analysis of olive oil adulteration with rapeseed oil	NIR MIR	SNV, SPA, SG, PLS-DA, PLSR	Data fusion results are better.
Karla Danielle et al.(2017)[73]	Detect and quantify adulteration of extra virgin olive oil with soybean edible oil	UV–Vis spectroscopies fluorescence	SPA, SPA-MLR, GA-MLR	fluorescence spectroscopy satisfactory results with detection of RMSEP 14.0 to 17.5 g/kg.
Konstantia Georgouli et al.(2017)[74]	Detection of adulteration hazelnut oil in extra virgin	MIR Raman	Continuous Locality Preserving Projections (CLPP),k-NN, SIMCA, PLS-DA	Correct classification, 80% accuracy
S.ok et al. (2017)[75]	olive oil adulteration	NMR, Uv,Vis	-	Correct Detection
Amit et al. 2020 [76]	Coconut oil adulteration detection	ATR-FTIR spectroscopy	PCA, LDA,PLSR	Correct detection R^2(0.997), RMSEC (1.001) and RMSEP (0.832)

M. Bevilacqua et al. [77] discussed supervised methods for classification or qualification methods and calibration or quantitation methods. Multivariate classification methods are designed to find mathematical models which recognize which class each sample belongs. They involve the use of various chemometric algorithms with two main statistical backgrounds related to classification and the class-modeling approaches. Applications of discriminant algorithms (LDA and QDA) in olive oil sample classification are explained

Jiménez-Carvelo et al. [78] presented the most commonly used multivariate methods in food analysis as conventional methods like PCA, PLS-DA, LDA, k-NN SIMCA. The algorithms alternative to conventional learning methods are SVM, ANN, CART, and Random Forest. The development of advanced analytical methods, with the alternative chemometric algorithms, resulted in better physical-chemical parameter information with a high level of detail compared to conventional algorithms.

Belén Vega-Márquez et al. [79] presented a deep learning approach for edible olive oil classification. Multilayer and unidirectional (feed-forward) neural networks were used, with an input layer, a hidden and an output layer. Accurate classification results were obtained with Rectified Linear Unit function (RELU) and Adam algorithm.

Xuewen Hou et al. [80] presented a convolution neural network approach for the classification of edible oils with NMR spectroscopy, 1D-CNN, and 2D-CNN results were compared. 1D-CNN was successful with 100% classification of edible oils.

2.3.5. Embedded systems for implementation inference models

Eduardo Garcia-Breijo et al. [81] presented a portable electronic tongue using an array of 18 thick-film electrodes array. Microchip PIC18F4550 microcontroller was used to implement an artificial neural network algorithm for the classification of water. A pre-trained model was implemented on a microcontroller. The results are compared with LDA and fuzzy ART algorithms.

Luis Gil-Sánchez et al. [82] presented a microcontroller-based electronic tongue system capable of discriminating between drinking water samples. A comparison of Fuzzy ARTMAP, a Multi-Layer Feed-Forward network (MLFF), and a Linear Discriminant Analysis (LDA) algorithms to obtain the best implementation on a microcontroller-based embedded platform. A pre-trained LDA algorithm was

implemented on a microcontroller and successfully discriminated against water samples with an accuracy of 82.5%.

S. Jayanthy et al. [83] have discussed a high-quality embedded system for assessing food quality using a histogram of oriented gradients(HOG). HOG algorithm, along with the SVM algorithm, was implemented to detect the fungus on food items. A webcam was used to capture the image, and features are extracted using the HOG algorithm. These features are fed to an SVM classifier for the detection of fungus. These algorithms were implemented on ARM cortex family microcontrollers.

B. Priyadharshini et al. [84] has deeloped an embedded system for the detection of adulterations in food. A sensor system and IoT platform were used to find the adulteration. pH sensors, temperature sensors, and NodeMCU microcontroller were used to build sensor nodes. Adulteration detection using pH change, the temperature change was done successfully, and the results are displayed on cloud platform using ThingspeakIoT.

Rajani K V et al. [85] have proposed a microcontroller embedded system for food quality detection using sensor technology. Temperature, pH, and gas sensors were used to monitor food quality. ARM microcontroller was used to acquire the sensor reading, and a threshold was set to detect the quality of the food samples. Results were processed and sent over wifi to display them on a web page.

Eduardo Garcia-Breijo et al. [86] have implemented Fuzzy-ARTMAP algorithm on an embedded platform to classify different types of honey samples. The microcontroller was programmed with the pre-trained parameters, and then new samples were analyzed using the portable system. Results are very good, with an 87.5% recognition rate showing the feasibility of embedded systems for food analysis.

Zhang et al. [87] have discussed the MobileNet-Single Shot Detector (SSD), an object detector model trained with the Caffe framework using a deep convolutional neural network. The pre-trained model was then deployed on FriendlyARM's NanoPi2, an ARM board with a Samsung Cortex-A9 Quad-Core S5P4418@1.4GHz SoC and 1 GB 32bit DDR3 RAM. MobileNet-SSD can operate at 1.13FPS.

Cerutti et al. [88].has used CMSIS-NN, an integrated library for the deployment of NNs on Cortex-M microcontrollers, to implement a CNN inference on STM Nucleo-L476RG for person detection. Weights are quantized to an 8-bit fixed point format to

minimize model size, which has an impact on performance. The network is ported with 20 KB of flash memory and 6 KB of RAM.

Respiro by Amiko [89] is a smart inhaler sensor with an ultra-low-power Cortex-M processor. This sensor interprets vibration data from an inhaler using machine learning. The processor enables the execution of machine learning algorithms in which the sensor is learned to identify breathing patterns and measure critical parameters. The information gathered is processed in an application, and feedback is obtained.

Literature for implementation of developed classification and adulteration detection in edible oils using analytical instrumentation is limited. The implementation of inference algorithms on a low-cost yet powerful embedded processor may lead to the development of cost-effective instruments for monitoring food quality.

2.4. Gaps in the existing research

Based on the extensive literature survey, following research gaps have been identified in the detection and quantification of adulterations in edible oils.

- There is a scope to develop a simple, nondestructive, rapid yet accurate method, with minimal sample preparation instrumental analytical method for the detection and quantification of adulterations in edible oils. Existing analytical instruments are complex, time-consuming, and require a lot of sample preparation time.

- There is a need for generic and accurate AI-based chemometric ensemble models for classification/identification of edible oils and detection of adulterations.

- To control the misuses of adulteration in edible oils, the system needs to be implemented at fields, so there is a scope for realization of developed chemometrics inference models on an embedded platform for the development of dedicated and portable intelligent analytical instrumentation systems to detect adulterations in edible oil.

Bibliography

[1] V. Kostik, S. Memeti, and B. Bauer, “fatty acid composition of edible oils and fats.”

[2] V. Ristić and G. Ristić, “Role and importance of dietary polyunsaturated fatty acids in the prevention and therapy of atherosclerosis,” *Medicinski pregled*, vol. 56, no. 1–2. Med Pregl, pp. 50–53, 2003, doi: 10.2298/MPNS0302050R.

[3] “A. K. Yusuf, "A Review of Methods Used for Seed Oil Extraction", International Journal of Science and Research (IJSR), https://www.ijsr.net/get_abstract.php?paper_id=1121804, Volume 7 Issue 12, December 2018, 233 - 238.

[4] S. P. J. Kumar, S. R. Prasad, R. Banerjee, D. K. Agarwal, K. S. Kulkarni, and K. V. Ramesh, “Green solvents and technologies for oil extraction from oilseeds,” *Chemistry Central Journal*, vol. 11, no. 1. BioMed Central Ltd., p. 9, Jan. 23, 2017, doi: 10.1186/s13065-017-0238-8.

[5] D. Nde and A. Foncha, “Optimization Methods for the Extraction of Vegetable Oils: A Review,” *Processes*, vol. 8, no. 2, p. 209, Feb. 2020, doi: 10.3390/pr8020209.

[6] B. A. S. Machado, C. G. Pereira, S. B. Nunes, F. F. Padilha, and M. A. Umsza-Guez, “Supercritical Fluid Extraction Using CO2: Main Applications and Future Perspectives,” *Separation Science and Technology (Philadelphia)*, vol. 48, no. 18. pp. 2741–2760, Dec. 2013, doi: 10.1080/01496395.2013.811422.

[7] Azadmard-Damirchi S, Torbati M. Adulterations in Some Edible Oils and Fats and Their Detection Methods. J. Food Qual. Hazards Control. 2015; 2 (2) :38-44.

[8] “ Statista - The Statistics Portal for Market Data, Market Research and Market Studies.” https://www.statista.com/ (accessed Feb. 10, 2021).

[9] D. P. Attrey, “Detection of food adulterants/contaminants,” in *Food Safety in the 21st Century: Public Health Perspective*, Elsevier Inc., 2017, pp. 129–143.

[10] S. Yadav, “Edible oil adulterations: Current issues, detection techniques, and health hazards,” 2018, doi: 10.13140/RG.2.2.27614.74564.

[11] H. E. Smyth, “The compositional basis of the aroma of Riesling and unwooded Chardonnay wine The Australian Wine Research Institute,” 2005.

[12] B. A. Harvey, M. S. Brauss, R. S. T. Linforth, and A. J. Taylor, “Real-time flavor release from foods during eating,” *ACS Symp. Ser.*, vol. 763, pp. 22–32, 2000, doi: 10.1021/bk-2000-0763.ch003.

[13] Y. Huang, Y. Lan, and R. E. Lacey, “Artificial senses for characterization of food quality,” *J. Bionic Eng.*, vol. 1, no. 4, pp. 159–173, Dec. 2004, doi: 10.1007/bf03399472.

[14] C. F. Ross, "Sensory science at the human-machine interface," *Trends in Food Science and Technology*, vol. 20, no. 2. Elsevier, pp. 63–72, Feb. 01, 2009, doi: 10.1016/j.tifs.2008.11.004.

[15] M. W. Heaven and D. Nash, "Recent analyses using solid phase microextraction in industries related to food made into or from liquids," *Food Control*, vol. 27, no. 1. Elsevier, pp. 214–227, Sep. 01, 2012, doi: 10.1016/j.foodcont.2012.03.018.

[16] H. Smyth and D. Cozzolino, "Instrumental methods (Spectroscopy, Electronic Nose, and Tongue) as tools to predict taste and aroma in beverages: Advantages and limitations," *Chem. Rev.*, vol. 113, no. 3, pp. 1429–1440, 2013, doi: 10.1021/cr300076c.

[17] C. Xing, X. Yuan, X. Wu, X. Shao, J. Yuan, and W. Yan, "Chemometric classification and quantification of sesame oil adulterated with other vegetable oils based on fatty acids composition by gas chromatography," *LWT*, vol. 108, pp. 437–445, Jul. 2019, doi: 10.1016/j.lwt.2019.03.085.

[18] C. Imai, H. Watanabe, N. Haga, and T. Ii, "Detection of adulteration of cottonseed oil by gas chromatography," *J. Am. Oil Chem. Soc.*, vol. 51, no. 7, pp. 326–330, Jul. 1974, doi: 10.1007/BF02633007.

[19] L. Cercaci, M. T. Rodriguez-Estrada, and G. Lercker, "Solid-phase extraction-thin-layer chromatography-gas chromatography method for the detection of hazelnut oil in olive oils by determination of esterified sterols," in *Journal of Chromatography A*, Jan. 2003, vol. 985, no. 1–2, pp. 211–220, doi: 10.1016/S0021-9673(02)01397-3.

[20] M. Hilali, Z. Charrouf, A. E. A. Soulhi, L. Hachimi, and D. Guillaume, "Detection of argan oil adulteration using quantitative campesterol GC-analysis," *JAOCS, J. Am. Oil Chem. Soc.*, vol. 84, no. 8, pp. 761–764, Aug. 2007, doi: 10.1007/s11746-007-1084-y.

[21] C. Ruiz-Samblás, F. Marini, L. Cuadros-Rodríguez, and A. González-Casado, "Quantification of blending of olive oils and edible vegetable oils by triacylglycerol fingerprint gas chromatography and chemometric tools," *J. Chromatogr. B Anal. Technol. Biomed. Life Sci.*, vol. 910, pp. 71–77, Dec. 2012, doi: 10.1016/j.jchromb.2012.01.026.

[22] M. Hajimahmoodi, Y. Vander Heyden, N. Sadeghi, B. Jannat, M. R. Oveisi, and S. Shahbazian, "Gas-chromatographic fatty-acid fingerprints and partial least squares modeling as a basis for the simultaneous determination of edible oil mixtures," *Talanta*, vol. 66, no. 5, pp. 1108–1116, Jun. 2005, doi: 10.1016/j.talanta.2005.01.011.

[23] D. Peng, Y. Bi, X. Ren, G. Yang, S. Sun, and X. Wang, "Detection and quantification of adulteration of sesame oils with vegetable oils using gas chromatography and multivariate data analysis," *Food Chem.*, vol. 188, pp. 415–421, May 2015, doi: 10.1016/j.foodchem.2015.05.001.

[24] T. E. Mungure, A. El-Din Bekhit, A. Carne, S. Roohinejad, K. Mallikarjunan, and J. Birch, "Application of HPLC in characterisation of triacylglycerols and detection of adulteration in cold pressed seed oils," in *Encyclopedia of Food Chemistry*, Elsevier, 2018, pp. 410–414.

[25] M. Fasciotti and A. D. Pereira Netto, "Optimization and application of methods of triacylglycerol evaluation for characterization of olive oil adulteration by soybean oil with HPLC-APCI-MS-MS," *Talanta*, vol. 81, no. 3, pp. 1116–1125, May 2010, doi: 10.1016/j.talanta.2010.02.006.

[26] H. Jabeur, A. Zribi, and M. Bouaziz, "Extra-Virgin Olive Oil and Cheap Vegetable Oils: Distinction and Detection of Adulteration as Determined by GC and Chemometrics," *Food Anal. Methods*, vol. 9, no. 3, pp. 712–723, Mar. 2016, doi: 10.1007/s12161-015-0249-9.

[27] I. M. Apetrei and C. Apetrei, "Voltammetric e-tongue for the quantification of total polyphenol content in olive oils," *Food Res. Int.*, vol. 54, no. 2, pp. 2075–2082, Dec. 2013, doi: 10.1016/j.foodres.2013.04.032.

[28] C. Apetrei, F. Gutierez, M. L. Rodríguez-Méndez, and J. A. de Saja, "Novel method based on carbon paste electrodes for the evaluation of bitterness in extra virgin olive oils," *Sensors Actuators, B Chem.*, vol. 121, no. 2, pp. 567–575, Feb. 2007, doi: 10.1016/j.snb.2006.04.091.

[29] I. M. Apetrei and C. Apetrei, "Detection of virgin olive oil adulteration using a voltammetric e-tongue," *Comput. Electron. Agric.*, vol. 108, pp. 148–154, Oct. 2014, doi: 10.1016/j.compag.2014.08.002.

[30] P. Oliveri, M. A. Baldo, S. Daniele, and M. Forina, "Development of a voltammetric electronic tongue for discrimination of edible oils," *Anal. Bioanal. Chem.*, vol. 395, no. 4, pp. 1135–1143, Oct. 2009, doi: 10.1007/s00216-009-3070-8.

[31] H. Men, D. Chen, X. Zhang, J. Liu, and K. Ning, "Data fusion of electronic nose and electronic tongue for detection of mixed edible-oil," *J. Sensors*, vol. 2014, 2014, doi: 10.1155/2014/840685.

[32] Z. H. S. W. R. Wei, "Rapid Detection of Adulterated Palm Oil in Camellia Seed Oil by Electronic Tongue," *FOOD Sci.*, vol. 34, no. 14, p. 218, Jul. 2013, doi: 10.7506/SPKX1002-6630-201314044.

[33] F. Tsopelas, D. Konstantopoulos, and A. T. Kakoulidou, "Voltammetric fingerprinting of oils and its combination with chemometrics for the detection of extra virgin olive oil adulteration," *Anal. Chim. Acta*, vol. 1015, pp. 8–19, Jul. 2018, doi: 10.1016/j.aca.2018.02.042.

[34] M. Bougrini, K. Tahri, Z. Haddi, T. Saidi, N. El Bari, and B. Bouchikhi, "Detection of adulteration in argan oil by using an electronic nose and a voltammetric electronic tongue," *J. Sensors*, vol. 2014, 2014, doi: 10.1155/2014/245831.

[35] Z. Haddi *et al.*, "Electronic nose and tongue combination for improved classification of Moroccan virgin olive oil profiles," *Food Res. Int.*, vol. 54, no. 2, pp. 1488–1498, Dec. 2013, doi: 10.1016/j.foodres.2013.09.036.

[36] E. Ordukaya and B. Karlik, "Quality Control of Olive Oils Using Machine Learning and Electronic Nose," *J. Food Qual.*, vol. 2017, 2017, doi: 10.1155/2017/9272404.

[37] S. Mildner-Szkudlarz and H. H. Jeleń, "The potential of different techniques for volatile compounds analysis coupled with PCA for the detection of the adulteration of olive oil with hazelnut oil," *Food Chem.*, vol. 110, no. 3, pp. 751–761, Oct. 2008, doi: 10.1016/j.foodchem.2008.02.053.

[38] M. C. Cerrato Oliveros, J. L. Pérez Pavón, C. García Pinto, M. E. Fernández Laespada, B. Moreno Cordero, and M. Forina, "Electronic nose based on metal oxide semiconductor sensors as a fast alternative for the detection of adulteration of virgin olive oils," *Anal. Chim. Acta*, vol. 459, no. 2, pp. 219–228, May 2002, doi: 10.1016/S0003-2670(02)00119-8.

[39] X. Wei, X. Shao, Y. Wei, L. Cheong, L. Pan, and K. Tu, "Rapid detection of adulterated peony seed oil by electronic nose," *J. Food Sci. Technol.*, vol. 55, no. 6, pp. 2152–2159, Jun. 2018, doi: 10.1007/s13197-018-3132-z.

[40] D. L. García-González and R. Aparicio, "Classification of different quality virgin olive oils by metal-oxide sensors," *Eur. Food Res. Technol.*, vol. 218, no. 5, pp. 484–487, Apr. 2004, doi: 10.1007/s00217-003-0855-4.

[41] Q. Peng *et al.*, "Discrimination of geographical origin of camellia seed oils using electronic nose characteristics and chemometrics," *J. fur Verbraucherschutz und Leb.*, vol. 15, no. 3, pp. 263–270, Sep. 2020, doi: 10.1007/s00003-020-01278-x.

[42] T. Majchrzak, W. Wojnowski, T. Dymerski, J. Gębicki, and J. Namieśnik, "Electronic noses in classification and quality control of edible oils: A review," *Food Chemistry*, vol. 246. Elsevier Ltd, pp. 192–201, Apr. 25, 2018, doi: 10.1016/j.foodchem.2017.11.013.

[43] W. Wojnowski, T. Majchrzak, T. Dymerski, J. Gębicki, and J. Namieśnik, "Portable Electronic Nose Based on Electrochemical Sensors for Food Quality Assessment," *Sensors*, vol. 17, no. 12, p. 2715, Nov. 2017, doi: 10.3390/s17122715.

[44] X. Li *et al.*, "Review of NIR spectroscopy methods for nondestructive quality analysis of oilseeds and edible oils," *Trends in Food Science and Technology*, vol. 101. Elsevier Ltd, pp. 172–181, Jul. 01, 2020, doi: 10.1016/j.tifs.2020.05.002.

[45] K. N. Basri, M. N. Hussain, J. Bakar, Z. Sharif, M. F. A. Khir, and A. S. Zoolfakar, "Classification and quantification of palm oil adulteration via portable NIR spectroscopy," *Spectrochim. Acta - Part A Mol. Biomol.*

Spectrosc., vol. 173, pp. 335–342, Feb. 2017, doi: 10.1016/j.saa.2016.09.028.

[46] Z. Yuan *et al.*, "Detection of flaxseed oil multiple adulteration by near-infrared spectroscopy and nonlinear one class partial least squares discriminant analysis," *LWT*, vol. 125, p. 109247, May 2020, doi: 10.1016/j.lwt.2020.109247.`

[47] S. Farres, L. Srata, F. Fethi, and A. Kadaoui, "Argan oil authentication using visible/near infrared spectroscopy combined to chemometrics tools," *Vib. Spectrosc.*, vol. 102, pp. 79–84, May 2019, doi: 10.1016/j.vibspec.2019.04.003.

[48] C. A. Nunes, "Vibrational spectroscopy and chemometrics to assess authenticity, adulteration and intrinsic quality parameters of edible oils and fats," *Food Research International*, vol. 60. Elsevier Ltd, pp. 255–261, Jun. 01, 2014, doi: 10.1016/j.foodres.2013.08.041.

[49] M. N. Hussain, M. F. A. Khir, M. H. Hisham, and Z. M. Yusof, "Feasibility study of detecting canola oil adulteration with palm oil using NIR spectroscopy and multivariate analysis," in *Proceedings of 2014 International Conference on Information, Communication Technology and System, ICTS 2014*, 2014, pp. 111–114, doi: 10.1109/ICTS.2014.7010567.

[50] Mildner-szkudlarz and h. H. Jeleń, "detection of olive oil adulteration with rapeseed and sunflower oils using mos electronic nose and smpe-ms," *J. Food Qual.*, vol. 33, no. 1, pp. 21–41, Feb. 2010, doi: 10.1111/j.1745-4557.2009.00286.x.

[51] Z. Hai and J. Wang, "Electronic nose and data analysis for detection of maize oil adulteration in sesame oil," *Sensors Actuators, B Chem.*, vol. 119, no. 2, pp. 449–455, Dec. 2006, doi: 10.1016/j.snb.2006.01.001.

[52] A. M. Marina, Y. B. Che Man, and I. Amin, "Use of the SAW Sensor Electronic Nose for Detecting the Adulteration of Virgin Coconut Oil with RBD Palm Kernel Olein," *J. Am. Oil Chem. Soc.*, vol. 87, no. 3, pp. 263–270, Mar. 2010, doi: 10.1007/s11746-009-1492-2.

[53] C. Apetrei, M. L. Rodríguez-Méndez, and J. A. De Saja, "Modified carbon paste electrodes for discrimination of vegetable oils," in *Sensors and Actuators, B: Chemical*, Nov. 2005, vol. 111–112, no. SUPPL., pp. 403–409, doi: 10.1016/j.snb.2005.03.041.

[54] M. L. Rodríguez-Méndez, C. Apetrei, and J. A. de Saja, "Evaluation of the polyphenolic content of extra virgin olive oils using an array of voltammetric sensors," *Electrochim. Acta*, vol. 53, no. 20, pp. 5867–5872, Aug. 2008, doi: 10.1016/j.electacta.2008.04.006.

[55] C. Apetrei, "Novel method based on polypyrrole-modified sensors and emulsions for the evaluation of bitterness in extra virgin olive oils," *Food Res. Int.*, vol. 48, no. 2, pp. 673–680, Oct. 2012, doi: 10.1016/j.foodres.2012.06.010.

[56] L. G. Dias, A. Fernandes, A. C. A. Veloso, A. A. S. C. Machado, J. A. Pereira, and A. M. Peres, "Single-cultivar extra virgin olive oil classification using a potentiometric electronic tongue," *Food Chem.*, vol. 160, pp. 321–329, Oct. 2014, doi: 10.1016/j.foodchem.2014.03.072.

[57] A. C. A. Veloso, L. G. Dias, N. Rodrigues, J. A. Pereira, and A. M. Peres, "Sensory intensity assessment of olive oils using an electronic tongue," *Talanta*, vol. 146, pp. 585–593, Jan. 2016, doi: 10.1016/j.talanta.2015.08.071.

[58] B. Öztürk, A. Yalçin, and D. Özdemir, "Determination of olive oil adulteration with vegetable oils by near infrared spectroscopy coupled with multivariate calibration," *J. Near Infrared Spectrosc.*, vol. 18, no. 3, pp. 191–201, Jun. 2010, doi: 10.1255/jnirs.879.

[59] H. Chen, Z. Lin, and C. Tan, "Fast quantitative detection of sesame oil adulteration by near-infrared spectroscopy and chemometric models," *Vib. Spectrosc.*, vol. 99, pp. 178–183, Nov. 2018, doi: 10.1016/j.vibspec.2018.10.003.

[60] H. Azizian, M. M. Mossoba, A. R. Fardin-Kia, P. Delmonte, S. R. Karunathilaka, and J. K. G. Kramer, "Novel, Rapid Identification, and Quantification of Adulterants in Extra Virgin Olive Oil Using Near-Infrared Spectroscopy and Chemometrics," *Lipids*, vol. 50, no. 7, pp. 705–718, Jul. 2015, doi: 10.1007/s11745-015-4038-4.

[61] X. Sun, W. Lin, X. Li, Q. Shen, and H. Luo, "Detection and quantification of extra virgin olive oil adulteration with edible oils by FT-IR spectroscopy and chemometrics," *Anal. Methods*, vol. 7, no. 9, pp. 3939–3945, May 2015, doi: 10.1039/c5ay00472a.

[62] Amit, R. Jamwal, S. Kumari, A. S. Dhaulaniya, B. Balan, and D. K. Singh, "Application of ATR-FTIR spectroscopy along with regression modelling for the detection of adulteration of virgin coconut oil with paraffin oil," *LWT*, vol. 118, p. 108754, Jan. 2020, doi: 10.1016/j.lwt.2019.108754.

[63] R. Jamwal *et al.*, "Rapid and non-destructive approach for the detection of fried mustard oil adulteration in pure mustard oil via ATR-FTIR spectroscopy-chemometrics," *Spectrochim. Acta - Part A Mol. Biomol. Spectrosc.*, vol. 244, p. 118822, Jan. 2021, doi: 10.1016/j.saa.2020.118822.

[64] K. Varmuza and P. Filzmoser, *Introduction to Multivariate Statistical Analysis in Chemometrics*. CRC Press, 2016.

[65] N. L. da Costa, M. S. da Costa, and R. Barbosa, "A Review on the Application of Chemometrics and Machine Learning Algorithms to Evaluate Beer Authentication," *Food Analytical Methods*, vol. 14, no. 1. Springer, pp. 136–155, Jan. 01, 2021, doi: 10.1007/s12161-020-01864-7.

[66] D. Granato, J. S. Santos, G. B. Escher, B. L. Ferreira, and R. M. Maggio, "Use of principal component analysis (PCA) and hierarchical cluster analysis (HCA)

for multivariate association between bioactive compounds and functional properties in foods: A critical perspective," *Trends in Food Science and Technology*, vol. 72. Elsevier Ltd, pp. 83–90, Feb. 01, 2018, doi: 10.1016/j.tifs.2017.12.006.

[67] S. Li, X. Zhu, J. Zhang, G. Li, D. Su, and Y. Shan, "Authentication of Pure Camellia Oil by Using Near Infrared Spectroscopy and Pattern Recognition Techniques," *J. Food Sci.*, vol. 77, no. 4, pp. C374–C380, Apr. 2012, doi: 10.1111/j.1750-3841.2012.02622.x.

[68] B. Chen, P. Tian, D. L. Lu, Z. Q. Zhou, and M. L. Shao, "Feasibility study of discriminating edible vegetable oils by 2D-NIR," *Anal. Methods*, vol. 4, no. 12, pp. 4310–4315, Dec. 2012, doi: 10.1039/c2ay25962a.

[69] B. F. Ozen and L. J. Mauer, "Detection of hazelnut oil adulteration using FT-IR spectroscopy," *J. Agric. Food Chem.*, vol. 50, no. 14, pp. 3898–3901, Jul. 2002, doi: 10.1021/jf0201834.

[70] F. Ge, C. Chen, D. Liu, and S. Zhao, "Rapid Quantitative Determination of Walnut Oil Adulteration with Sunflower Oil Using Fluorescence Spectroscopy," *Food Anal. Methods*, vol. 7, no. 1, pp. 146–150, Jan. 2014, doi: 10.1007/s12161-013-9610-z.

[71] Y. Li, T. Fang, S. Zhu, F. Huang, Z. Chen, and Y. Wang, "Detection of olive oil adulteration with waste cooking oil via Raman spectroscopy combined with iPLS and SiPLS," *Spectrochim. Acta - Part A Mol. Biomol. Spectrosc.*, vol. 189, pp. 37–43, Jan. 2018, doi: 10.1016/j.saa.2017.06.049.

[72] Y. Li, Y. Xiong, and S. Min, "Data fusion strategy in quantitative analysis of spectroscopy relevant to olive oil adulteration," *Vib. Spectrosc.*, vol. 101, pp. 20–27, Mar. 2019, doi: 10.1016/j.vibspec.2018.12.009.

[73] K. D. T. M. Milanez, T. C. A. Nóbrega, D. S. Nascimento, M. Insausti, B. S. F. Band, and M. J. C. Pontes, "Multivariate modeling for detecting adulteration of extra virgin olive oil with soybean oil using fluorescence and UV–Vis spectroscopies: A preliminary approach," *LWT*, vol. 85, pp. 9–15, Nov. 2017, doi: 10.1016/j.lwt.2017.06.060.

[74] K. Georgouli, J. Martinez Del Rincon, and A. Koidis, "Continuous statistical modelling for rapid detection of adulteration of extra virgin olive oil using mid infrared and Raman spectroscopic data," *Food Chem.*, vol. 217, pp. 735–742, Feb. 2017, doi: 10.1016/j.foodchem.2016.09.011.

[75] S. Ok, "Detection of olive oil adulteration by low-field NMR relaxometry and UV-Vis spectroscopy upon mixing olive oil with various edible oils," *Grasas y Aceites*, vol. 68, no. 1, pp. e173–e173, Jan. 2017, doi: 10.3989/gya.0678161.

[76] Amit *et al.*, "Utilizing ATR-FTIR spectroscopy combined with multivariate chemometric modelling for the swift detection of mustard oil adulteration in virgin coconut oil," *Vib. Spectrosc.*, vol. 109, p. 103066, Jul. 2020, doi:

10.1016/j.vibspec.2020.103066.

[77] M. Bevilacqua, R. Nescatelli, R. Bucci, A. D. Magrì, A. L. Magrì, and F. Marini, "Chemometric classification techniques as a tool for solving problems in analytical chemistry," *J. AOAC Int.*, vol. 97, no. 1, pp. 19–28, Jan. 2014, doi: 10.5740/jaoacint.SGEBevilacqua.

[78] A. M. Jiménez-Carvelo, A. González-Casado, M. G. Bagur-González, and L. Cuadros-Rodríguez, "Alternative data mining/machine learning methods for the analytical evaluation of food quality and authenticity – A review," *Food Research International*, vol. 122. Elsevier Ltd, pp. 25–39, Aug. 01, 2019, doi: 10.1016/j.foodres.2019.03.063.

[79] B. Vega-Márquez, I. Nepomuceno-Chamorro, N. Jurado-Campos, and C. Rubio-Escudero, "Deep Learning Techniques to Improve the Performance of Olive Oil Classification," *Front. Chem.*, vol. 7, p. 929, Jan. 2020, doi: 10.3389/fchem.2019.00929.

[80] X. Hou, G. Wang, X. Wang, X. Ge, Y. Fan, and S. Nie, "Convolutional neural network based approach for classification of edible oils using low-field nuclear magnetic resonance," *J. Food Compos. Anal.*, vol. 92, p. 103566, Sep. 2020, doi: 10.1016/j.jfca.2020.103566.

[81] E. Garcia-Breijo *et al.*, "A comparison study of pattern recognition algorithms implemented on a microcontroller for use in an electronic tongue for monitoring drinking waters," *Sensors Actuators, A Phys.*, vol. 172, no. 2, pp. 570–582, Dec. 2011, doi: 10.1016/j.sna.2011.09.039.

[82] L. Gil-Sánchez *et al.*, "Embedded pattern recognition systems for liquids classification: A comparison study," in *Proceedings of IEEE Sensors*, 2011, pp. 1720–1723, doi: 10.1109/ICSENS.2011.6126921.

[83] S. Jayanthy, S. V. Ayswaria, M. Vaishali, and P. Udhaya Poorani, "a high quality embedded system for assessing food quality using histogram of oriented gradients," *ictact j. Soft Comput.*, p. 1, 2018, doi: 10.21917/ijsc.2018.0248.

[84] B. Priyadharshini, V. Ramya, S. S. Jayasree, and L. A. . Isabella, "Detection of Food Adulteration Using Embedded System", IJRESM, vol. 3, no. 8, pp. 554–558, Sep. 2020..

[85] R. K. V, A. Ashfaq, and A. R. Kumar Nayaka, "Embedded based Food Quality Detection with Sensor Technology," *SJ Impact Factor 6*, vol. 887, 2018, doi: 10.22214/ijraset.2018.5078.

[86] E. Garcia-Breijo, J. Garrigues, L. Sanchez, and N. Laguarda-Miro, "An Embedded Simplified Fuzzy ARTMAP Implemented on a Microcontroller for Food Classification," *Sensors*, vol. 13, no. 8, pp. 10418–10429, Aug. 2013, doi: 10.3390/s130810418.

[87] Y. Zhang, S. Bi, M. Dong, and Y. Liu, "The implementation of CNN-based

object detector on ARM embedded platforms," in *Proceedings - IEEE 16th International Conference on Dependable, Autonomic and Secure Computing, IEEE 16th International Conference on Pervasive Intelligence and Computing, IEEE 4th International Conference on Big Data Intelligence and Computing and IEEE 3*, 2018, pp. 379–382, doi: 10.1109/DASC/PiCom/DataCom/CyberSciTec.2018.00074.

[88] G. Cerutti, R. Prasad, and E. Farella, "Convolutional Neural Network on Embedded Platform for People Presence Detection in Low Resolution Thermal Images," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, May 2019, vol. 2019-May, pp. 7610–7614, doi: 10.1109/ICASSP.2019.8682998.

[89] Brian Fuller, "AI Technology Helping Asthma Sufferers Breathe Easier," 2018. Accessed: Apr. 27, 2021. [Online]. Available: https://www.hackster.io/news/ai-technology-helping-asthma-sufferers-breathe-easier-50775aa7b89f.

Chapter 3

Investigation of Instrumental Methods for Analysis of Edible Oils

3.1. Preamble

Edible oils play a vital role in the human diet as sources of essential fatty acids (EFA)and minerals. These essential fatty acids are not synthesized in the body and must be supplied through the diet to maintain the integrity of cell membranes. To meet the quality standards manufacturers measure (monitor) certain oil parameters to ensure the quality of edible oils. Standards like ISO9001-2008 (Quality Management System (QMS) in food manufacturing), ISO 22000 (Food Safety Management Systems), and Food Safety and Standards Act, 2006 have emerged [1], [2].

The quality of edible oil is determined based on sensory evaluation, chemical composition, physical properties, the level of microbiological and toxic contamination. Earlier human senses were used for quality analysis by smell, taste, and appearance. In some cases, the senses may be incapable of accurately and reliably assessing quality, as in the case of adulteration of edible oils, which is of particular interest to us. Later, chemical methods based on American Oil Chemist Society (AOCS) standards were used to assess the quality and detect adulteration in edible oils. Following this, the use of analytical instruments for edible oil analysis increased. The development of new analytical methods that accurately mimic the human senses is on the rise. Compared to human senses, analytical instruments have the advantage of providing an instant and unbiased evaluation of edible oil analysis.

In this chapter, the authentication of edible oils based on standard AOCS chemical methods by measuring physicochemical properties is presented. The results of AOCS methods have been compared to the specifications of edible oils provided by the Food Safety and Standards Authority of India (FSSAI) to ensure their authenticity and check for no adulteration in the samples. Later, these authenticated edible samples have been used for edible oil analysis using electrochemical methods such as Voltammetry, Electrochemical Impedance Spectroscopy (EIS), and Spectroscopy methods in Near-Infrared (NIR), Mid Infrared (MIR) ranges. We have used these methods in our research work.

A detailed description of mentioned analytical methods covering their principle of operation, instrumentation deployed, sample preparation, experiment methodologies are presented in this chapter.

3.2. Chemical analysis of edible oils for quality check

The edible oil samples for experimental work in this research were obtained from the local market. Chemical analysis of edible oil samples was performed at the Council of Scientific and Industrial Research–Indian Institute of Chemical Technology (CSIR-IICT) in Hyderabad, a FSSAI certified laboratory for edible oil analysis. These chemical analysis methods were designed in accordance with AOCS (American Oil Chemists Society) standards for the physicochemical characterization of edible oils. The properties of edible oils recommended by FSSAI for quality analysis were evaluated using these AOCS methods. These properties include free fatty acid and trans-fatty acid composition, acidity value, peroxide value, saponification value, iodine value, and refractive index, which ensures the quality and authenticity of edible oils when compared with the specifications provided by FSSAI for each edible oil.

3.2.1. Sample collection

Edible oil samples of different brands, namely groundnut oil (GNUT), canola oil (CAN), mustard oil (MUS), olive oil (OL), safflower oil (SAFF), soya oil (SOYA), sunflower oil (SUN), palm oil (PALM), rice bran oil (RB), sesame oil (SES) and cottonseed oil (COT) were purchased from a local vendor. All edible oils were stored in an airtight container in a dark place to avoid oxidation and rancidity of oils because of moisture in the air. Table 3.1 lists the edible oils and brands that have been used in experiments.

Table 3.1 Edible oil sample collected for the Experiment

S.NO	Edible Oil	Indicator	Brand (Manufacturer)	Quantity
1	Soybean Oil	SOYA	Fortune	(2Lt)
2	Sunflower Oil	SUN	Fortune	(2Lt)
3	Rice Bran Oil	RB	Freedom	(2Lt)
4	Mustard Oil	MUS	Fortune	(2Lt)
5	Palm Oil	PALM	Ruchi Gold	(2Lt)
6	Groundnut Oil	GNUT	Freedom	(2Lt)
7	Canola Oil	CAN	Disano	(2Lt)
8	Safflower Oil	SAFF	Daana	(2Lt)
9	Sesame Oil	SES	Dabur	(2Lt)
10	Olive Oil	OL	Borges	(2Lt)
11	Cottonseed Oil	COT	Dabur	(2Lt)

Next sections explain the methodology for measuring physicochemical properties in laboratories using AOCS standards.

3.2.2. Determination of Free Fatty Acid (FFA) and Trans Fatty Acids (TFA)

The fatty acid composition of edible oils was determined after they were derivatized to Fatty Acid Methyl Esters (FAME) using a method developed in the laboratory by Kanjilal et al. [3]. According to this procedure, about 1 ml of stock solution was taken in 5 ml of 2% H_2SO_4 in methanol and refluxed for 3-4 hours. Methanol was evaporated in a rotary evaporator after complete conversion (as determined by Thin Layer Chromatography), and the FAME was solubilized in ethyl acetate. The ethyl acetate layer was washed twice with water, dried over anhydrous Na_2SO_4, concentrated in a rotary evaporator, and kept ready for GC analysis.

A Hewlett-Packard 6890 Series Gas Chromatograph (GC) with a flame ionization detector was used for the GC analysis. Nitrogen (N_2) was used as the carrier gas in a fused-silica capillary column DB225 (30 m × 0.25 mm i. d.× 0.20 μm film thickness). The column was operated at 160°C for 2 minutes, then raised to 230°C at a rate of 5°C/min for 20 minutes until the analysis was completed. The injection port and detector temperatures were kept at 230°C and 260°C, respectively, and the split ratio (column carrier gas flow rate divided by vent flow rate) was set to 10:1. A standard fatty acid methyl ester mixture was injected to identify the peak (Supelco FAME Mix: Cat No. CRM47885). Using HP ChemStation software (Hewlett-Packard), the relative percentages of individual fatty acids were identified and quantified. The presence of α-linolenic acid (ALA), γ-linolenic acid, and other polyunsaturated fatty acids were confirmed by GC-MS (Gas Chromatography-Mass Spectrometry).

The trans-fat content of oil was determined using GC with Flame-Ionization Detection (GC-FID) as their methyl esters (AOCS Cd 14c-94) on an HP 88 column (100 m x 0.25 mm x 20 m). The program is as follows: Oven temperature at 165°C for 15 minutes, then increased to 200°C at 5°C/min with a 35-minute hold time. Nitrogen (N_2) gas was used as carrier gas with a flow rate of 1 ml/min. Temperatures of the injector and detector were maintained at 250°C and 280°C, respectively. Trans fatty acid identification was accomplished through injecting and comparison with standard trans fatty acid samples.

3.2.3. Determination of Unsaponifiable Matter (USM)

Unsaponifiable matters in extracted oil are those that cannot be saponified by caustic treatment but are soluble in common fat and oil (AOCS Ca 6a-40; 2011). The USM of the oils and fats was determined using the AOCS official method Ca 6a-40, and the content was quantified using GC against the internal standard 6α-cholestane [4]. This method determines an unsaponifiable matter, which includes substances commonly found dissolved in oils, such as higher aliphatic alcohols, sterols, and hydrocarbons.

In a round bottom flask, the sample (oil, 5g) was taken, and 60 ml of absolute ethanol and 50 percent aqueous KOH solution (5 ml) were added. The mixture was refluxed for 10-12 hours before being evaporated to a volume of 10 ml in a rotary evaporator. The material was then transferred into a separating funnel using a 1:1, v/v ratio of water and hexane (40 ml). The contents were cooled, and 50 ml of hexane was added. The separating funnel was shaken vigorously for 1 minute to separate the two layers. The organic layer was separated, and the aqueous layer was extracted at least six times, each time with 50 ml of fresh hexane and vigorous shaking. Water was used to wash the combined hexane layer until it became neutral. The hexane layer was then dried over anhydrous sodium sulphate and evaporated to obtain a constant weight (A). The free fatty acid content was determined by titrating with 0.02 N NaOH and determining the free fatty acid content (B). The amount of unsaponifiable matter was calculated as

$$\text{Unsaponifiable matter (\%)} = \frac{(\text{A}-\text{B})*100}{\text{Mass of the sample(g)}} \tag{3.1}$$

Where, A = Mass of residue and B = Mass of fatty acids

3.2.4. Determination of Acid Value (AV)

The acid value is defined as the number of milligrams of potassium hydroxide required to neutralize the free fatty acids present in one gram of fat. This value is also expressed as a percentage of free fatty acids calculated as oleic acid, lauric acid, ricinoleic acid, and palmitic acid. The acid value was determined by titrating the oil/fat in an alcoholic medium directly against a standard potassium hydroxide/sodium hydroxide solution (AOCS Cd 3d-63; 2011).

A 5-gram oil mixture was added in a 250 ml conical flask, and 50 ml to 100 ml of freshly neutralized hot ethyl alcohol and about 1ml of phenolphthalein indicator solution was added. The mixture was boiled for around five minutes and titrated when hot against normal alkali solution, shaking vigorously during the titration. The weight

of the oil/fat used, as well as the strength of the alkali used for titration, must be such that the amount of alkali used for titration does not exceed 10 ml. The acid value is calculated as

$$\text{Acid value} = \frac{56.1 * V * N}{W} \tag{3.2}$$

where,

V = Volume in ml of standard potassium hydroxide or sodium hydroxide used
N = Normality of the potassium hydroxide solution or sodium hydroxide solution
W = Weight of the sample in grams

3.2.5. Determination of Saponification Value (SV)

The saponification value is the number of milligrams of potassium hydroxide (KOH) needed to saponify one gram of oil and was determined using the standard method described in AOCS Cd 3-25; 2011. The saponification value is a measure of the mean molecular weight of the fatty acids and glycerides comprising a fat. Lower the saponification value, higher the molecular weight of fatty acids in glycerides, and vice versa. The oil sample is saponified by refluxing it with a known alcoholic KOH solution. The amount of alkali available for saponification is determined by titrating excess KOH with standard hydrochloric acid (HCl).

If the sample is not already liquid, melt it and filter it through filter paper to remove any impurities and last traces of moisture. Check to see if the sample is completely dry. Thoroughly mix the sample and weigh 1.5 to 2.0 gm of dry sample into a 250 ml Erlenmeyer flask. Fill the flask with 25 ml of the alcoholic KOH solution. Along with the sample, perform a blank determination. Connect the air condensers to the sample and blank flasks; continue to boil on the water bath, gently and steadily, until saponification is complete, as indicated by the absence of any oily matter and the appearance of a clear solution. Clarity can be obtained after only one hour of boiling. After the flask and condenser have cooled down, wash the inside of the condenser with about 10 ml of hot ethyl alcohol neutral to phenolphthalein. Titration with 0.5N HCl with approximately 1.0 ml phenolphthalein indicator determines the excess KOH.

$$\text{Saponification value} = \frac{56.1 * (B - S) * N}{W} \tag{3.3}$$

Where,

B = Volume in ml of standard HCl required for the blank.
S = Volume in ml of standard HCl required for the sample
N = Normality of the standard HCl

W = Weight of the oil taken for the test in grams.

3.2.6. Determination of Refractive Index (RI)

The refractive index is the ratio of the velocity of light in vacuum to the velocity of light in liquid. More broadly, it is the ratio of the sine of the angle of incidence to the sine of the angle of refraction when a known wavelength of light (typically 589.3 nm, the mean of D lines of Sodium) passes from air into oil. Temperature and wavelength affect the refractive index. A Refractometer - Abbe or Butyro Refractometer is used to calculate the refractive index of the sample. The instrument is first calibrated with distilled water, which has a refractive index of 1.3330 at 20°C and 1.3306 at 40°C. The refractive index of vegetable oil is calculated and recorded at 40°C in accordance with regulations. The measurement was carried out following the protocol outlined in the FSSAI manual.

3.2.7. Determination of Peroxide Value (PV)

Peroxide value indicates the degree of oxidation suffered by oil and was calculated using the standard process outlined in AOCS Cd 8b-90. In a 250 ml stoppered conical flask, correctly measure 5 gm of oil sample, then apply 30 ml of acetic acid: chloroform (3:2, v/v) solvent mixture and swirl to dissolve. Using a mohr pipette, add 0.5 ml of saturated potassium iodide (KI) solution. Allow for one minute in the dark with intermittent shaking until 30 ml of water is added.

Titrate the liberated iodine (I_2) with 0.1 N sodium thiosulphate ($Na_2S_2O_3$) solution slowly and vigorously until the yellow color is almost gone. Add about 0.5 ml starch solution as an indicator and continue titration, shaking vigorously to release all I_2 from the chloroform layer until the blue color disappears. If less than 0.5 ml of 0.1 N sodium thiosulphate is used, repeat with 0.01 N sodium thiosulphate. Perform a blank decision (must be less than 0.1 ml 0.1 N sodium thiosulphate). Peroxide value expressed as milliequivalent of peroxide oxygen per kg sample (meq/kg)

$$\text{Peroxide value} = \frac{\text{Titre value} * \text{N} * 1000}{\text{Sample weight}} \tag{3.4}$$

Where, titre value is the ml of Sodium Thiosulphate used (blank corrected) and N is Normality of sodium thiosulphate solution.

3.2.8. Determination of Iodine Value (IV)

The iodine value of an oil is the amount of iodine absorbed by 100gm of oil when measured with Wijs solution. The carbon tetrachloride (CCl_4)-treated oil sample is treated with a known excess of iodine monochloride solution in glacial acetic acid (Wijs solution). The excess iodine monochloride is treated with potassium iodide, and the liberated iodine is calculated using a titration with sodium thiosulfate ($Na_2S_2O_3$) solution (AOCS Cd 1-25; 2011).

To summarize, weigh an appropriate amount of dry oil into a 500 ml glass stoppered conical flask containing 25 ml of CCl_4 and mix thoroughly. The sample weight should be such that there is an excess of 50 to 60% Wijs solution over that actually needed. After wetting with a potassium iodide solution, pipette 25 ml of Wijs solution and replace the glass stopper. Swirl the flasks to ensure proper mixing and place them in the dark for 30 minutes for non-drying and semi-drying oils and one hour for drying oils. Carry out a blank determination without an oil sample at the same time. Add 15 ml of potassium iodide solution, then 100 ml of boiled and cooled water, rinsing in the stopper also. Titrate the liberated iodine with standardized sodium thiosulphate ($Na_2S_2O_3$) solution, using starch as an indicator, until the blue color formed vanishes after thorough shaking with the stopper on. Carry out blank determinations in the same manner as test sample determinations, but without the addition of oil. Because chloroform has a high coefficient of expansion, slight temperature changes have a significant effect on the titre of iodine solution. It is necessary that blanks, and determinations are made at the same time.

$$\text{Iodine Value} = \frac{12.69*(B-S)*N}{W} \tag{3.5}$$

Where,

B = Volume in ml of standard $Na_2S_2O_3$ solution required for the blank.
S = Volume in mL of standard $Na_2S_2O_3$ solution required for the sample.
N = Normality of the standard $Na_2S_2O_3$ solution.
W = Weight of the sample in grams.

The physicochemical properties of collected edible oils were evaluated, and the results are shown in Table 3.2. Table 3.3 lists the specifications of edible oil properties that are permissible for edible oils by FSSAI. For better visualization of results, the measured physicochemical properties and the FSSAI specifications were presented in Figures 3.1, 3.2, 3.3 and 3.4.

Table 3.2 Physicochemical properties measured using AOCS methods at CSIR-IICT Hyderabad.

Oil	**Acid value**	**Iodine value**	**Peroxide value**	**Sap. value**	**Unsap. matter**	**Refractive Index**
Soybean	0.06	132.4	1.88	190.4	0.83	1.4671
Sunflower	0.01	132.4	0.94	188.9	0.80	1.4671
Rice Bran	0.41	100.1	2.81	184.3	3.92	1.4664
Mustard	0.72	108.6	3.78	173.7	0.84	1.4660
Palm	0.24	55.9	1.98	196.5	0.83	1.4589
Groundnut	2.63	92.0	10.1	192.5	0.84	1.4632
Olive	0.46	84.8	8.7	192.8	0.19	1.4618
Canola	0.4	148	0.85	192	1.8	1.472
Safflower	0.5	141	2.81	188	0.7	1.4749
Sesame	0.45	110	2.7	178	1.96	1.4752
Cottonseed	0.55	116	3.6	196	1.39	1.4704

Table 3.3 FSSAI specifications of all the studied oils

Oil	**Acid Value (AV)**	**Iodine Value (IV)**	**Peroxide Value (PV)**	**Saponification Value SV)**	**Unsap Matter**	**Refractive Index (RI)**
Soybean	≤ 2.5	120-141	Not more than 10 ppm	189-195	≤ 1.5%	1.4649-1.4710
Sunflower	≤ 6.0	100-145		188 - 194	≤ 1.5%	1.4640 - 1.4691
Rice Bran	≤ 0.5	90-105		180 - 195	≤ 3.5% (Chem.Ref) ≤ 4.5% (Phy. Ref)	1.4600 - 1.4700
Mustard	≤ 6.0	105-126		168 - 177	≤ 1.2%	1.4646 - 1.4662
Palm	≤ 6.0	54-62		195-205	≤ 1.2%	1.4550 - 1.4610
Groundnut	≤ 6.0	85-99		188 - 196	≤ 1.0%	1.4620 - 1.4640
Saff Flower	<6.0	135-148		186-196	<1%	1.4674-1.4689
Sesame	<6.0	103-120		169-182	<1.5%	1.4646-1.4665
Olive	≤ 6.0	75-94		184 - 196	≤ 1.5%	1.4600 - 1.4630
Cottonseed	<0.5	98 - 123		190 to 198	<1.5%	1.4630-1.4660

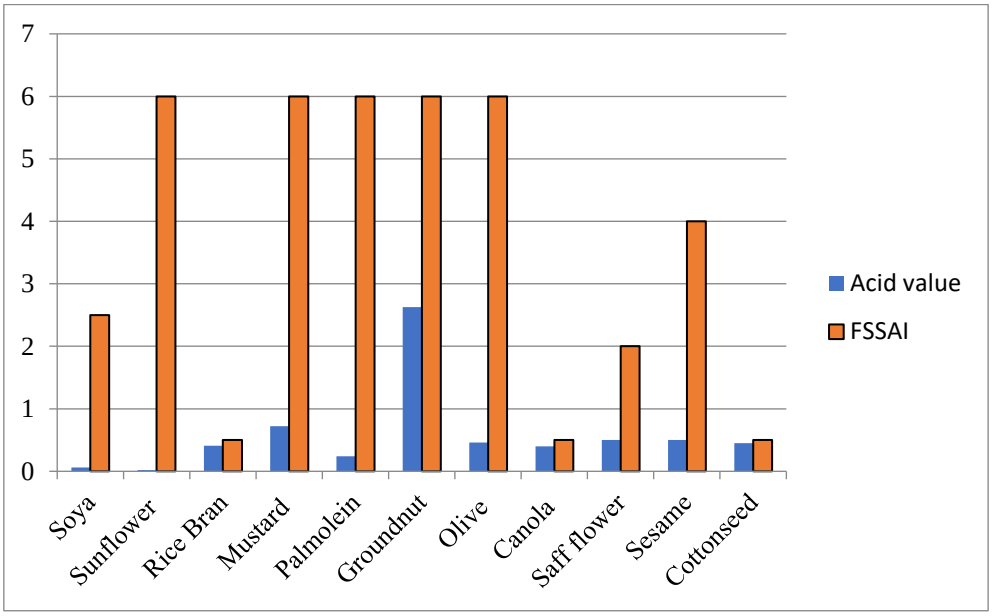


Figure 3.1 The measured Acid value of edible oils and FSSAI limits

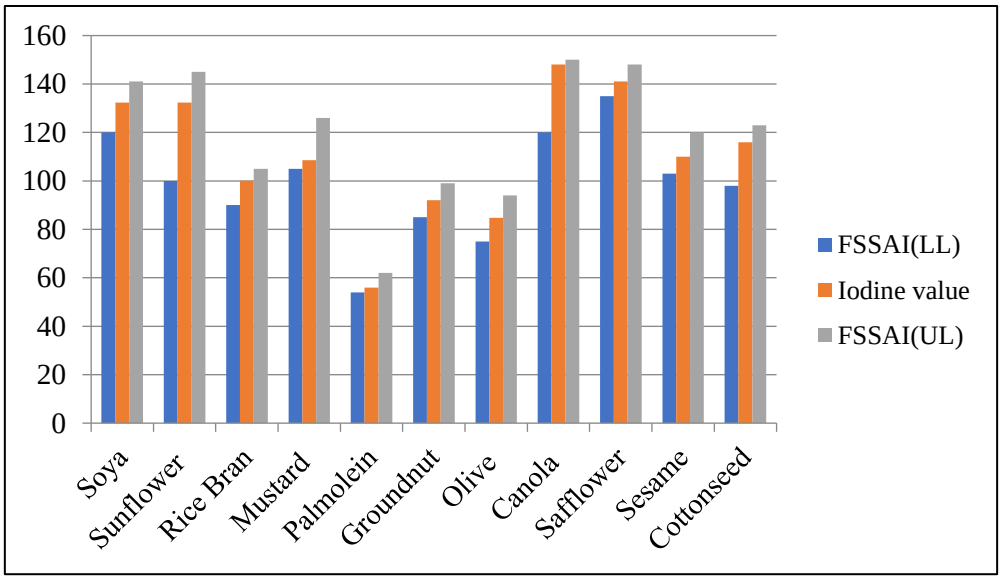


Figure 3.2 Measured Iodine value and upper and lower limits of FSSAI specification

According to Table 3.2, measured edible oil properties, and Table 3.3, FSSAI standard specifications for edible oil quality, all of the edible oil's physiochemical properties fall within the FSSAI-specified range. This demonstrates the authenticity of edible oils and no adulteration in them. Table 3.4 shows the Tocols (Tocopherols and Tocotrienols) content in a few of the edible oils used in the experiment.

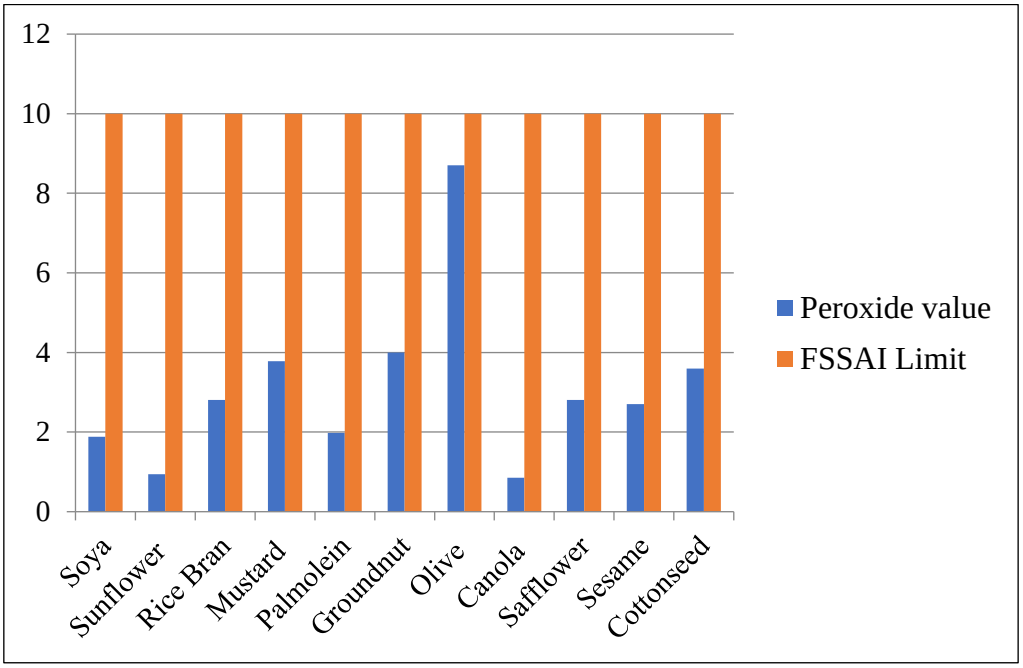


Figure 3.3 Measured peroxide value and FSSAI specifications for peroxide value

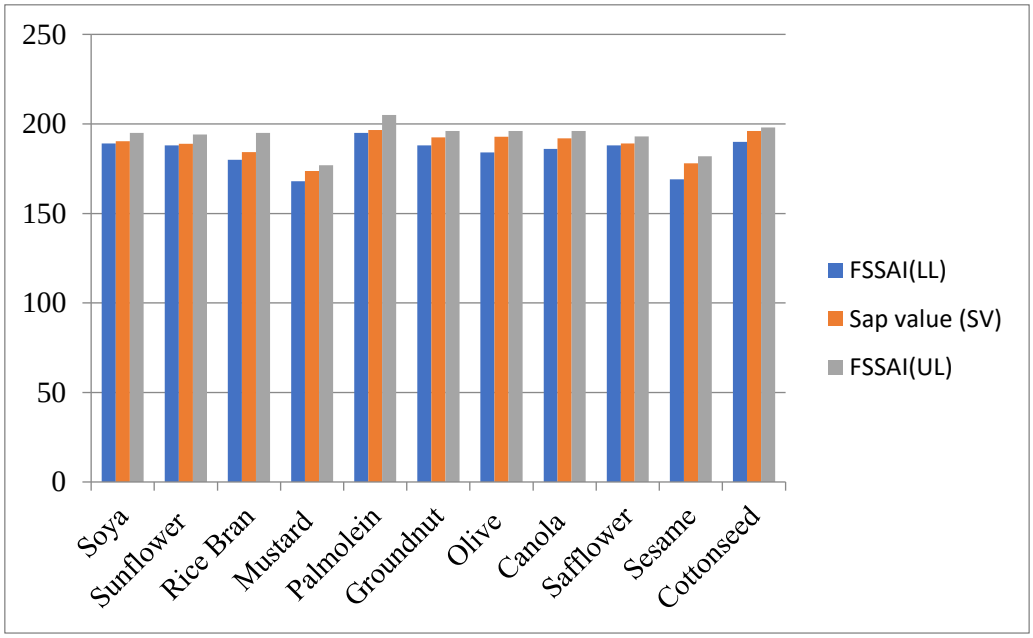


Figure 3.4 Measured saponification value to the FSSAI specifications

It is observed from Table 3.4, only rice bran and palm oil are found to contain tocotrienols, mainly γ-tocotrienols, and hence this property could be a good marker for checking adulteration of edible oils as both these oils are cheap compared to other oils. All other oils are rich in either α-tocopherol or γ-tocopherol. The acid profiles of edible oils measured using the AOCS methods are shown in Table 3.5. These fatty acid

profiles are also useful in the investigation of adulterations because they are unique to each edible oil. For adulteration detection, the measured fatty acid profiles are compared to the FSSAI ranges. The fatty acid composition of all studied edible oils met the specifications of FSSAI.

Table 3.4 Tocols (Tocopherols and Tocotrienols) content in the edible oils

Oil	**Tocopherols**			**Tocotrienols**			**Total Tocols (µg/mL)**
	α-T	**γ-T**	**δ-T**	**α-T3**	**γ-T3**	**δ-T3**	
Olive	131	7.3	-	-	-	-	138.3
Soya Bean	90	455.2	108	-	-	-	653.2
Sunflower	632	10.5	-	-	-	-	642.5
Rice Bran	44	42	-	12	164	-	262.0
Mustard	51	451.5	-	-	-	-	502.5
Palm	125	-	-	141	160.2	-	426.2
Groundnut	107	95	6	-	-	-	208.0

Table 3.5 Fatty acid profiles of edible oils under study

Fatty Acid	**Composition (in wt%)**								
	Rice bran	**Sun flower**	**Mustard**	**Soybean**	**Palm**	**Ground nut**	**Canola**	**Cottonseed**	**Olive**
C8:0	-	-	-	-	-	-	-	-	-
C10:0	-	-	-	-	-	-	-	-	-
C12:0	-	-	-	-	0.1	-	-	-	-
C14:0	0.3	-	0.1	-	0.9	-	-	0.9	-
C16:0	19.9	6.9	3.1	11.4	37.0	10.5	4.3	31.0	11.3
C16:1	0.4	0.2	0.4	0.1	0.3	-	0.2	0.8	0.9
C18:0	1.7	2.7	1.2	3.1	4.0	3.1	1.8	2.1	3.0
C 18:1	41.6	29.7	15.6	27.1	45.7	51.7	62.3	19.5	76.5
C18:2	33.5	59.8	19.8	52.6	10.9	27.2	19.4	60.2	6.9
C18:3	1.2	-	12.4	5.0	0.2	-	9.2	0.7	0.6
C20:0	0.6	0.2	0.7	0.3	0.4	1.5	0.6	0.5	0.4
C20:1	0.5	0.1	7.7	0.2	0.3	1.3	1.2	-	0.3
C20:2	-	-	1.0	-	-	-	0.1	-	-
C22:0	0.1	0.4	0.5	0.2	0.1	3.1	-	--	0.1
C22:1	-	-	35.1	-	-	0.1	-	-	-
C22:2	-	-	1.0	-	-	-	-	-	-
C24:0	0.2	-	0.3	-	0.1	1.4	-	-	-
C24:1	-	-	1.1	-	-	-	-	-	-

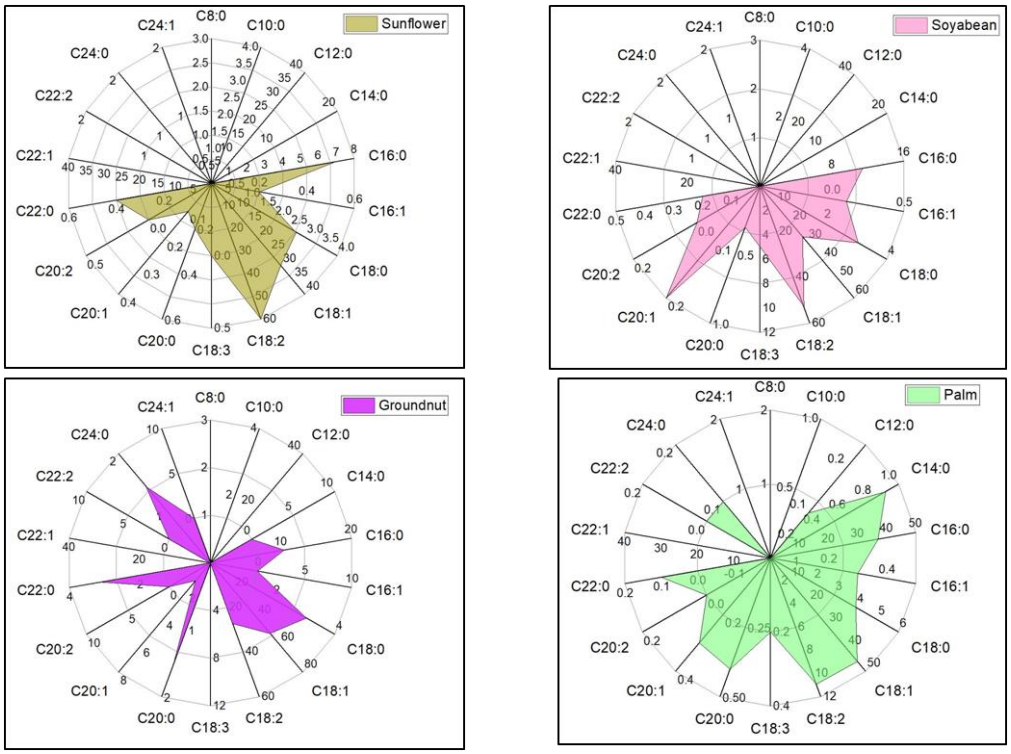


Figure 3.5 Chemical fingerprint of fatty acid profiles (Sunflower, Soya, Groundnut and Palm oils)

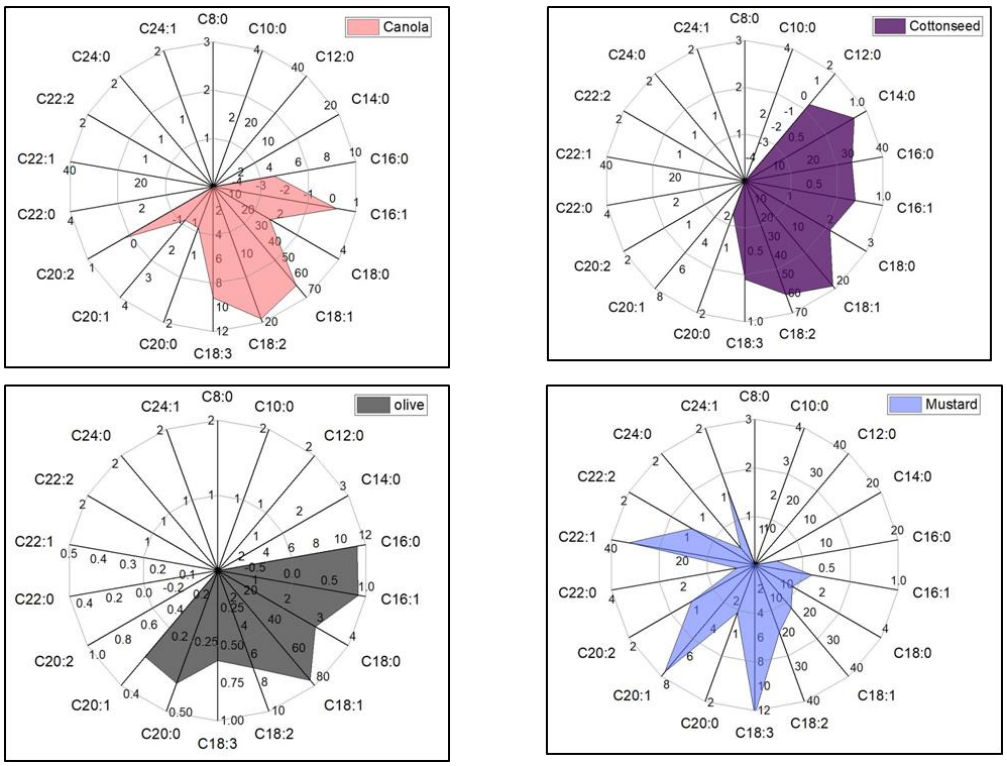


Figure 3.6 Chemical fingerprint of fatty acid profiles (canola, cottonseed, olive, and mustard oils)

In conclusion, the chemical analysis of edible oils for evaluating physicochemical properties revealed that all studied oils met FSSAI specifications. Ground nut oil had a high acid value, but it was still within the specification range. The fatty acid composition of all edible oils studied met FSSAI specifications as well. One mustard oil sample had a low erucic acid content (C22:1=35.1%), but it was still within the specification. These edible oils are used in analytical experiments like electronic tongue, spectroscopy which is described in the following section.

3.3. Electronic tongue

The development of electronic senses began to create devices that mimic the senses of smell, taste, and appearance. The functionality of human gustation (perception of taste) is an inspiration for the development of an analytical instrument electronic tongue (e-tongue), which reproduces the sensory appreciation of the taste in an artificial way. The first electronic tongue system mimicking the sense of taste by the transformation of taste information (sweetness, sour, umami, bitterness, etc.) to an electrical characteristic signature was presented in 1990 by Hayashi et al. [5]. An electronic tongue comprises an array of non-specific/semi-specific electrodes with partial specificity, high stability and cross-sensitivity electrodes, instrumentation for excitation and acquiring sensory response, and appropriate pattern recognition algorithms to process the signal for qualitative and quantitative analysis [6], [7].

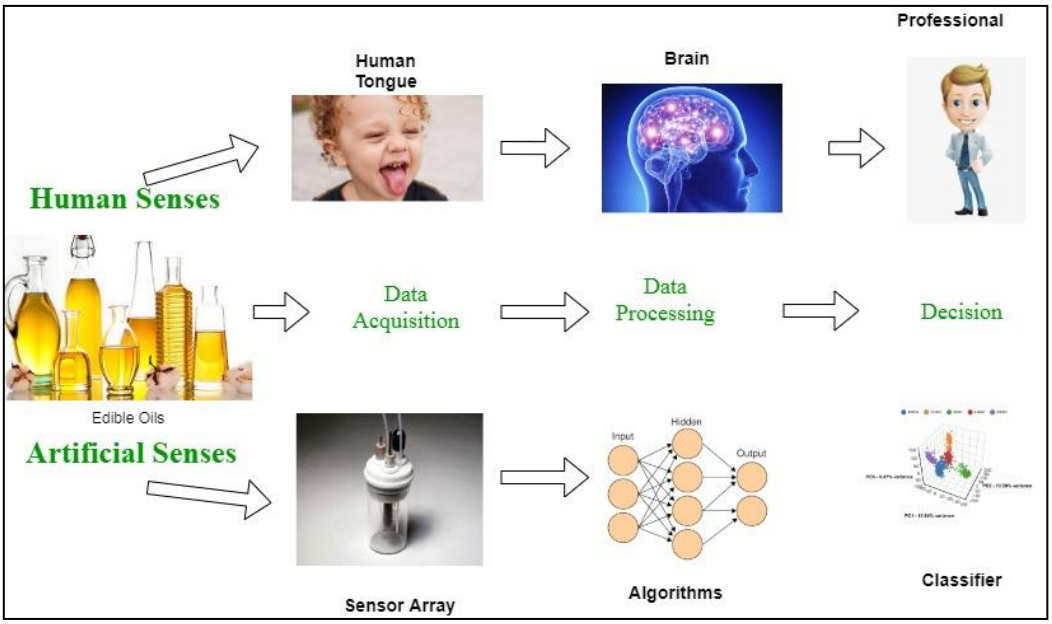


Figure 3.7 Pictorial analogy of electronic tongue to human taste receptor

Similar to the taste receptors in the human tongue, which perceive the taste and recognizes it by the recollection of gustation perception in the human brain, the cross selective electrodes in the electronic tongue function by measuring the overall pattern of a given sample and recognizing its nature from a database of previous learning [8]–[10]. Figure 3.7 depicts pictorial analogy of electronic tongue to human taste receptor. Similarly, the human olfaction system was the inspiration for the development of an electronic system for distinguishing different odors called an electronic nose (e-nose). An electronic nose comprises an array of non-specific/semi-specific gas sensors coupled to a pattern recognition system. Similar to the human nose (recognizing the aroma of a sample), the electronic nose functions by measuring the overall pattern of the given sample and comparing it with the previous learning stored in a database [11].

3.3.1. Electronic tongue sensing principles

There are different sensing principles such as electrochemical, optical, measurements of mass change for qualitative and quantitative analysis to be used in the development of electronic tongue [8], [9]. Among these principles, electrochemical sensing is the most widely used sensing principle. In electrochemical sensing or measurement, potentiometric, voltammetric, and impedance spectroscopy methods are often used for investigating samples in the liquid phase. The following sections explain the above-mentioned sensing principles.

3.3.1.1. Potentiometric Sensing

The potentiometric measurement system contains two electrodes configuration - one working electrode and another reference electrode [13], [14]. Potentiometric sensor arrays may be used for the classification and analysis of pharmaceutical and food samples [15]–[17]. An Ion-Sensitive Field-Effect Transistors (ISFETs) (pH electrode), Ion-Selective Electrodes (ISE) for calcium, potassium, sodium, chloride ions, etc. are the best examples of potentiometric sensors [18]. The working electrode interacts with the target molecule in an electrolyte solution in the electrochemical cell, while the reference electrode generates constant potential. The potential difference between the working electrode and reference electrode at equilibrium state is the signal response of liquid under analysis using potentiometric sensing. This response is characteristic of the liquid under test. There is a wide scope of research in the development of different types of membrane materials having different recognition capabilities that may be used to develop potentiometric e-tongue [16], [19]. The main drawbacks of potentiometric measurements are their temperature dependency and the adsorption of solution

components (ions) by the electrodes (affects the membrane potential) [20]. Applications of potentiometric e-tongue include quality control in the food industry, taste assessment, discrimination of liquid food (e.g., beer, honey, and teas) of different brands and types, and classification of oils [19], [21]. It has applications in environmental and industrial analysis, also like monitoring water contamination.

3.3.1.2. Voltammetric Sensing

The voltammetric sensing technique consists of a three electrodes configuration: 1). working electrode 2). counter or auxiliary electrode and 3). reference electrode. Electronic tongue based on voltammetry was first developed by Winquist and his team in 1997 [22]. In this method, the exciting potential is applied to a working electrode while the resulting current due to the redox reaction in the electrochemical cell is measured utilizing a potentiostat circuit [23]. Working electrodes are made up of metals or alloys composed of copper, nickel, palladium, silver, tin, titanium, zirconium, gold, platinum, and rhodium. The multi-sensor array is placed in the solution to be investigated in an electrochemical cell. A specific potential is imposed on the working electrode, and the redox current is measured to analyze the features qualitatively and quantitatively. The size and shape of applied potential determine whether the target molecules shall loose or gain electrons. This technique is well suited for the measurement of chemical compounds where oxidation-reduction reactions occur.

E-tongue based on voltammetry makes use of non-specific metal electrodes for acquiring the information related to the redox-active species through the measurement of the current through these electrodes when a potential having a specific shape is applied. Voltammetric sensing has advantages such as simplicity, robustness, and a wide range of analytical possibilities. It has applications in the dairy, oil, and fat, pulp, and paper industry [24], [25]. Voltammetry methods vary depending on the shape and size of the applied waveform, such as Cyclic Voltammetry (CV), Normal Pulse Voltammetry (NPV), and Differential Pulse Voltammetry (DPV)[26]. The voltammetric electronic tongue, which involves a potential sweep, produces a charge transfer within the electrochemical cell because of redox reactions at the surface of the electrode, as shown in Figure 3.8. The resulting current because of redox reactions is proportional to the concentration of electroactive species.

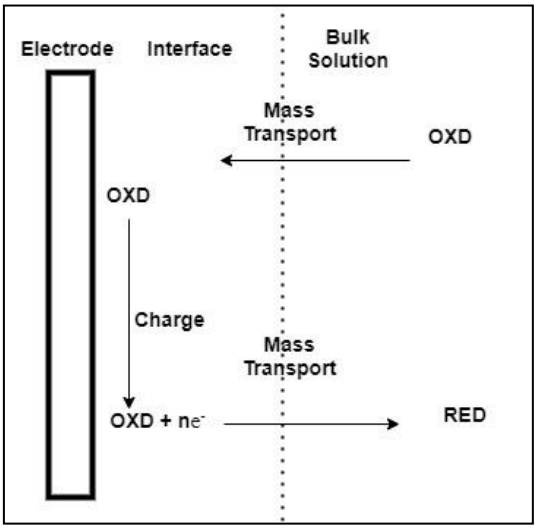


Figure 3.8 Electrochemical reaction events

3.3.1.3. Cyclic voltammetry (CV)

Cyclic Voltammetry (CV) is the most used excitation potential sweep techniques. In Linear Sweep Voltammetry (LSV), the excitation potential on the working electrode is based on a linear potential waveform, where the voltage E is changed linearly with respect to time (t). The rate in potential change is known as the scan rate $(v = dE/dt)$, which varies between $1\ mV/s$ and $1\ V/s$. Figure 3.9 shows a linear sweep voltage waveform. Figure 3.10 depicts the corresponding current response. In LSV, the applied potential region is varied from the initial potential (E_{init}) to the final potential (E_{final}).

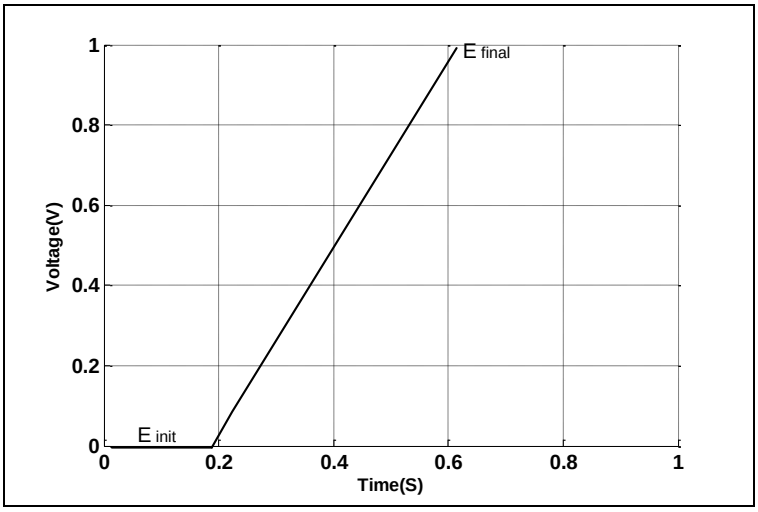


Figure 3.9 Linear sweep voltammetry excitation voltage

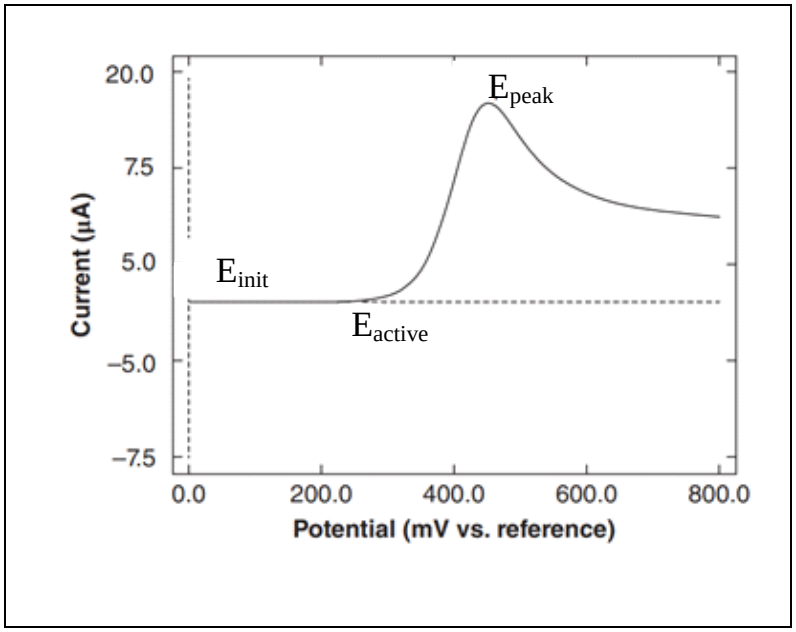


Figure 3.10 Current response of Linear sweep voltammetry

In the case of Cyclic voltammetry, the direction of the potential scan is reversed at the end of the first scan switching potential, E_{sw1}.In some experiments, the application of potential stopped at E_{final}, or the voltage is scanned till the second switching potential value (E_{sw2}), where the direction of the potential scan is reversed till E_{final} as shown in Figure 3.11. The current response as a function of the applied voltage is represented as a voltammogram as shown in Figure 3.12.

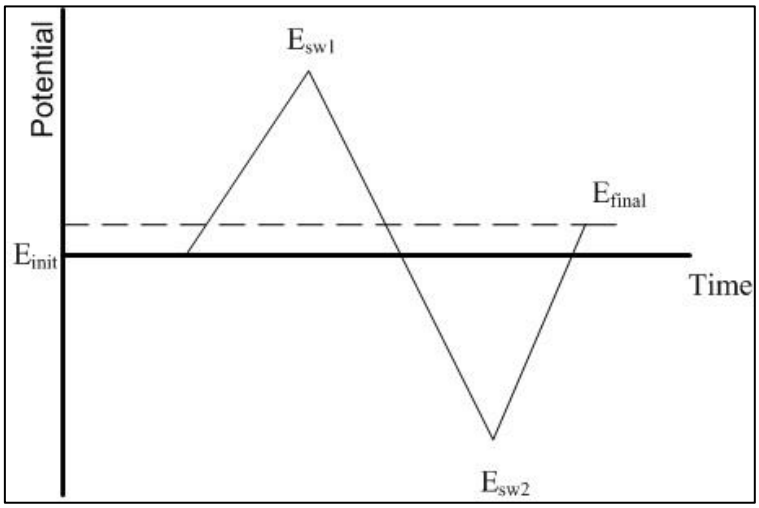


Figure 3.11 Cyclic voltammetry sweep signal

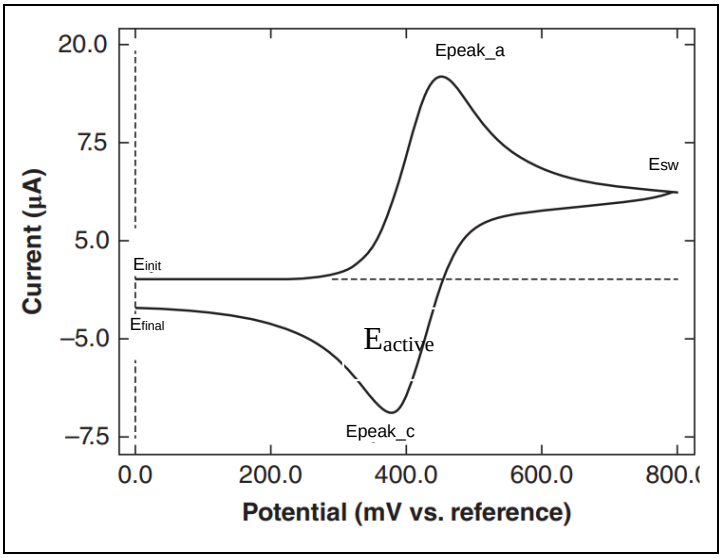


Figure 3.12 Voltammogram of cyclic voltammetry

The current response, $\mathbf{I_{total}}$, in voltammetry experiments corresponds to the addition of charging ($\mathbf{I_c}$) and faradaic ($\mathbf{I_f}$) current contributions. The cumulative current because of the applied potential sweep is measured as $\mathbf{I_{total}}$ is addition of

- Faradaic current (I_f) due to the redox reactions that occur at the working electrode due to the applied potential. This is due to the charge transfer across the electrode – solution interface when potential is applied.
- Charging current (non-faradaic current) (I_C) resulting due to the processes such as adsorption, desorption wherein the structure of the electrode-solution interface changes with changing potential or solution composition. This is the external current that flows transiently when the potential, electrode area, or the solution composition changes

$$I_{total} = I_C + I_f = vC_d + I_f \tag{3.6}$$

$$I_f = \frac{nFA\sqrt{D}C}{\sqrt{\pi t}} \tag{3.7}$$

Where A (cm^2), is the area of the working electrode, D is the diffusion coefficient (cm^2/sec) and $C(mol/cm^3)$ is the concentration of analyte. The most common response of a CV experiment is a peak-shaped curve (Figure 3.12) from the activation potential (E_{active}) at which the electrochemical reaction takes place. As the potential moves from E_{active}, continuous depletion of electroactive species near the electrode surface occurs, reaching the peak potential (E_{peak}), in which the electrochemically

reactive species has been completely transformed. Beyond this potential value, mass transport is responsible for current in the electrochemical reaction. Cyclic voltammetry is used to study qualitative information on the presence of intermediates in oxidation-reduction reactions, the reversibility of a redox reaction, determination of diffusion coefficient, and reduction potential of an analyte. Besides, an unknown solution concentration can be determined with a calibration curve between current and concentration.

3.2.1.4. Impedance spectroscopy

The first electronic tongue based on electrochemical impedance spectroscopy (EIS) was developed by Riul et al. [27]. EIS is an electrochemical technique commonly used to characterize electrochemical cells which can give accurate and error-free kinetics of multiple electrochemical processes. EIS is now being used to study batteries, biosensors, fuel cells, physical electrochemistry, and semiconductors [28]. EIS is a powerful analytical method utilizing small amplitude, alternating voltage (AC) signals as excitation signals on working electrodes. The resultant current because of redox reactions on the surface of the electrode is measured. The current output is analyzed as a sum of the sinusoidal waveform using Fast Fourier Transform (FFT). The voltage and current response are used to probe the impedance characteristics of a cell. This technique measures the impedance of a system over a range of frequencies (1 mHz - 30kHz) to generate an impedance spectrum for the electrochemical cell under test. This spectrum is characteristic of constituents of the sample. Figure 3.13 shows an instance of applied potential and current response for a linear or pseudo linear system.

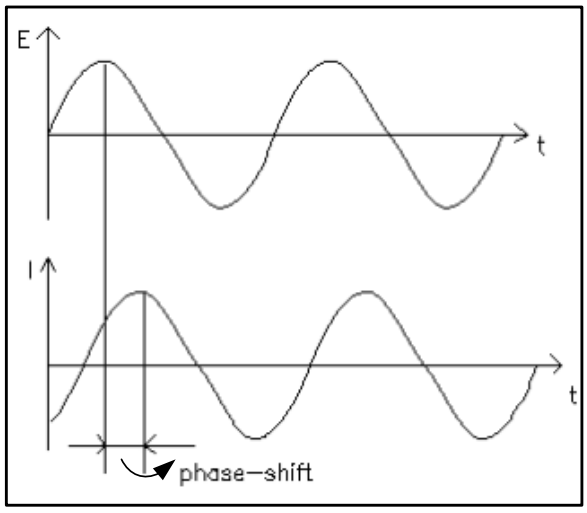


Figure 3.13 Voltage and current responses in a linear EIS

Aside from the sensing principles discussed above, there are optical sensing based electronic tongues that have high repeatability and long-term stability [29] and mass sensitive based electronic tongues[29], [30], [31]. An electronic tongue based on voltammetric, and electrochemical impedance spectroscopy is investigated for this research work for edible oil analysis. The sections that follow describe the experimental methodology for the same.

3.3.2. Electronic tongue -experimental methodology

The emphasis in this thesis has been on developing an electronic tongue based on cyclic voltammetry (CV) and electrochemical impedance spectroscopy (EIS). Experimental data of CV and EIS has been used for further data analysis using AI-based pattern recognition techniques covered in subsequent chapters.

3.3.2.1. Experimental setup

The experimental setup for CV and EIS is similar, and the block diagram is shown in Figure 3.14. The setup consists of an array of working electrodes spaced at regular intervals from a reference electrode and a counter electrode. To reduce the influence of uncompensated solution resistance effect on redox current flow, the reference electrode was kept close to the working electrode. Voltammetry employed three working electrodes: platinum (Pt), copper (Cu), and nickel (Ni) (Ni). The reference electrode was an Ag/AgCl electrode, whereas the auxiliary or counter electrode was stainless steel. For the EIS experiment, platinum (Pt) and gold (Au) electrodes were used as working electrodes. As shown in Figure 3.14, a Teflon material cylindrical structure was developed to accommodate the working and reference electrodes, which were enclosed by a stainless-steel counter electrode. The sample was kept in a glass electrochemical cell. All these electrodes are connected to a typical potentiostat in a three-electrode configuration.

A potentiostat, as illustrated in Figure 3.15, is an electronic circuit used to apply a predefined shape and size potential to the working electrodes and acquire current flowing between counter and working electrodes in an electrochemical cell because of redox reactions. Every time a measurement is taken, a relay module (Figure 3.16, developed in the lab of the Digital Systems Group, Central Electronics Engineering Research Institute, Pilani, India) for sequentially switching the electrodes was used to connect each of the working electrodes to the potentiostat circuitry to form a three-electrode configuration. Figure 3.17 depicts the experimental lab setup.

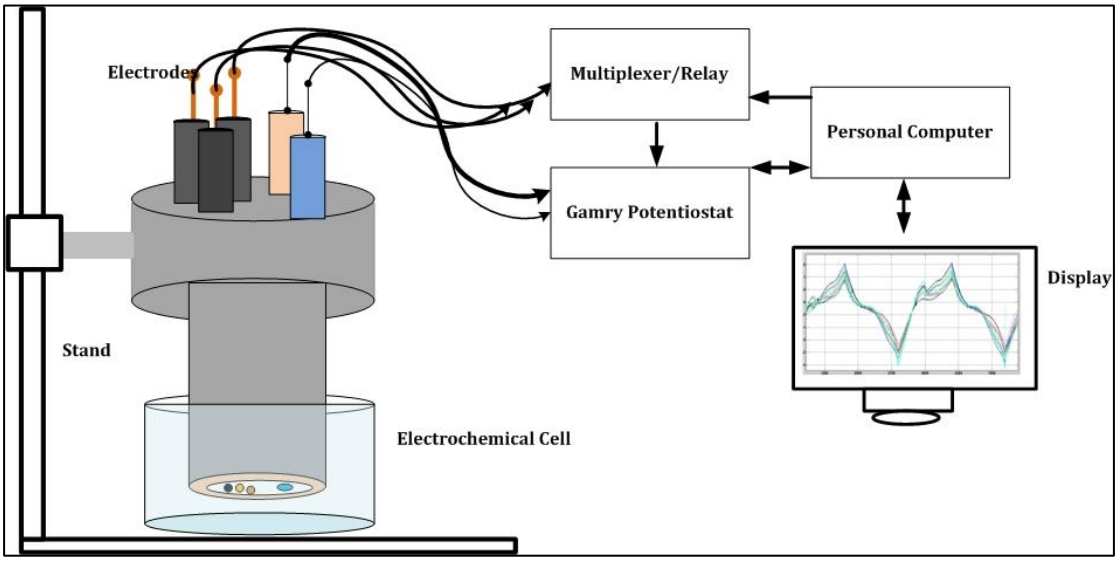


Figure 3.14 Experimental setup for electronic tongue setup

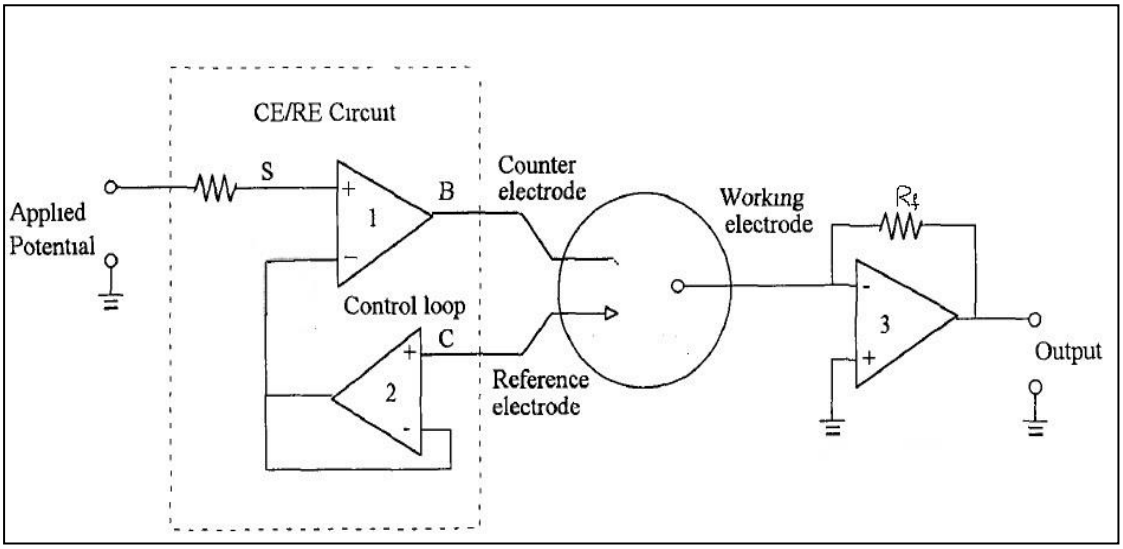


Figure 3.15 Generic potentiostat circuit with a three-electrode configuration

All the above-mentioned working electrodes are of diameter 2.0 mm and 99% purity, manufactured by CH instruments, USA and eDAQ, Australia. Gamry Instruments potentiostat (Series G-300),USA used was interfaced to a personal computer. The voltammetric signal of interest is then imposed onto each of the working electrodes in sequence via a potentiostat and the generated redox current responses, which are indicative of the redox species present in the solution, is acquired and sent to the PC via potentiostat for further signal processing and analysis [32].

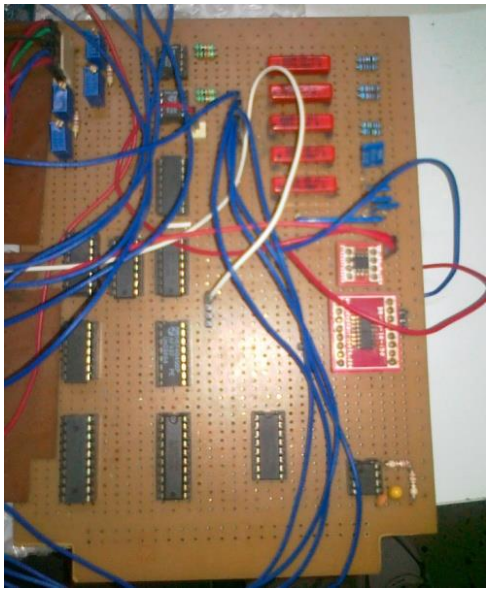


Figure 3.16 Relay circuitry for switching the working electrodes

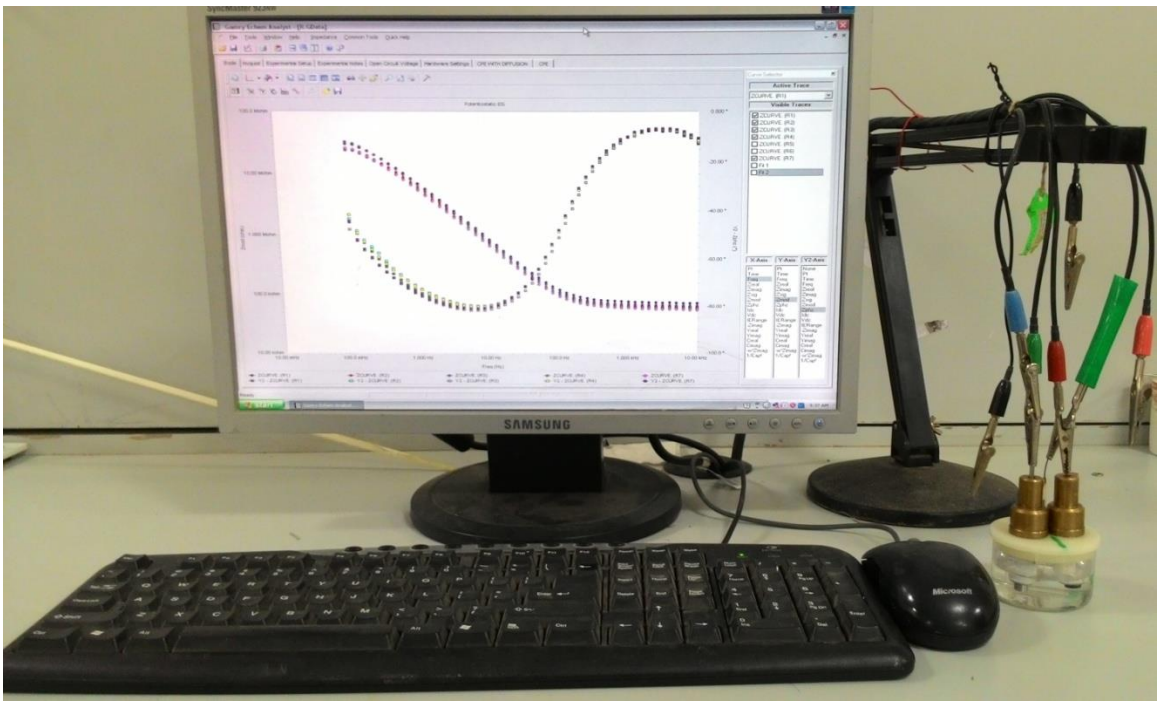


Figure 3.17 Experimental setup for electronic tongue

3.3.2.2. Edible oil samples used for the e-tongue experiment

In the voltammetry experiment for edible oil classification, edible oil samples such as groundnut oil (GNUT), canola oil (CAN), mustard oil (MUS), olive oil (OL), safflower oil (SAFF), soya oil (SOYA), sunflower oil (SUN), palm oil (PALM), and sesame oil (SES) were utilized.

3.3.2.3. Sample Preparation for Voltammetry and EIS experiment

Edible oil samples cannot be used directly in the electronic tongue experiment since they are non-conductive liquids. To make them compatible for the research, petroleum ether, a laboratory solvent, is employed for sample preparation. In the first step, a magnetic stirrer is used to thoroughly mix 40 ml of an edible oil sample with 40 ml of distilled water in a conical flask. The above mixture (oil and distilled water) is then thoroughly stirred for 3 minutes using a magnetic stirrer to ensure consistent mixing. The prepared sample is placed into a separating funnel and allowed to settle for 15 minutes. Two static layers of oil and emulsion will develop in the funnel [31]. The emulsion layer is carefully separated from the separating funnel. The sample solutions are kept in a dark, airtight container. A similar process is adopted for all edible oils taken for analysis.

3.3.2.4. Voltammetry- Measurement Procedure

Three working electrodes made of Platinum (Pt), Copper (Cu) and Nickel (Ni) from eDAQ (Australia) are used in a cyclic voltammetry-based electronic tongue. A current to voltage (I to V) converter circuit is used to measure the redox current between the working electrode and the counter electrode with respect to the reference electrode.

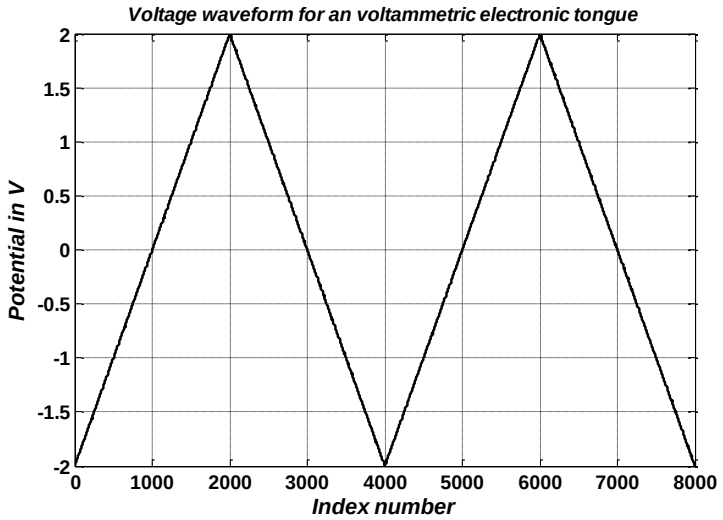


Figure 3.18 Voltage waveform used in voltammetry experiments

In this cyclic voltammetry experiment, a potential sweep is applied between the working reference electrode. Figure 3.18 depicts the applied potential waveform. At the start of the experiment, the working electrode is held at a base potential E_{init} (-2V) and

after a fixed waiting period, the potential is swept linearly to a switching potential (E_{sw1}) of 2V. Then sweep is reversed till it reaches a second switching potential E_{sw2} of -2V. Finally, forward sweep until the E_{final} (+2V) A current will then flow to the electrode, initially sharp when a Helmholtz double layer of charged species is formed and electroactive compounds next to the electrode surface are oxidized or reduced.

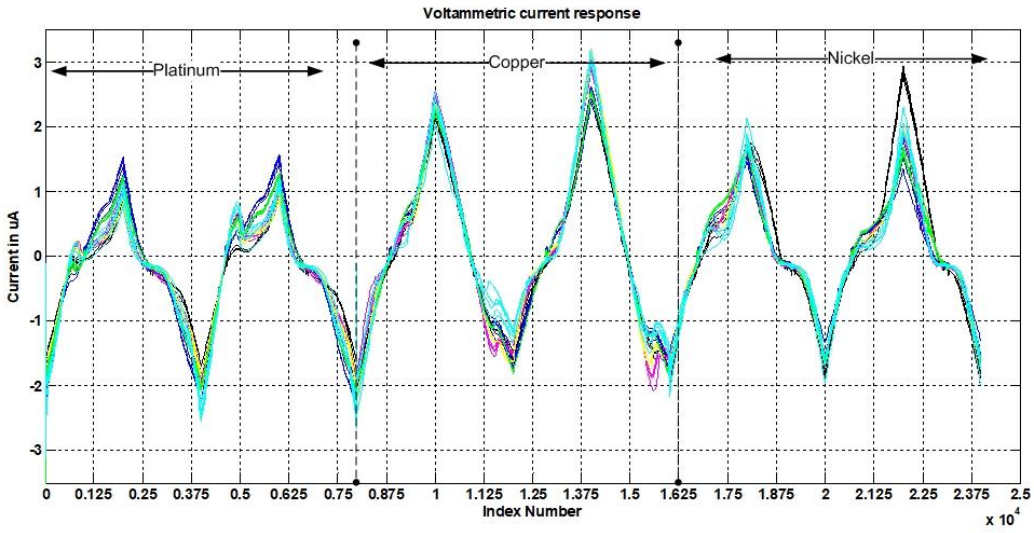


Figure 3.19 Voltammetry current response of e tongue system

The current will then decay as the double layer capacitance is charged and electroactive compounds are consumed until only the diffusion-limited faradic current remains. When the potential of the electrode is stepped back to its initial value, similar but opposite reactions occur. The size and shape of the transient current responses reflect the amount and diffusion coefficients of both redox-active and charged compounds in the solution. Figure 3.19 depicts the current response from voltammetry experiment using three working electrodes.

Each oil sample was analyzed five times in a row, and the data from the five measurements were averaged to yield a single measurement. Each edible oil was subjected to eight such measurements. For each electrode, each measurement cycle generated an array of data points with a size of 1x8000 (i.e., acquired current response data points while applying the measurement pulse sequence). The experiments with a single electrode for eight edible oils (eight measurements each) generate a 64x8000 data matrix. With three electrodes, the total data matrix is 64x24000. Chapter 4 describes the signal processing of the electronic tongue data.

3.3.2.5. EIS: Measurement Procedure

A standard three-electrode configuration is used for electrochemical impedance spectroscopy (EIS). It is composed of three working electrodes (99.9% pure 65 mm length and 2mm diameter) of Platinum (Pt), Gold (Au), and Glassy carbon (GC), an Ag/AgCl reference electrode, and a stainless-steel counter electrode. All electrodes were manufactured by EDAQ (Australia), and the potentiostat EIS experiment was carried out with a commercial benchtop potentiostat from Gamry Instruments (Series G300), USA. The experiment was carried out by switching three working electrodes one at a time using a relay circuit to create a three-electrode configuration, as shown in Figure 3.20.

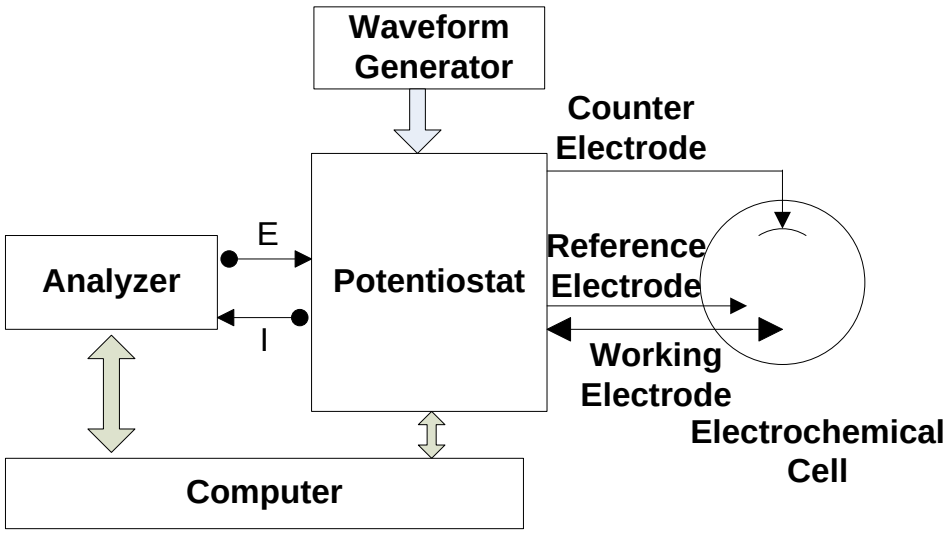


Figure 3.20 Block diagram of EIS experiment

A low amplitude sinusoidal AC voltage of $50mV_{PP}$ was applied to the working electrode, with a frequency sweep of 0.1mHz to 30 kHz. The frequency sweep is set to 10 points per decade. The redox current between the counter and the working electrode was measured and recorded at each applied potential. The working electrode was initially held at zero potential. The impedance of the electrochemical cell is calculated using the applied voltage and acquired current. This experiment is repeated for each working electrode using the same frequency sweep and sinusoidal signal. The magnitude and phase difference between the applied voltage and the acquired current signal are calculated using FFT, as shown in Figure 3.21.

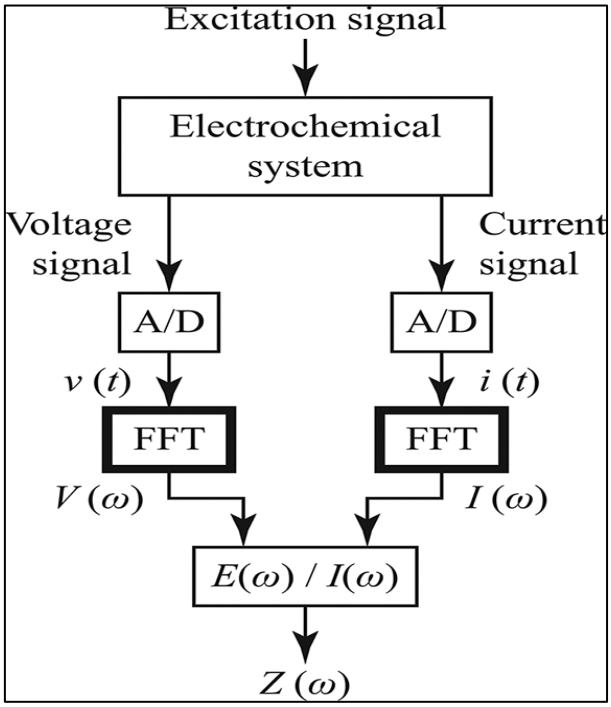


Figure 3.21 EIS measurement of impedance

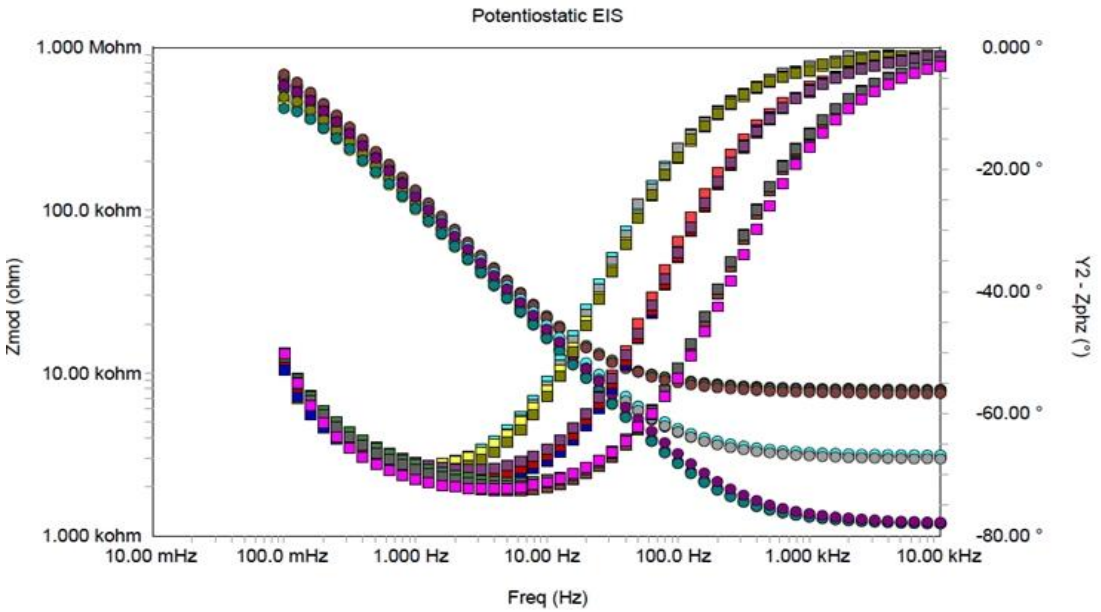


Figure 3.22 An instance of EIS magnitude and phase measurement

Figure 3.22 depicts an instance of EIS measurement. Three edible oil sample magnitude and phase plots are easily recognizable from the plot. The magnitude and phase data are used for additional data analysis (described in Chapter 4).

3.4. Spectroscopy-analysis of edible oils

Spectroscopy is a widely used analytical tool for edible oil analysis. Spectroscopy is defined as the study of the quantized interaction of electromagnetic radiations with the matter. The interaction of electromagnetic radiation with a medium is described by the electromagnetic theory comprising maxwell's equations. This interaction is of two types either absorption or emission. The properties of electromagnetic radiation are described in terms of frequency, wavelength, and amplitude. The energy of a photon is given by $E = hv$

The relationship between the absorption (A) of light through a substance and properties of that substances is given by Beer-Lambert law, also known as Beer's law, formulated by German mathematician and chemist August Beer in 1852.

$$A = -\log_{10}\frac{I}{I_0} = \mathcal{E}cl \tag{3.8}$$

$$I = I_0 e^{-\mathcal{E}cl} \tag{3.9}$$

l is the path length (cm) of the absorbing medium, $\mathcal{E}$ is molar absorptivity coefficient ($mol^{-1}cm^{-1}$), c is the concentration ($mol\,L^{-1}$) of the sample. $\mathcal{E}$ depends on the nature of chemical and wavelength of light used. The absorption of a specific wavelength of electromagnetic radiation by an atom or molecule causes to gain energy from a photon making the atoms of the molecule go from a lower energy state to a more excited state. The absorbed wavelength will be missing in the measured spectra, as shown below Figure 3.23 [33]. This is the basic principle of absorption spectroscopy.

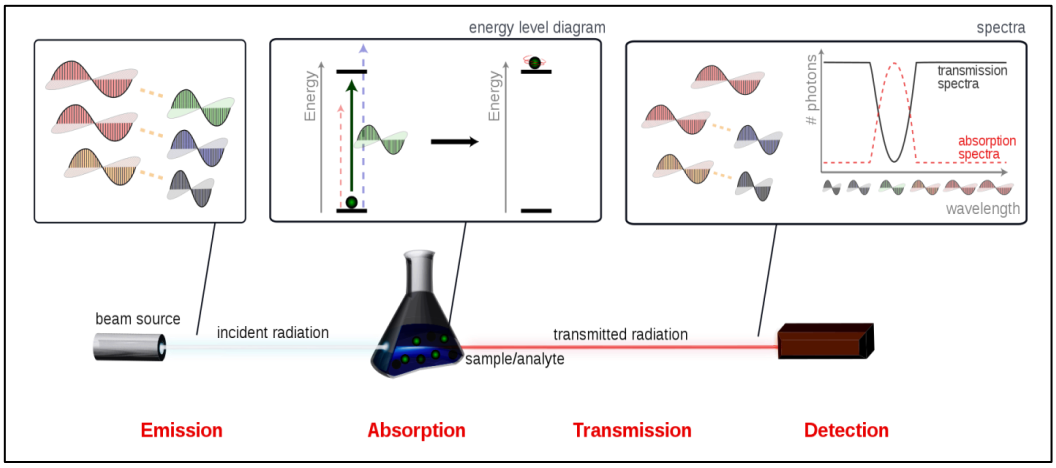


Figure 3.23 Working principle of absorption spectroscopy

Spectroscopy is a non-destructive, rapid technique applied for assessment of quality of food. Food products with a chemical composition exposed to a light source produce a characteristic signature called a spectrum. Spectroscopic methods measure radiation intensity as a function of wavelength to analyze the inherent properties of a sample. This spectrum of sample under investigation is compared to the spectrum of the standard sample to check authenticity. Depending upon the electromagnetic radiation frequency range and principle of operations there are different spectroscopic techniques for the analysis of edible items.

The interaction of the incident light beam with the sample is dependent on the physical properties of samples, such as particle size and refractive index, etc. which change the direction of the incident radiation path. When the radiation or beam touches a surface, either it is reflected as specular reflectance or enter the sample and give diffuse reflection (uniform reflection at all angles). This scattered light can be partially absorbed within the sample or reflected before it exits. The scattered light has interacted with the sample and therefore has chemical information [34]. Depending upon the interactions with medium there are different types of spectroscopy techniques: Diffuse Reflectance Spectroscopy, Diffuse Transmission Spectroscopy, Transflection Spectroscopy, Attenuated Total Reflection Spectroscopy, Transmission, or Absorption Spectroscopy. Figure 3.24 shows the visualization of some of the above-mentioned spectroscopy methods.

Samples that reflect the light can be measured by Diffuse Reflectance and Diffuse Transmission. Liquid samples that do not reflect the light can be measured by transmission or transfection. NIR spectroscopy uses about 0.5 mm to 3 cm path lengths and is therefore far more sensitive to particle size. On the other hand, the use of Attenuated Total Reflection (ATR) sampling with the MIR is normally performed on samples with μm path-lengths because of high absorbance in this region.

In this study, we have used Infrared Spectroscopy (NIR and MIR) techniques for edible oil analysis and adulteration detection. Details of Infrared spectroscopy and experimental procedure for edible oil analysis with NIR and MIR spectroscopy are covered in subsequent sections.

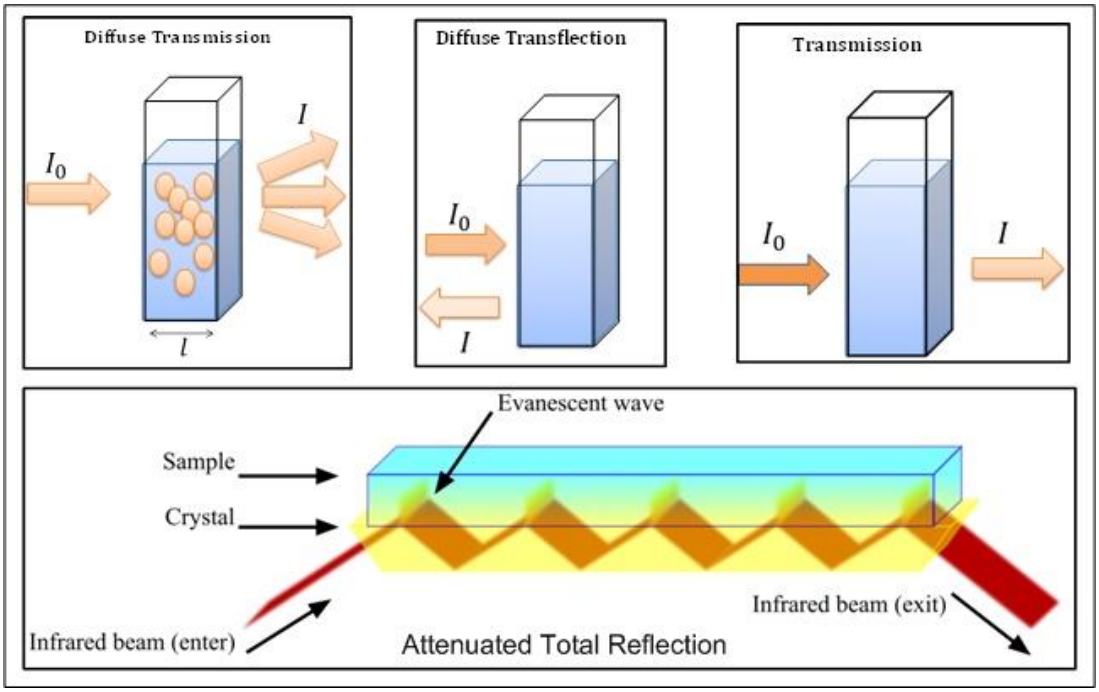


Figure 3.24 Types of spectroscopy principles

3.5. Infrared spectroscopy

The infrared (IR) region in the electromagnetic spectrum was discovered by F.W.Herschel in 1800. This IR radiation extends from 760nm to 1mm (430THz - 300GHz). This IR region is further divided into three regions, Near Infrared(NIR:780nm -2.5µm), Mid-infrared (MIR:2.5µm -50µm) Far-infrared (FIR:50µm-1mm) [35].

IR spectroscopy is a type of absorption spectroscopy, which means that absorption of a particular wavelength corresponds to specific vibrational and rotational modes, such as bond bending, stretching, and wagging. It is used to identify the functional groups in a compound. A functional group is a collection of atoms or molecules with a common bonding pattern. Each functional group bends, stretches, and wags at different frequencies. The group reacts to radiation in a typical way independent of other molecules. When electromagnetic radiation in the infrared region interacts with the samples, a functional group absorbs the frequency of light that corresponds to the frequency of its stretching, bending, or wagging. The absorbed frequency bands can be seen in the IR spectra obtained at the detector. These absorption bands are characteristics of a specific functional group in the molecule. Near-infrared (NIR) spectroscopy is based on molecular overtone and combination vibrations and operates

in the near-infrared region of the electromagnetic spectrum (from 800 to 2500 nm) [36], [37].

3.5.1. Near-Infrared spectroscopy for the analysis of edible oils

The basic idea behind NIR spectroscopy is to irradiate a sample with a NIR light source and then record the transmitted radiation. NIR spectroscopy has proven useful in compositional, functional, and sensory analysis of food [8], [9], [38].

In NIR spectroscopy, no sample preparation is required, so analysis of the sample is very simple and relatively fast, with the added benefit of online analysis. One of the advantages of NIR technology is that it allows multiple constituents to be measured concurrently. NIR spectroscopy requires the use of a cuvette to hold the samples under test. Because edible oils are sticky and viscous liquids, cleaning the cuvette is a significant issue with this spectroscopy method. The double beam NIR absorption spectroscopy is depicted in Figure 3.25.

3.5.2. Experimental setup for NIR spectroscopy

A double beam spectrophotometer UV3600 from SHIMADZU (Kyoto, Japan) was used for NIR absorption spectroscopy for edible oil analysis. The spectrophotometer instrumentation consists of three different detectors (photomultiplier tube for ultraviolet and InGaAs for visible regions, and cooled PbS detector for the near-infrared region), as well as a double monochromator to achieve high resolution (maximum 0.1 nm) in the large wavelength range of 185-3000 nm. TCC-240 accessory from SHIMADZU was used for temperature-controlled measurement. UVProbe (version 2.34) software from SHIMADZU was used for instrument control and spectral collection. Figure 3.26 depicts the experimental setup for NIR spectroscopy.

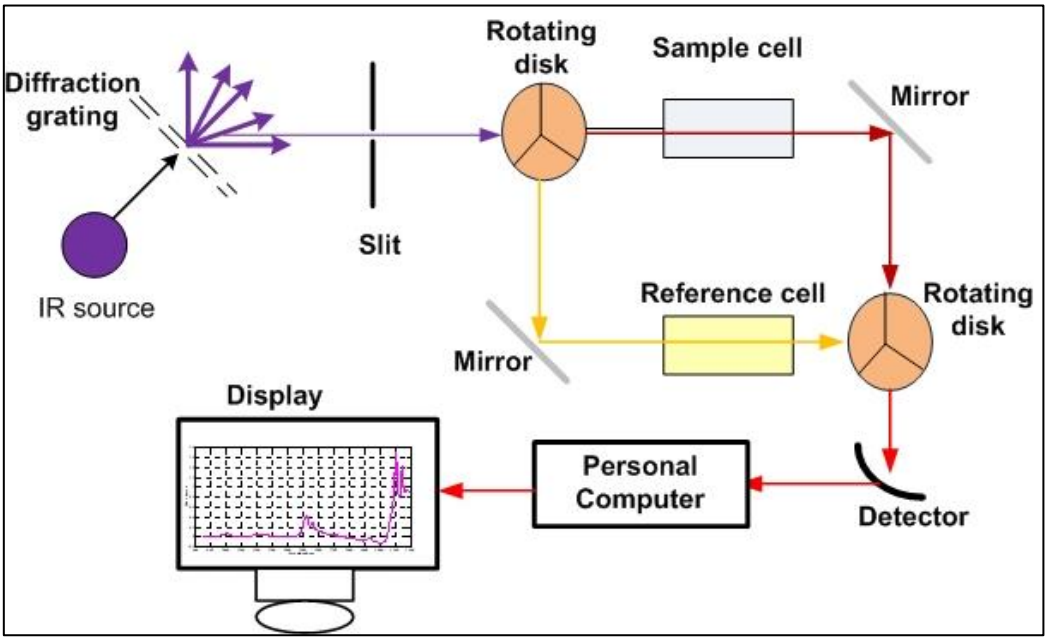


Figure 3.25 Block Diagram of NIR spectrometer.

Figure 3.26 Experimental setup for NIR spectroscopy

3.*5.3. Experimental procedure*

Five types of edible oil samples were used in the NIR spectroscopy experiment. Each edible oil sample was transferred to a 0.35µL quartz cuvette with a 1mm path length. Before placing it in the temperature-controlled chamber TCC240, it was inspected for any bubble formation. The absorption spectra of each sample were collected in the wavelength range of 1095 nm to 2400 nm with a resolution of 1nm and a slit width of

5nm. Sample's spectra were recorded at 30°C, and the time for spectra collection was 6 minutes per spectra. Each sample was measured three times before being averaged as a single measurement for further analysis. Figure 3.27 depicts edible oil spectra in the 1095-2400 nm range. A total of 28 spectra measurements were collected for each sample, providing 28x1351 data. The total data with five different types of edible oils was 140x1351. Chapter 4 presents data analysis of NIR spectra for the classification.

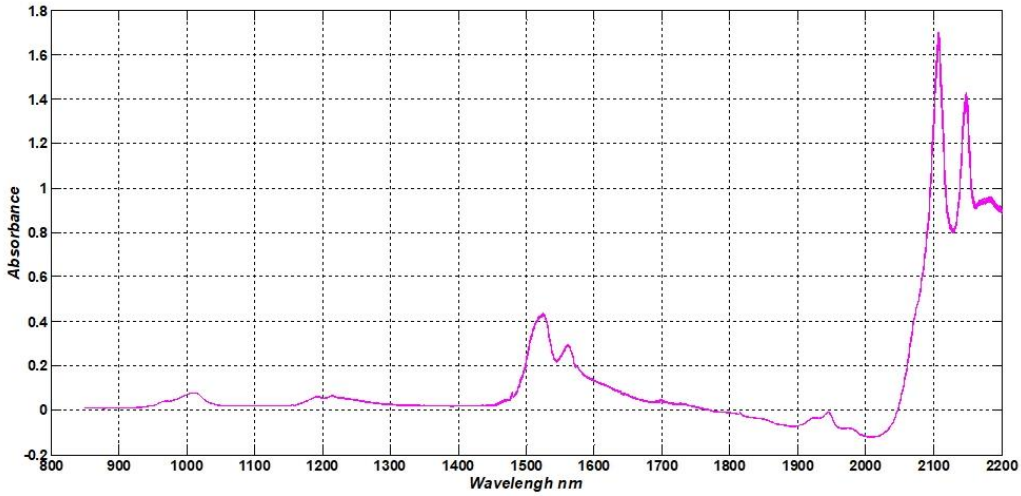


Figure 3.27 NIR spectra of edible oils

3.6. Mid infrared spectroscopy for analysis of edible oils

Attenuated Total Reflection (ATR) sampling technique is a versatile and powerful optical sensing technique that is used in conjunction with infrared spectroscopy in the mid-infrared (5μm-50μm) region, allowing samples to be examined directly in the solid or liquid state. The ATR sampling technique does not necessarily require any sample preparation. ATR sampling is a comparatively straightforward technique, and the equipment required is inexpensive, making it even more attractive.

3.6.1. MIR spectroscopy with ATR sampling

ATR sampling technique is based on the total internal reflection principle in MIR region, which occurs at the boundaries of two mediums when incident light completely reflects within the medium. This phenomenon is observed when light travels from a medium with a higher refractive index to a medium with a lower refractive index and enters the second medium at an angle greater than a critical angle.

An Infrared beam propagating in an optically dense medium with refractive index μ_1 undergoes total reflection at the interface of an optically rare medium with refractive

index μ_2, when the angle of incidence exceeds the critical angle θ_c. This IR beam propagates through the optical interface and generates an evanescent field in the second medium. The electric field amplitude of the evanescent wave falls off exponentially with the distance (d) from the surface as

$$E = E_o e^{-d/d_p} \tag{3.10}$$

Here E_0 is the amplitude of electric field at the interface and d_p is the depth of penetration of IR beam. Depth of penetration as a function of refractive index of ATR crystal μ2,refractive index of sample (μ_1), wavelength of incident light (λ) and angle of incidence θ is given by

$$d_p = \frac{\lambda}{2\pi\sqrt{(sin^2\theta - \left(\frac{\mu_2}{\mu_1}\right)^2}} \tag{3.11}$$

ATR spectroscopy measures changes in a total internally reflected infrared beam at the detector. In the MIR region, an infrared beam is directed through an ATR crystal at an angle greater than the critical angle. The IR beam is reflected multiple times within the ATR crystal. This internal reflectance creates an evanescent wave that extends beyond the surface of the crystal into the sample, which is in direct contact with the crystal, as shown in Figure 3.28. The penetration depth of the evanescent wave will be in the order of a few microns (0.5 μ - 5 μ) beyond the crystal surface and into the sample. At the evanescent wave, the input IR beam will be attenuated or altered before reaching the detector, this attenuation being the characteristic of the sample under test.

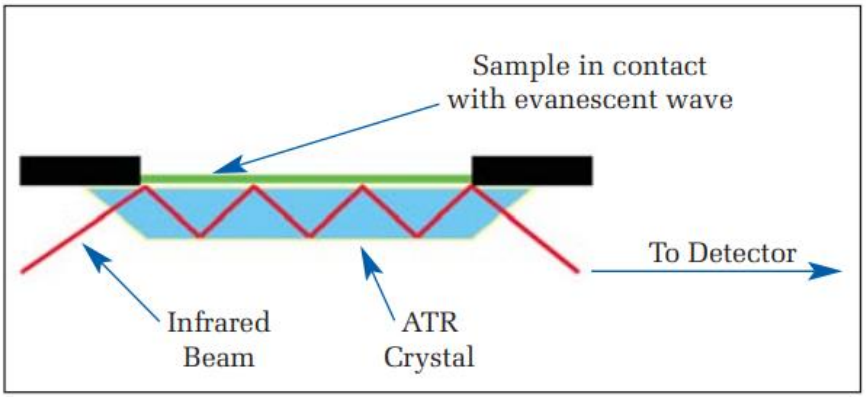


Figure 3.28 ATR Crystal Evanescent wave

For the technique to be successful, the following requirements must be fulfilled.

- The test sample must be in direct contact with the ATR crystal without forming gap or air bubbles.

- The refractive index of the ATR crystal must be significantly greater than that of the sample, for internal reflectance to occur.

There are several types of ATR crystals like Zinc selenide (ZnSe), Germanium (Ge), Diamond, and Silicon (Si). ZnSe crystal has a refractive index of 2.4 with a spectral range from 650 cm^{-1} to 15000cm^{-1} and a penetration depth of 2.01 microns. Cleaning ATR crystal is an important task while doing the experiments. A scratch on the crystal may make it unusable further. It is recommended to use liquids with a pH of more than 8 for cleaning the crystal.

3.6.2. ATR sampling -experimental setup

Figure 3.29 depicts the block diagram of the experimental setup for the edible oil analysis using MIR spectroscopy with ATR sampling technique. The experiment was carried out using a Pyreos 750 ATR spectrometer with a spectral range of (5.5µm-11µm). This spectrometer features an IR source based on Micro-Electro-Mechanical Systems (MEMS), a variable filter array, and a 128 uncooled pyroelectric detectors array unit, each separated by 100 µm. The sample can be directly placed on the Znse ATR crystal, which is mounted horizontally on the top side of the spectrometer. An Ethernet to USB converter is used to connect the spectrometer to the computer.

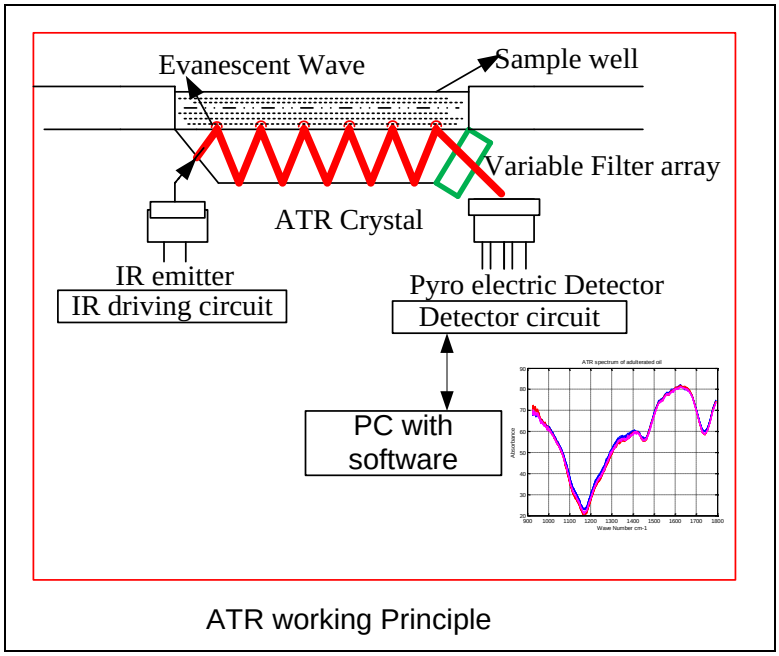


Figure 3.29 Block diagram of ATR sampling experimental setup

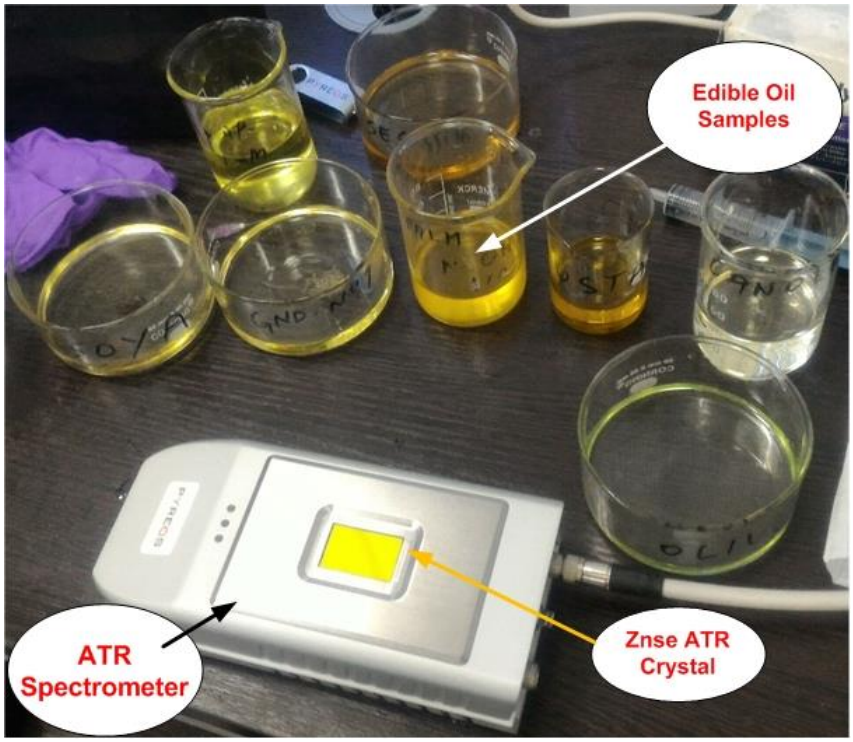


Figure 3.30 ATR spectroscopy lab setup

The IR source and ATR crystal are configured so that the IR beam is reflected nine times within the crystal before reaching the detector. The manufacturer's software (Sphinx Suite) was used to control the interfacing and data acquisition.

The MIR spectroscopy with ATR sampling experiment was used to classify edible oils such as groundnut oil (GNUT), canola oil (CAN), mustard oil (MUS), olive oil (OL), safflower oil (SAFF), soya oil (SOYA), sunflower oil (SUN), palm oil (PALM), and sesame oil (SES). Groundnut, mustard, sesame, cottonseed, and palm oils, on the other hand, are used for adulteration detection and calibration. Adulterated oil sample preparation is explained below.

3.6.3. Preparation of lab made adulterated samples

Five adulterated sample sets have been prepared for the experimentation. The edible oils are mixed in the proportions of 5%(v/v), 10%, 15%, 25%, 50%, and 75%. These prepared adulterated samples were thoroughly mixed with the help of a magnetic stirrer for uniform mixing. The following adulteration sample sets were prepared for ATR sampling experiment using the above-mentioned proportions.

Sample set 1: Adulteration of groundnut oil with cottonseed oil (5%,10%,15%,25%,50% and 75%(v/v)).
Sample set 2: Adulteration of sesame oil with cottonseed oil (5%,10%,15%,25%,50% and 75% (v/v)).
Sample set 3: Adulteration of Sunflower oil with palm oil (5%,10%,15%,25%,50% and 75% (v/v)).
Sample set4: Adulteration of Mustard oil with palm oil (5%,10%,15%,25%,50% and 75% (v/v)).
Sample set 5: Adulteration of groundnut oil with palm oil (5%,10%,15%,25%,50% and 75% (v/v)).

3.6.4. Experimental procedure

Before conducting the experiment, the ATR crystal is cleaned with acetone and a lint-free cloth (which does not produce fluff and is less likely to accumulate charge). A background spectrum is obtained in the absence of any sample on the crystal. This is used as a reference for acquiring and correcting spectra. Using a micropipette, 2ml of an edible oil sample is poured onto ATR crystal, ensuring that no bubbles form. The software provided by the manufacturer is used to acquire ATR spectra. The acquisition of spectra for one sample is repeated five times, and the average of the five readings is considered one sample spectrum. For each sample, fifteen (15) such spectra are collected, resulting in a total data matrix of size 105x128.

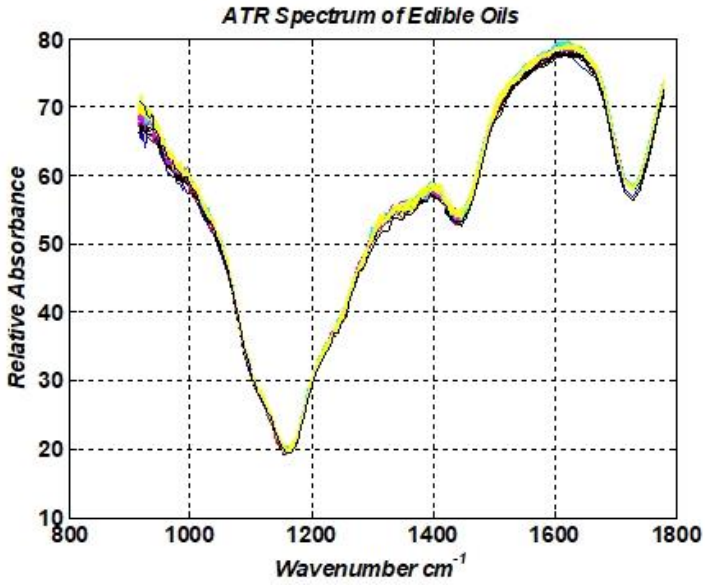


Figure 3.31ATR sampling spectra of edible oils

For the remaining samples (edible oil samples for classification and prepared samples for adulteration detection), the same procedure is followed. Following each sample reading, the ATR crystal is cleaned with acetone and a lint-free cloth. Figure

3.31 depicts the acquired sample spectrum. ATR sampling spectra data analysis for classification of edible oils and detection of adulteration is presented in Chapter 4.

3.7. Summary

Analytical methods for edible oil analysis and their working principle are explained. Sample preparation for electronic tongue and spectroscopy is presented. Data acquisition methodology for edible oil analysis using electronic tongue based on voltammetry, electrochemical impedance spectroscopy, near-infrared spectroscopy, and attenuated total reflection spectroscopy is presented. A total of nine varieties of edible oils are used for experimentation. For the classification of edible oils electronic tongue, NIR spectroscopy, and MIR spectroscopy with ATR sampling methods are employed. The simple visualization of these experimental responses is presented. Electronic tongue experiments (Voltammetry and EIS method) for edible oil analysis necessitate the use of chemicals such as petroleum ether in sample preparation, which is a laborious and time-consuming procedure. Though the results of the electronic tongue are promising in edible oil classification, it has not been used in adulteration detection experiments in our research work. NIR spectroscopy experiments, on the other hand, do not necessitate the use of any chemicals in sample preparation, and the classification outcomes are as predicted. However, the sample acquisition period with a NIR spectrometer is nearly 6 minutes. Since NIR spectroscopy involves the use of a cuvette for sample holding during the experiment, cleaning the cuvette properly is extremely difficult due to the sticky nature of oil samples, since residual stains in edible oil samples may interfere with other sample readings, so they are not used for adulteration detection experiments. MIR spectroscopy with ATR sampling experiments is simple and efficient for the edible oil analysis and cleaning of ATR crystal is also easy and straightforward. Considering the shortcomings of electronic tongue and NIR spectroscopy, as well as our motivation to use simple but accurate instrumental technique for edible oil analysis, MIR spectroscopy with ATR sampling has been considered for adulteration detection experiments. Five sets of adulterated edible oil samples are prepared: cottonseed adulteration in groundnut oil and sesame oil, palm oil adulteration in groundnut, mustard, and sesame oils, and sesame oil adulteration in groundnut, mustard, and sesame oils. The acquired data is saved, and a simple visualization of the analytical instrument response, such as a voltammogram and spectrum, is presented.

Bibliography

[1] ISO, "Food safety management systems — Requirements for any organization in the food chain," *En Iso 220002018*, vol. 3, p. 32, 2018,Available: https://www.iso.org/standard/65464.html.

[2] S. Sumaedi and M. Yarmen, "The Effectiveness of ISO 9001 Implementation in Food Manufacturing Companies: A Proposed Measurement Instrument," *Procedia Food Sci.*, vol. 3, pp. 436–444, Jan. 2015, doi: 10.1016/j.profoo.2015.01.048.

[3] S. Kanjilal, R. B. N. Prasad, T. N. B. Kaimal, Ghafoorunissa, and S. H. Rao, "Synthesis and estimation of calorific value of a structured lipid- potential reduced calorie fat," *Lipids*, vol. 34, no. 10, pp. 1045–1055, 1999, doi: 10.1007/s11745-999-0456-7.

[4] G. Brufau, R. Codony, M. A. Canela, and M. Rafecas, "Rapid and quantitative determination of total sterols of plant and animal origin in liver samples by gas chromatography," *Chromatographia*, vol. 64, no. 9–10, pp. 559–563, Nov. 2006, doi: 10.1365/s10337-006-0034-4.

[5] K. Hayashi, M. Yamanaka, K. Toko, and K. Yamafuji, "Multichannel taste sensor using lipid membranes," *Sensors Actuators B. Chem.*, vol. 2, no. 3, pp. 205–213, Aug. 1990, doi: 10.1016/0925-4005(90)85006-K.

[6] A. K. Deisingh, D. C. Stone, and M. Thompson, "Applications of electronic noses and tongues in food analysis," *Int. J. Food Sci. Technol.*, vol. 39, no. 6, pp. 587–604, Jun. 2004, doi: 10.1111/j.1365-2621.2004.00821.x.

[7] W. Wardencki, T. Chmiel, and T. Dymerski, "Gas chromatography-olfactometry (GC-O), electronic noses (e-noses) and electronic tongues (e-tongues) for in vivo food flavour measurement," in *Instrumental Assessment of Food Sensory Quality*, Elsevier, 2013, pp. 195–229.

[8] F. J. Heredia, M. L. González-Miret, A. J. Meléndez-Martínez, and I. M. Vicario, "Instrumental assessment of the sensory quality of juices," in *Instrumental Assessment of Food Sensory Quality*, Elsevier, 2013, pp. 565-610e.

[9] R. M. Aadil, G. M. Madni, U. Roobab, U. Ur Rahman, and X. A. Zeng, "Quality control in beverage production: An overview," in *Quality Control in the Beverage Industry: Volume 17: The Science of Beverages*, Elsevier, 2019, pp. 1–38.

[10] M. L. Rodríguez-Méndez, C. Apetrei, and J. A. De Saja, "Electronic Tongues Purposely Designed for the Organoleptic Characterization of Olive Oils," in *Olives and Olive Oil in Health and Disease Prevention*, Elsevier Inc., 2010, pp. 525–532.

[11] A. D. Wilson and M. Baietto, "Applications and advances in electronic-nose technologies," *Sensors*, vol. 9, no. 7. Multidisciplinary Digital Publishing Institute (MDPI), pp. 5099–5148, Jul. 2009, doi: 10.3390/s90705099.

[12] A. J. Bard, L. R. Faulkner, and J. Wiley, "Electrochemical Methods Fundamentals and Applications," 2001.

[13] U. Author, "Electrochemistry (potentiometry)," *Med. Chem.*, pp. 15–18, 2015, Accessed: Feb. 22, 2021. [Online]. Available: http://tera.chem.ut.ee/~koit/arstpr/pot_en.pdf1http://tera.chem.ut.ee/~koit/arstpr/pot_en.pdf.

[14] A. Bratov, N. Abramova, and A. Ipatov, "Recent trends in potentiometric sensor arrays-A review," *Analytica Chimica Acta*, vol. 678, no. 2. Elsevier, pp. 149–159, Sep. 30, 2010, doi: 10.1016/j.aca.2010.08.035.

[15] M. Łabańska, P. Ciosek-Skibińska, and W. Wróblewski, "Critical Evaluation of Laboratory Potentiometric Electronic Tongues for Pharmaceutical Analysis—An Overview," *Sensors*, vol. 19, no. 24, p. 5376, Dec. 2019, doi: 10.3390/s19245376.

[16] V. Semenov *et al.*, "Determination of three quality parameters in vegetable oils using potentiometric e-tongue," *J. Food Compos. Anal.*, vol. 75, pp. 75–80, Jan. 2019, doi: 10.1016/j.jfca.2018.09.015.

[17] T. V. Shishkanova, G. Broncová, A. Skálová, V. Prokopec, M. Člupek, and V. Král, "Potentiometric Electronic Tongue for Taste Assessment of Ibuprofen Based Pharmaceuticals," *Electroanalysis*, vol. 31, no. 10, pp. 2024–2031, Oct. 2019, doi: 10.1002/elan.201900334.

[18] F. Winquist, C. Krantz-Rülcker, and I. Lundström, "Electronic Tongues and Combinations of Artificial Senses," *Sensors Updat.*, vol. 11, no. 1, pp. 279–306, Dec. 2002, doi: 10.1002/seup.200211107.

[19] X. Ceto, M. Gutierrez-Capitan, D. Calvo, and M. Del Valle, "Beer classification by means of a potentiometric electronic tongue," *Food Chem.*, vol. 141, no. 3, pp. 2533–2540, Dec. 2013, doi: 10.1016/j.foodchem.2013.05.091.

[20] M. S. Cosio, M. Scampicchio, and S. Benedetti, "Electronic Noses and Tongues," in *Chemical Analysis of Food: Techniques and Applications*, Elsevier Inc., 2012, pp. 219–247.

[21] M. Juan-Borrás, J. Soto, L. Gil-Sánchez, A. Pascual-Maté, and I. Escriche, "Antioxidant activity and physico-chemical parameters for the differentiation of honey using a potentiometric electronic tongue," *J. Sci. Food Agric.*, vol. 97, no. 7, pp. 2215–2222, May 2017, doi: 10.1002/jsfa.8031.

[22] F. Winquist, P. Wide, and I. Lundström, "An electronic tongue based on voltammetry," *Anal. Chim. Acta*, vol. 357, no. 1–2, pp. 21–31, Dec. 1997, doi:

10.1016/S0003-2670(97)00498-4.

[23] F. Winquist, C. Krantz-Rülcker, and I. Lundström, "A Miniaturized Voltammetric Electronic Tongue," *Anal. Lett.*, vol. 41, no. 5, pp. 917–924, Apr. 2008, doi: 10.1080/00032710801934809.

[24] Y. Yu *et al.*, "Pure Milk Brands Classification by Means of a Voltammetric Electronic Tongue and Multivariate Analysis," 2015. Accessed: Feb. 22, 2021. [Online]. Available: www.electrochemsci.org.

[25] R. Gulaboski, V. Mirčeski, and S. Mitrev, "Development of a rapid and simple voltammetric method to determine total antioxidative capacity of edible oils," *Food Chem.*, vol. 138, no. 1, pp. 116–121, May 2013, doi: 10.1016/j.foodchem.2012.10.050.

[26] S. Mendoza, E. Bustos, J. Manríquez, and L. A. Godínez, "Voltammetric Techniques," in *Agricultural and Food Electroanalysis*, Chichester, UK: John Wiley & Sons, Ltd, 2015, pp. 21–48.

[27] A. Riul, R. R. Malmegrim, F. J. Fonseca, and L. H. C. Mattoso, "An artificial taste sensor based on conducting polymers," *Biosens. Bioelectron.*, vol. 18, no. 11, pp. 1365–1369, Oct. 2003, doi: 10.1016/S0956-5663(03)00069-1.

[28] Gamry Instruments, "Electrochemical Impedance Spectroscopy-EIS-Electrochemical Technique." https://www.gamry.com/application-notes/EIS/a-snapshot-of-electrochemical-impedance-spectroscopy/ (accessed Mar. 05, 2021).

[29] P. Ciosek and W. Wróblewski, "Sensor arrays for liquid sensing - Electronic tongue systems," *Analyst*, vol. 132, no. 10. Royal Society of Chemistry, pp. 963–978, Sep. 24, 2007, doi: 10.1039/b705107g.

[30] Y. Zou, H. Wan, X. Zhang, H. Da, and P. Wang, "Electronic nose and electronic tongue," in *Bioinspired Smell and Taste Sensors*, Springer Netherlands, 2015, pp. 19–44.

[31] A. Wadehra and P. S. Patil, "Application of electronic tongues in food processing," *Analytical Methods*, vol. 8, no. 3. Royal Society of Chemistry, pp. 474–480, Jan. 21, 2016, doi: 10.1039/c5ay02724a.

[32] P. C. Panchariya and A. H. Kiranmayee, "Statistical Feature Extraction and Recognition of Beverages Using Electronic Tongue," *Sensors Transducers J.*, vol. 112, pp. 47–63, 2010.

[33] Wikipedia, "Spectroscopy overview - Absorption spectroscopy - Wikipedia." https://en.wikipedia.org/wiki/Absorption_spectroscopy#/media/File:Spectroscopy_overview.svg.

[34] T. Ringsted, "Near infrared spectroscopy of food systems using a

supercontinuum laser," 2017.

[35] M. H. Penner, "Basic Principles of Spectroscopy," Springer, Boston, MA, 2010, pp. 375–385.

[36] B. Stuart, "Infrared Spectroscopy," in *Kirk-Othmer Encyclopedia of Chemical Technology*, Hoboken, NJ, USA: John Wiley & Sons, Inc., 2015, pp. 1–18.

[37] P. J. Larkin, *Infrared and Raman Spectroscopy: Principles and Spectral Interpretation*. 2017.

[38] B. Öztürk, A. Yalçin, and D. Özdemir, "Determination of olive oil adulteration with vegetable oils by near infrared spectroscopy coupled with multivariate calibration," *J. Near Infrared Spectrosc.*, vol. 18, no. 3, pp. 191–201, Jun. 2010, doi: 10.1255/jnirs.879.

Chapter 4

Data Analysis: Edible Oil Classification and Qualitative Detection of Adulteration

4.1. Preamble

The identification of different varieties of edible oils is the first challenging task in edible oil analysis, followed by the accurate detection of adulteration in edible oils. The two different methods for edible oil analysis are Subjective and Objective [1]. Subjective methods are based on human perception. Objective methods use analytical instruments for the evaluation of quality of edible oils. Analytical instruments can accurately and consistently assess the quality of edible oils. However, modern analytical instruments generate a large amount of data, and every component of recorded data may not be necessary to derive important information; There may also be redundant data. There is a need for statistical or multidimensional data analysis to understand patterns in data, visualize the information, and draw useful conclusions from it. Data analysis is the process of cleaning, transforming, and extracting information from data. Depending on the data distribution and field of application, there are various types of data analysis techniques [2], [3].

This chapter describes the analysis of data acquired from analytical experiments using chemometrics and artificial intelligence (AI) algorithms to classify different types of edible oils and detect adulterations in few of them using lab made adulterated samples as reference. The first section of the chapter discusses the foundations of statistics and multivariate data analysis, as well as the methodology of several supervised and unsupervised algorithms. This chapter's later sections focus on the development of AI based chemometric algorithms for the classification of edible oils using e-tongue and spectroscopy data. Developed AI based algorithms for detecting and quantifying adulteration using mid infrared spectroscopy with ATR sampling data are presented. The performance of these algorithms in classification and adulteration detection, as well as a suitable algorithm for embedded implementation, are also presented.

4.2. Chemometrics and data analysis

Chemometrics is a branch of data science that utilizes statistical analysis and AI algorithms to extract useful information from multidimensional data (acquired from analytical instruments) [4], [5]. Statistical analysis is the application of a statistical process on a dataset to establish a mathematical relationship between one or more variables. Mathematical modeling [6] helps to convert real-world problems into mathematical formulations that can be analyzed to understand more about the data and its application. Chemometric algorithms include statistical model-based algorithms and soft computing algorithms. Artificial intelligence algorithms are a set of machine learning programs that try to mimic the human brain and perform tasks like human beings [7]. The following sections explain the statistics, statistical algorithms, and soft computing algorithms for univariate and multivariate data.

Statistics is a branch of applied mathematics that deals with the collection, analysis, interpretation, inference, and presentation of data. Statistics is broadly classified into two categories: descriptive statistics and inferential statistics [8], [9].

4.2.1. Descriptive statistics

Descriptive statistics involves the description of the features of sample data, which provides essential information about the variables to be investigated [10]. Two fundamental properties of sample data are the measure of central tendency (location) and the measure of variability (dispersion) which are further described by more terms. Figure 4.1 depicts a summary of descriptive statistics.

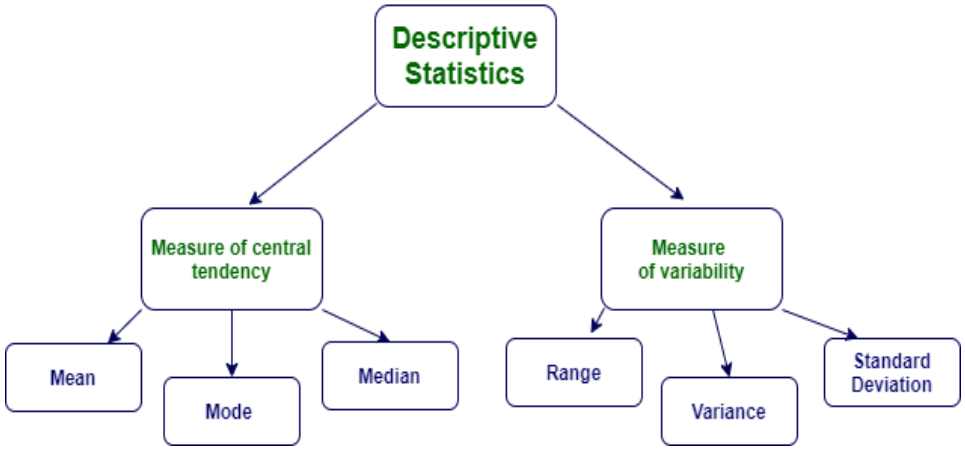


Figure 4.1 Descriptive statistics

The clustering of data around a central value is regarded as a central tendency, which is a statistical measure. This central value appropriately describes the entire data set. The mean and median are two commonly used measures of central tendency. Mean is defined by Eq. 4.1.

$$\text{mean } (\bar{x}) = \frac{x_1+x_2+x_3 \ldots x_n}{n} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{4.1}$$

The outliers in a data set have a large impact on the mean value of the data set [11] and intelligent decision is to be taken about their removal.

Median is the value separating the higher half from the lower half of the data set, or a probability distribution. For a given dataset with n number of variables, arranged in increasing or decreasing order, the median is the middlemost value. It is always unique for a given data set. When n is odd, $\frac{(n+1)}{2}$ thvariable is the median of the data set. When n is even, average of the $\frac{(n+1)^{th}}{2}$ and $\left(\frac{n}{2}\right)^{th}$ variables give the median. Because it is determined by the order of data, the median is usually unaffected by outliers in the data. The order is unaffected by the highest or lowest value [11].

The variability (dispersion) of a data set corresponds to the discrepancies among the data values. The goal of dispersion measures is to determine how widely the variables or data points are distributed. The most common dispersion measures are range, variance, and standard deviation. The range is the simplest measure of spread in the data. For a given data set $X = \{x_1, x_2, x_3, \ldots x_n\}$, the range is defined as the interval from minimum to maximum values.

$$\text{Range} = \max(X) - \min(X) \tag{4.2}$$

The presence of outliers in the data set influences the range value. As a result, it is not a reliable measure for the data set. Variance is one of the statistic measures which tells about the extent of the dispersion or spread of the distribution around its mean. The variance is not robust to outliers since any data value or point separate from the body of the data can increase the value of the variance by an arbitrarily large amount. Suppose the data entries are denoted by $X = \{x_1, x_2, x_3, \ldots x_n\}$where n is the sample size. Variance is denoted by S^2 or σ^2 is given by

$$\sigma^2 = \frac{(x_1-\bar{x})^2+(x_2-\bar{x})^2+(x_3-\bar{x})^2 \ldots (x_n-\bar{x})^2}{(n-1)} \tag{4.3}$$

Variance is an indicator for the closeness of data points to the mean of the data set. Standard deviation is the square root of the variance, which is denoted by σ. It also describes the spread or the closeness of data points to their mean. σ has same unit as the variables (x). Describing data with a few numbers, location measures, and spread measures is a simple task, but it can be misleading.

Aside from above mentioned standard statistical measures, there are some graphical and pictorial representations of data. These graphical representations serve as a tool for comprehending the characteristics and nature of the data and understand the relation between variables. There are various types of descriptive plots, but the box plot and histogram are the most important and widely used [11], [12].

In brief, to understand the data and to find the relation between the variables in the data set, descriptive statistics are used.

4.2.2. Inferential Statistics

Inferential statistics make use of patterns in sample data to draw conclusions about the population being represented. These inferences could take the form of answering yes/no questions about the data (hypothesis testing), estimating numerical characteristics of the data (estimation), describing associations within the data (correlation), or modeling relationships within the data (for example, using regression analysis). The inference may extend to forecasting, predicting, and estimating unobserved values either in or associated with the population being studied. It can even include time series or spatial data extrapolation and interpolation, as well as data mining [12].

4.3. Multivariate data analysis and Artificial Intelligence

Data collected in an experiment may include observations or measurements of a single variable, such as a person's height, weight, temperature, or pH value of a sample. Because it only contains information about one characteristic variable, this data is referred to as univariate data [13]. Univariate data can be visualized using histograms, bar charts, and pie charts, and inferences can be drawn as needed.

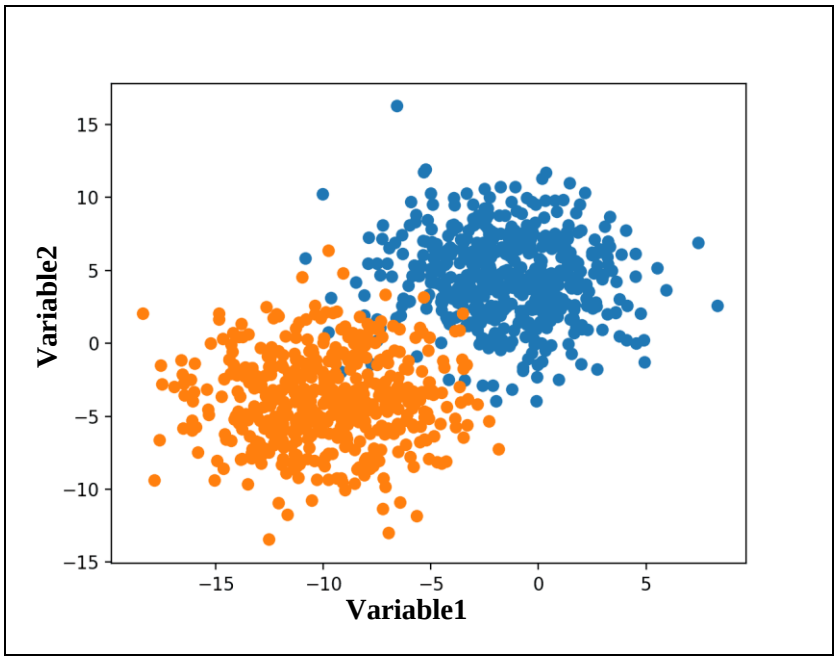


Figure 4.2 Example for a bivariate scatter plot

Bivariate data consists of observations of two distinct characteristic variables. The relationship or trend of one variable with another variable is investigated in the analysis [14]. A scatter plot shown in Figure 4.2 for visualizing bivariate data.

Multivariate data, on the other hand, includes observations from more than two variables. Multivariate statistics and Artificial Intelligence (AI) are used to analyse multivariate data and draw conclusions about correlation, patterns and clusters [15]. Artificial intelligence is a technology that allows us to create intelligent systems that mimic human intelligence. Machine Learning (ML) is an application of AI that enables machines to learn and perform tasks proficiently. Machine learning research is currently focusing on computer vision, pattern recognition, image processing, cognitive computing, and intelligent instrumentation. Based on the learning process and input-output data four broad types of learning methods are described in the following subsections. They are

1. Supervised learning
2. Unsupervised learning
3. Semi-supervised learning
4. Reinforced learning

4.3.1. Supervised learning

Supervised learning is a machine learning task that is carried out on a known dataset with known intended results for each instance. Based on this data, supervised learning constructs a mapping from input variables to output variables using mathematical and statistical modeling [16]. Figure 4.3 depicts the supervised learning methodology.

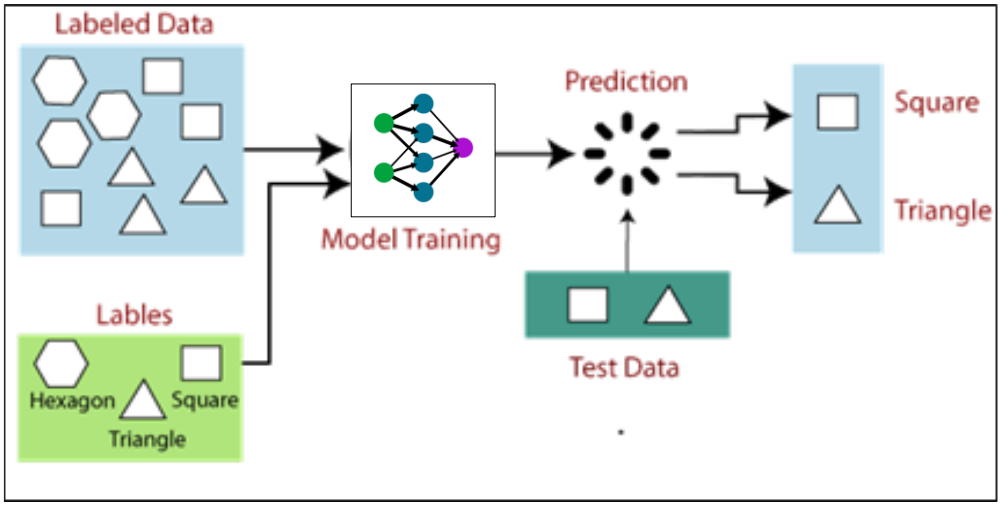


Figure 4.3 Supervised learning method

Output data may be categorical label data or continuous/discrete numerical data. Once the model is trained, the mapping function is used to predict unknown sample output using the same features that were used to train the model. The model is trained using training data (data with correct labeling), and its learning performance is evaluated using test data which has not been shown before. The developed supervised learning model assigns a label to test data based on what it has learned from training data. Discriminant algorithms (LDA), class modeling algorithms (SIMCA), and regression models (PLSR, MLR) are examples of supervised algorithms [17].

4.3.2. Unsupervised learning

Unsupervised learning is a machine learning task that learns from data without knowing the actual desired output. The algorithm attempts to identify patterns and trends in the data and then draws inferences based on these patterns. Clustering algorithms (CA) like k-means, dimension reduction algorithms like Principal Component Analysis(PCA), kohenon SOM are examples of unsupervised learning methods [18]. Figure 4.4 depicts the unsupervised learning principle.

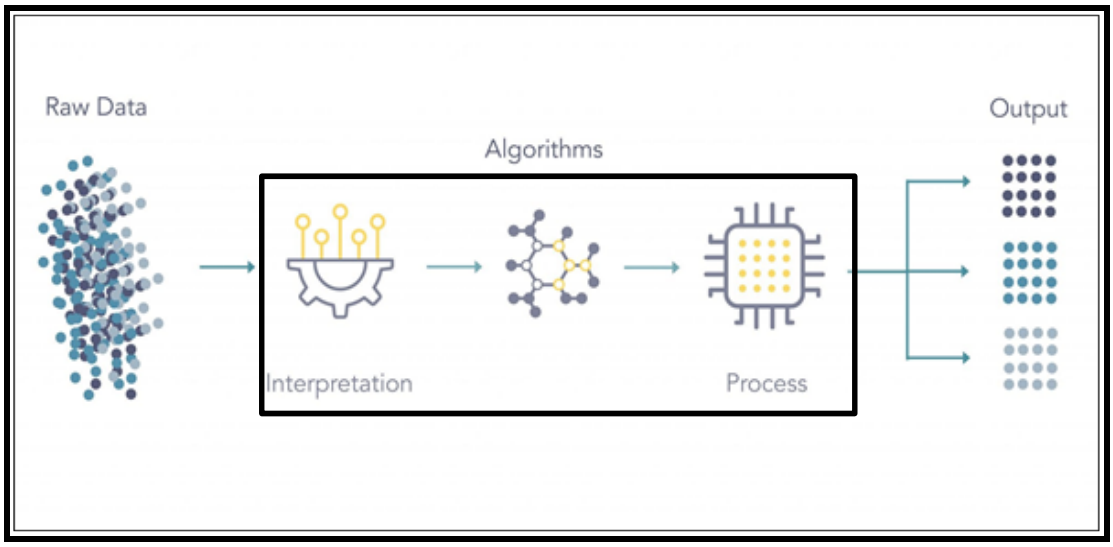


Figure 4.4 Unsupervised learning method

4.3.3. Semi-supervised learning

Semi-supervised learning is a machine learning approach in which a model learns from a small amount of labelled data while being trained on a large amount of unlabeled data [19]. Semi-supervised learning is a type of learning that falls between unsupervised learning (with no labeled training data) and supervised learning (with only labeled training data). Generative Adversarial Networks (GAN) are an example of a semi-supervised algorithm.

4.3.4. Reinforcement learning

Reinforcement learning (RL) is the learning process of interacting with the immediate environment. An RL agent (the learner and the decision-maker) learns from the consequences of its actions rather than being explicitly taught, and it decides its actions based on earlier experiences (exploitation) along with new choices (exploration), which is essentially trial and error learning. The RL-agent receives a piece of reinforcement information in the form of a numerical reward that encodes the success of an action's outcome, and the agent seeks to learn to select actions that maximize the accumulated reward over time [20]. The interactions between the agent and the environment are depicted in Figure 4.5.

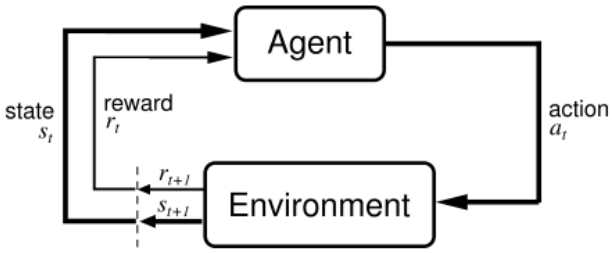


Figure 4.5 Agent and environment interaction in Reinforcement learning

In this thesis, unsupervised and supervised learning approaches are used to construct chemometric classification and regression models. The ML approaches and algorithms employed in the design of classification and regression models for the analysis of edible oils are depicted in Figure 4.6.

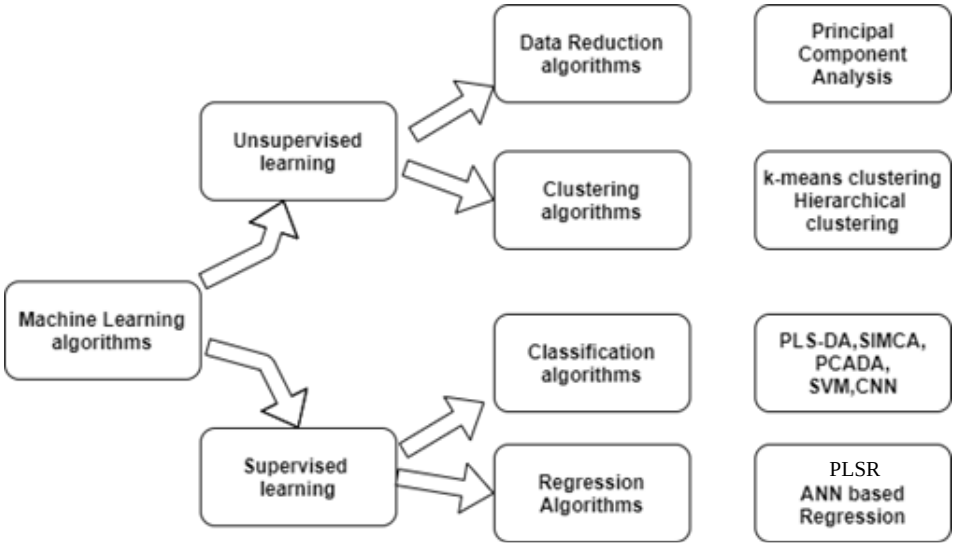


Figure 4.6 Machine learning methods and algorithms overview

There are several steps involved in developing artificial intelligence and machine learning models for a specific application. Each step in the model development process is critical for achieving the best results. In this work, the following steps were employed for model development (classification and regression), and they are explained in the sections that follow. The steps involved in the development of algorithms are depicted in Figure 4.7.

1. Data pre-processing
2. Feature selection
3. Chemometric models development, testing, and validation

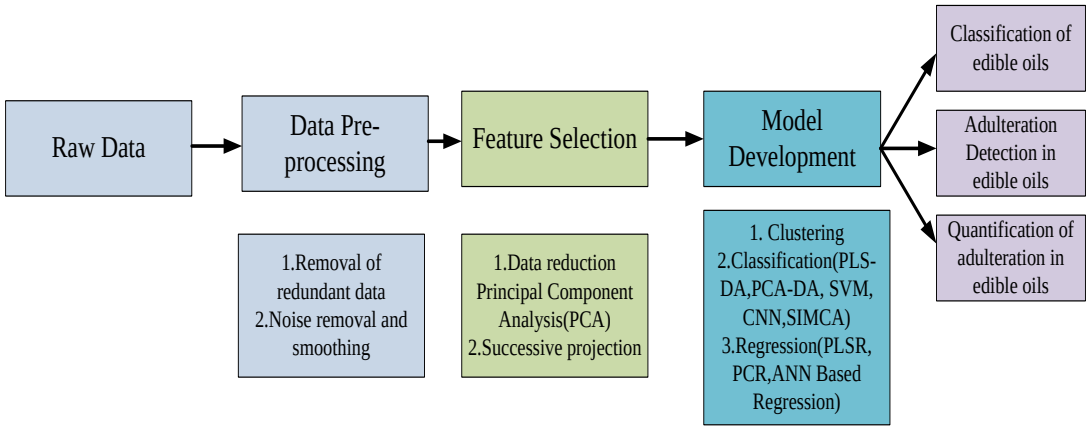


Figure 4.7 Data analysis steps in AI model development

4.4. Data pre-processing

Data pre-processing is an important step in multivariate data analysis. It is used to reduce random noise in data and remove redundant information. In the case of electronic tongues, data will be collected from multiple sensors(electrodes). Each electrode generates a large amount of data. Not every instance may be required for the analysis and extraction of useful information. Furthermore, in sensor fusion technology, each sensor can provide measurements in different scales. Each measurement must be brought to the same scale for further analysis to perform multivariate analysis. It is also necessary to make the data compatible with any distribution system (normal or gaussian/poisson) to perform statistical inference [21]. Improper use of pre-processing methods on experimental data may not yield the desired results. The data pre-processing methods used in this research work are described in the following sections.

4.4.1. Normalization

Normalization is the process of converting all the measured variables in multivariate data to a common scale without distorting the data trend or pattern and without losing any information. The standard method in data pre-processing is min-max normalization [22], which converts the measured variables from zero to one {0 to1}.

If $X = \{x_1, x_2, x_3, \ldots x_n\}$ is an input data, the min max normalized variable is given by

$$X_{norm} = \frac{(x_i - X_{min})}{(X_{max} - X_{min})} \tag{4.4}$$

4.4.2. Data standardization

The process of fitting any distribution data into a standard normal distribution is known as standardization. This is accomplished by subtracting the mean from the data and dividing by the standard deviation. A standard normal distribution will have a zero mean and a unity standard deviation [23]. Standard normal variate (denoted as z-score) of a value in the data set $X = \{x_1, x_2, x_3, \ldots x_n\}$ is denoted by (z) and calculated as

$$z = \frac{(x-\bar{x})}{\sigma} \tag{4.5}$$

The z score of observation indicates how far it deviates from the mean. A positive z score indicates the value is to the right side to the mean, and the negative value indicates that it is left side to the mean. The z score values tell us the area covered under the normal distribution curve. In spectroscopy, the z-score is very useful for reducing particle scattering effects.

4.4.3. Multiplicative Scattering Correction (MSC)

Multiplicative Scatter Correction (MSC) is a common pre-processing technique in spectroscopy data analysis. It is used to correct both scaling and offset variations[24]. For a single spectrum, the equation describing the scattering contributions in addition to the spectral signal, is expressed by an additive and a multiplicative term,

$$x = a + b\hat{x} + e \tag{4.6}$$

where, $\hat{x}$ is the reference spectrum, usually taken as an average of all spectra. a and b coefficients are calculated by fitting a sample spectrum (x) to reference spectrum $(\widehat{x})$ by least square regression principle. The corrected spectrum (x_{msc}) is given by

$$x_{msc} = \frac{x-a}{b} \tag{4.7}$$

4.4.4. Data smoothing

Data smoothing is a statistical method for reducing noise, revealing patterns, and removing outliers from data. Moving average is a simple smoothing technique.

$$y[n] = \sum_{k=0}^{m-1} \frac{x[n-k]}{m} \tag{4.8}$$

Savitzky-Golay smoothing is a digital filter for smoothing and improving the S/N ratio without distorting the signal [25], [26]. This is accomplished by using a weighting

coefficient (convolution coefficients) and the least-squares method to fit successive adjacent data points with a polynomial of low degree. This method is computationally more effective and much faster. The selection of the number of points is critical in this smoothing filter. The smoothed data point by this filter is given by

$$Y_k = \frac{\sum_{i=-n_L}^{n_R} A_i Y_{i+k}}{\sum_{i=-n_L}^{n_R} A_i} \tag{4.9}$$

Where A_i is the convolution coefficient, in this case, n_L is the number of points used "to the left" of a data point Y_i (i.e., earlier than Y_i so negative sign is used, whereas $n_R$is the number used to the right, i.e., later than Y_i).

After pre-processing the data, unsupervised, and supervised models are employed to draw inferences from it. The sections that follow discuss unsupervised learning (dimensionality reduction and clustering algorithms) and supervised learning models (classification and regression models) that have been used in this work for the analysis of edible oils.

4.5. Unsupervised learning methods

Unsupervised learning algorithms aim to identify data trends and groups of similar instances(clustering) in training data, which consists of a set of input vectors with no corresponding desired outputs. Another choice with this method is to use transformations to represent a higher dimensional variable data set in a lower-dimensional variable data set, which is known as dimensionality reduction. The dimension reduction and clustering methods are described in detail in the following section.

4.5.1. Dimensionality reduction

From a given data set of variables, to draw inferences, every instance of a variable may not be necessary. Dimensionality reduction is a process of reducing the number of input variables in a dataset. These techniques are transformation or projection-based methods. Moreover, in multivariate data, the collinearity of variables, i.e., the dependency of one measured variable on another, will influence performance of model.

Dimensionality reduction techniques use projections and orthogonal transformations to find a new set of variables as a linear combination of the original input variables,

reducing collinearity among the variables [27], [28]. Principal Component Analysis (PCA) is a popular data reduction technique that is explained in the following section.

PCA is the most widely used unsupervised method for reducing the dimensionality of a dataset while retaining as much statistical information (variability) as possible. It employs an orthogonal transformation to convert a set of possibly correlated observations into a set of values of linearly uncorrelated variables known as principal components. In PCA, the original data is transformed into another feature space such that the maximum variance among the data lies in the first coordinate, referred to as the first principal component, and the second greatest variance lies on the second coordinate, and so on. The coordinates in the transformed space are linear combinations of the original feature vectors with variance preserved [29].

PCA summarizes all variances in a fewer number of axes called principal axes with coordinates called scores or Principal Components (PC). It extracts features from the observations for the users that are otherwise not apparent. PCA is used to visualize the variance among the multidimensional data in a fewer dimensional plot called a score plot. Mathematically PCA composes the data matrix X of r rank into a sum of r matrices with rank r,

$$X = TP^T + E \tag{4.11}$$

where T is the score matrix, P is the loading vector and E is the residuals.

$$X = t_1 p_1^T + t_2 p_2^T + t_3 p_3^T + t_4 p_4^T \ldots\ldots + E \tag{4.12}$$

The pseudo logic for PCA algorithm is given below as a step-by-step procedure.

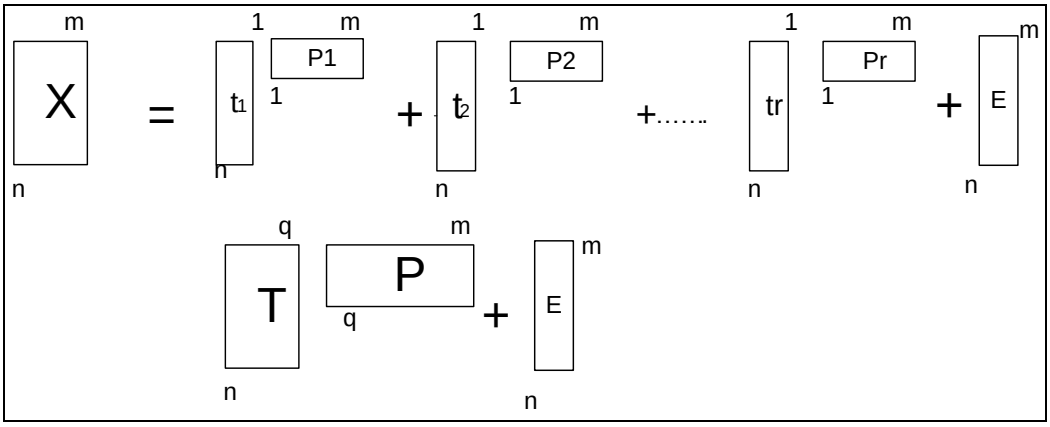


Figure 4.8 Representation of PCA

Algorithm 4.1 Principal component analysis

1: Initialize the data matrix with X_{nxp} , with n samples and p variables ;
2: Calculate mean from data set X as
$\bar{x} = \frac{1}{p}\sum_{i=1}^{p} x_i$
3: Calculate mean-cantered data (subtract mean from data)
$\mathbf{X_c = X - Mean}$
4: Calculate Covariance matrix of Xc
$\mathbf{C_{nxn} = \frac{1}{p-1}} \boldsymbol{X_c}(\mathbf{X_c})^{\mathbf{T}}$;
5: Calculate eigenvalues of the covariance matrix.
$\mathbf{C_{nxn} - \lambda I = 0, \lambda 1, \lambda 2 .... eigenvalues.}$
6: Calculate the corresponding eigenvector T_{nxk}
End

4.6. Clustering methods

Clustering is the most important unsupervised learning problem; it is concerned with discovering similar data trends in a collection of unlabeled data. Clustering could be defined as "the process of organizing objects into groups whose objects are similar to each other in a cluster and dissimilar to objects in other class. Each object is described by a set of characteristic features. The first step in dividing objects into clusters is to determine the distance between them. Selecting an appropriate distance measure is critical to the clustering process's success. The following section explains the methodology of the clustering models used in the present work.

4.6.1. K-means clustering

K-means clustering is an unsupervised non-hierarchical clustering algorithm that provides an easy way to classify a given data set into a fixed number of clusters [30]. The number of clusters(K) can be pre-defined or determined iteratively by the clustering procedure. The main idea behind this method is to define centroids for each cluster and then use the distance parameter to find data points near these centroids [31], [32]. Three types of distances, Euclidian, squared Euclidian, and city block distance are commonly used in clustering are given by the following equations.

If $\boldsymbol{p} = \{p_1, p_2, \ldots \ldots p_n\}$ and $\boldsymbol{q} = \{q_1, q_2, \ldots \ldots q_n\}$ are the data points, then

$$\text{Euclidian distance } \mathrm{d(p,q)} = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2} \tag{4.13}$$

$$\text{Squared Euclidian distance } \mathrm{d(p,q)} = \sum_{i=1}^{n}(p_i - q_i)^2 \tag{4.14}$$

$$\text{City block diastance } d(p,q) = \sum_{i=1}^{n} |p_i - q_i| \quad (4.15)$$

After the distances are calculated, each data point is clustered with a centre that is closer to it. New k cluster centres are evolved, and the procedure is repeated until no more cluster changes are possible. If $X = \{x_1, x_2 \ldots \ldots x_n\}$ be the data points and $U = \{\mu_1, \mu_2 \ldots . \mu_k\}$ be the set of initial cluster centers. The following pseudocode shows the step-by-step procedure for k-means clustering.

Algorithm 4.2: K-means clustering pseudo algorithm

Input: $X = \{x_1, x_2 \ldots \ldots x_n\}$ // Input data set
K // Number of desired clusters
Output: K //Set of clusters

Step1: Choose initial cluster centres $\{\mu_1, \mu_2 \ldots . \mu_k\}$ randomly
Step 2: Associate each data point with nearest centroid
loop
for i =0 to max iterations
$c^{(i)} = \text{argmin} \left|\left|x^i - \mu_k\right|\right|^2$ (distance between initial clusters and data points)
end for
Step 3: Recalculate the position of centres
for k=1 to K
$\mu_k = \frac{\sum_{i=1}^{m} I\{c^i=k\} * x^i}{\sum_{i=1}^{m} I\{c^i=k\}}$ // re estimate the cluster mean
end for
Repeat step 2 and 3 until there are no changes in membership of data points.
Step 4: Return data points with cluster membership ID.

4.6.2. Hierarchical clustering

Hierarchical clustering is an unsupervised clustering method that creates clusters based on a hierarchical structure. It is a tree clustering method that forms clusters based on object distance or similarity and linkage criteria. Agglomerative clustering and Divisive clustering are the two types of hierarchical clustering methods. Clustering in the Agglomerative method begins with each object as a separate cluster, and the two most similar clusters are joined into a new single cluster at each step. The Divisive method considers entire data points to be a single cluster; in each step, data points are separated into new clusters based on their dissimilarity [33], [34].

The hierarchical clustering method can be used with various distance measures such as Euclidian distance, squared Euclidian distance, and city block distances. Distances between objects are computed and arranged in a matrix called a distance matrix. The

distance matrix is a symmetric matrix with zero diagonal elements. The clustering results will be presented in the form of dendrograms, which depict the arrangement of clusters. The height of a dendrogram represents the distance between two connected objects. As shown in Figure 4.9 dendrograms can be used to interpret the number of clusters based on the distance between the objects. The following pseudo code shows the step wise procedure for agglomative clustering.

Algorithm 4.3: Hierarchical Clustering Algorithm with Complete Linkage

Input: X=$\{p_1, p_2, \ldots\ldots p_n\}$// Input data set
distance function $d(p,q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$

Step1: Assume each data point as a single cluster
for i=0 to n (number of data points)
$C_i = \{p_i\}$

Step 2: Find most similar points in the data set
$(p_{min1}, p_{min2}) = \underset{p_i p_j, \in C}{min}[d(p_i, p_j)]$

Step 3: Remove p_{min1} and p_{min2} from C

Step 4: Merge the clusters p_{min1} and p_{min2} into single cluster $p_{new1} = \{(p_{min1}, p_{min2}\}$ and add to C

Step 5: If all the data points are clustered into a single cluster, then stop, else repeat from step 2

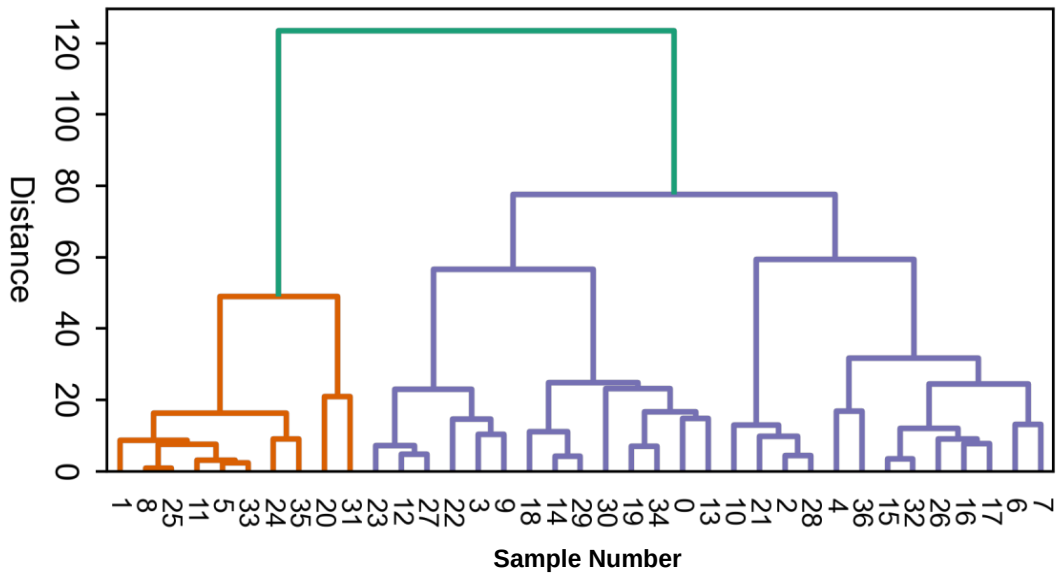


Figure 4.9 Dendrogram representation in hierarchical clustering

4.6.3. Subtractive Clustering

Subtractive clustering was proposed to overcome the computational cost of mountain clustering algorithm which estimates cluster centers by constructing and destroying the mountain feature (density) on a grid space [35]. Subtractive clustering, on the other hand, computes the mountain function on the data points instead of the grid nodes. Nikhil et, al. [36] stated that subtractive clustering has less computational cost than mountain clustering. However, since cluster centers were chosen solely from the dataset, the results might be less accurate. The subtractive clustering algorithm is described as follows:

Consider a set of data points,$X = \{x_1, x_2, \ldots, x_n,\}$ where x_i is a normalized feature space vector. Each data point x_i is considered as a potential cluster center, and its potential (P_i), a measure of the point to serve as a cluster center is defined as

$$P_i = \sum_{j=0}^{n} \frac{(||\, x_i - x_j||)^2}{\left(r_a/_2\right)^2} \tag{4.16}$$

where || || denotes the Euclidean distance and r_a is a positive constant defining a neighborhood radius. A data point with many adjacent data points has a high potential value (PV), and points outside of r_a have no influence on it. The first cluster center, c_1, is selected as the point with the highest potential. PV (c_1) represents the potential of cluster c_1, Following that, the potential of each data point x_i is revised in the following manner:

$$P_i = P_i - PV\ (c_1) exp\ (-\frac{(||x_i - c_i||)^2}{\left(r_b/_2\right)^2}) \tag{4.17}$$

r_b = 1.5r_a is commonly used to avoid cluster centers that are too close together. The data points near the first cluster center will have greatly reduced potential and will be unlikely to be chosen as the next cluster center. In general, after determining the kth cluster center c_k, the potential is revised as

$$P_i = P_i - PV\ (c_k) exp\ (-\frac{(||x_i - c_k||)^2}{\left(r_b/_2\right)^2}) \tag{4.18}$$

where c_k represents the location of the kth cluster center and PV (c_k) represents its potential value. The procedure is repeated until the stopping criterion is met i.e., all the data is within the influence range of a cluster centre. Figure 4.10 depicts the methodology of subtractive clustering.

Subtractive clustering assumes that each data point is a potential cluster center. The steps of subtractive clustering algorithm are,

1. Calculate the likelihood that each data point would define a cluster centre, based on the density of surrounding data points.
2. Choose the data point with the highest potential to be the first cluster centre.
3. Remove all data points near the first cluster centre. The vicinity is determined using a neighbourhood radius.
4. Choose the remaining point with the highest potential as the next cluster centre.
5. Repeat steps 3 and 4 until all the data is within the influence range of a cluster centre.

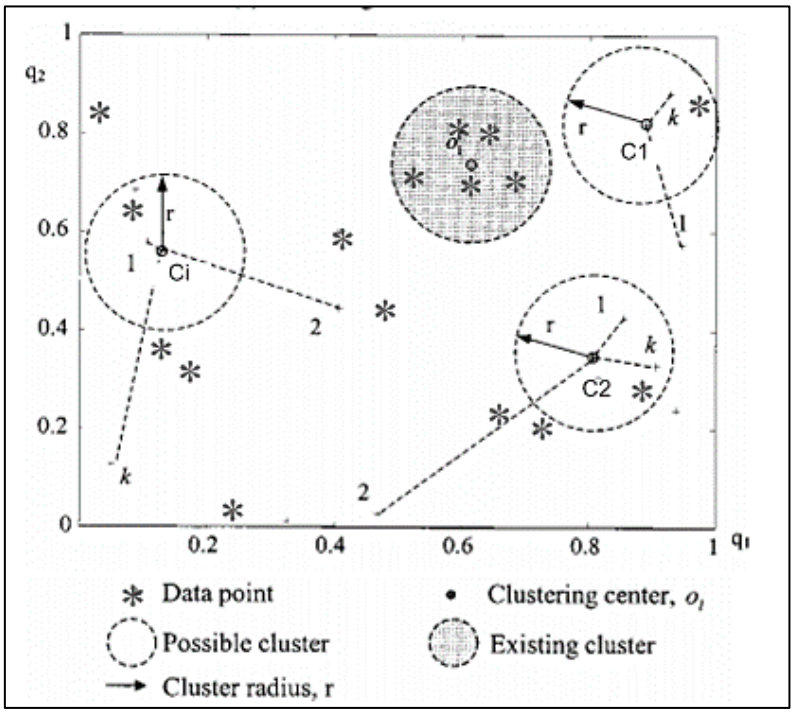


Figure 4.10 Illustration of Subtractive clustering with two-dimensional data

4.7. Classification methods

Classification methods are supervised learning techniques that categorize objects using predefined labels based on their properties. There are two types of classification algorithms: Discriminant analysis methods and Class analysis modeling methods.

4.7.1. *Discriminant Analysis (LDA)*

Discriminant analysis is a supervised classification technique that involves development of discriminant functions that maximize the ratio of between-class

variance to within-class variance to discriminate the desired categorical variable correctly. The dependent variable is categorical, while the predictor or independent variable is numerical data. To build discriminant functions, linear combinations of independent variables are used. The term categorical variable refers to the fact that the dependent variable is classified into several groups. The discrimination function attempts to rotate the axes so that the differences between the groups are maximized when the categories are projected on the new axes as shown in Figure 4.11. LDA finds the vectors in the underlying space that best discriminate among classes[37]. For all samples of all classes, the between-class scatter matrix S_B, and the within-class scatter matrix S_W are defined by

$$S_B = \sum_{i=1}^{c} N_i \, (x_i - \mu)(x_i - \mu)^T \tag{4.19}$$

$$S_W = \sum_{i=1}^{c} \sum_{x_k \varepsilon X_i} (x_i - \mu_i) \, (x_i - \mu_i)^T \tag{4.20}$$

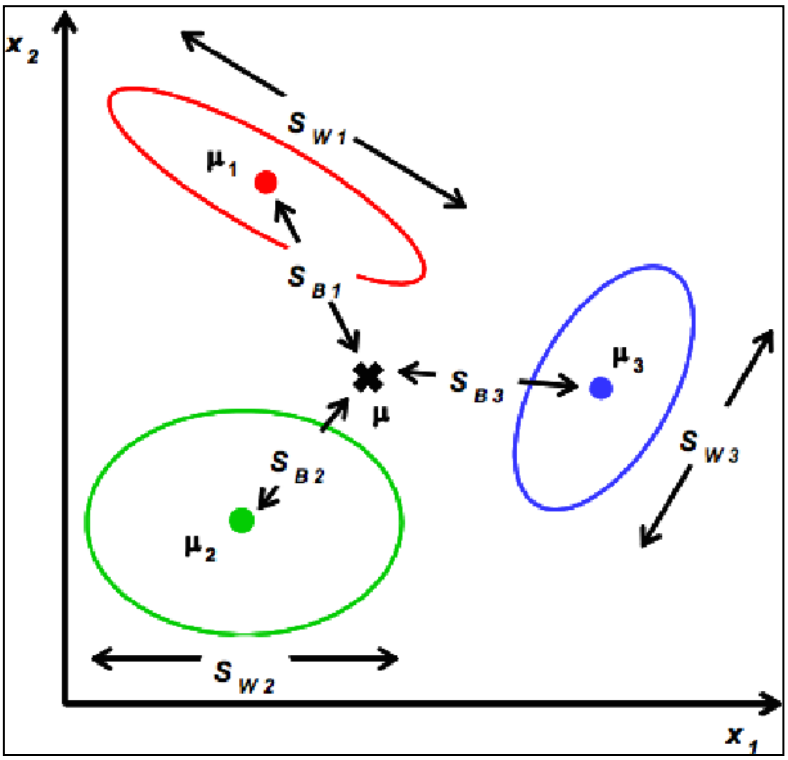


Figure 4.11 Linear Discriminant Algorithm (LDA) methodology for three class variables

where N_i is the number of training samples in class i and c is the number of distinct classes, μ_i is the mean of samples belonging to class i and X_i represents the set of samples belonging to class i with x_k being the k^{th} sample of that class. S_W represents the variation of features around the mean of each class and S_B represents the variation

of features around the overall mean for all given classes. The goal of LDA is to maximize the inter class variation (S_B) while minimizing within-class variation (S_W)[38]–[40].

4.7.2. Partial Least Squares -Discriminant Algorithm (PLS-DA)

Partial least squares discriminant analysis (PLS-DA) is a type of discriminant classification algorithm. It is a variant of partial least square regression (PLSR), with the output variable as categorical[41]–[43]. PLS-DA is derived from PLSR, where the response vector Y assumes discrete values. If X is $n \times j$ data matrix, Y is the $n \times 1$ desired result matrix. In the usual multiple linear regression model (MLR) approaches, we have

$$\mathrm{Y} = \mathrm{XB} + \mathrm{F} \tag{4.21}$$

Where B is the $j \times 1$ regression coefficients matrix and F is the $n \times 1$ error vector, and. In this approach, the least-squares solution for the B matrix is given by

$$\mathrm{B} = (\mathrm{X^T X})^{-1}\mathrm{X^T Y} \tag{4.22}$$

If input data consist of several correlated variables, then finding the inverse of $(X^T X)$ is not possible as a singular matrix is non-invertible. PLS-DA addresses this problem by converting a set of correlated variables to uncorrelated variables in lower-dimensional space like in PCA.

PLS-DA decomposes the input data matrix into two matrices, an orthogonal scores matrix $T_{n \times p}$ and loadings matrix $P_{j \times p}$.The response vector Y is also decomposed into an orthogonal score matrix $T_{n \times p}$ and loadings matrix Q_{1xp}. There are two fundamental equations in the PLS-DA model

$$\mathrm{X} = \mathrm{TP^T} + \mathrm{E} \tag{4.23}$$

$$\mathrm{Y} = \mathrm{TQ^T} + \mathrm{F} \tag{4.24}$$

Matrix T can be derived from X using a weight matrix W_{jxp}

$$T = XW(P^T W)^{-1} \tag{4.25}$$

The response vector Y can be expresses as

$$\mathrm{Y} = XW(P^T W)^{-1}\mathrm{Q^T} + \mathrm{F} = \mathrm{XB} + \mathrm{F} \tag{4.26}$$

The regression coefficient matrix B is given by

$$B = XW(P^T W)^{-1} \tag{4.27}$$

The PLS-DA algorithm estimates the matrices W, T, P, and Q through the following steps.

Algorithm 4.4: Partial Least Squares Discriminant Analysis pseudo algorithm

1: Initialize the residuals matrices $E_0 = X_{nxp}$ and $F_0 = Y_{nx1}$;
2: **for** i = 1 to p **do**
3: Calculate PLS weights vector
$\mathbf{W_i} = \mathbf{E_0}^{\mathbf{T}}\mathbf{F_0}$
4: Calculate and normalize scores vector
$\mathbf{T}_{\mathbf{i}} = \mathbf{E_0 W_i}(\mathbf{W_i}^{\mathbf{T}}\mathbf{E_0}^{\mathbf{T}}\mathbf{E_0 W_i})^{\frac{-1}{2}}$
5: Calculate X loading vector
$\mathbf{P_i} = \mathbf{E_0}^{\mathbf{T}}\mathbf{T_i}$
6: Calculate Y loading vector
$\mathbf{Q_i} = \mathbf{F_0}^{\mathbf{T}}\mathbf{T_i}$
7: Update the **X** residuals vector
$\mathbf{E_0} = \mathbf{E_0} - \mathbf{T_i}\mathbf{P_i}^{\mathbf{T}}$
8: Update the Y residuals vector
$\mathbf{F_0} = \mathbf{F_0} - \mathbf{T_i}\mathbf{Q_i}^{\mathbf{T}}$
9: **end for**
10: Obtain output matrices $\mathbf{W}, \mathbf{T}, \mathbf{P}, \mathbf{Q}$.

In the present work PLS-DA algorithm has been used for the classification of edible oils using electronic tongue based on voltammetry, electrochemical impedance spectroscopy. PLSDA has also been used for classification of edible oils, and detection of adulterations in edible oils using MIR spectroscopy with ATR sampling technique.

4.7.3. Soft Independent Modeling of Class Analogy (SIMCA)

Soft independent modeling of class analogy (SIMCA) was the first class-modeling technique introduced into chemometrics by S. Wold et al. in 1977 [44], [45]. The method builds class models based on a separate PCA sub-model performed on each class and calculating its boundary. A new unknown test sample is assigned to a specific class if it lies within these boundaries. The overall distance from each class boundary is computed as a linear combination of the sample's Q residual (Q) and Hotelling's scores (T^2) from that class PCA sub-model. PCA for three class variable data with Hotelling's T^2 and Q residual is shown graphically in Figure4.12. These overall

distance to class boundaries is compared to indicate whether a sample belongs to a specific class or not[46].

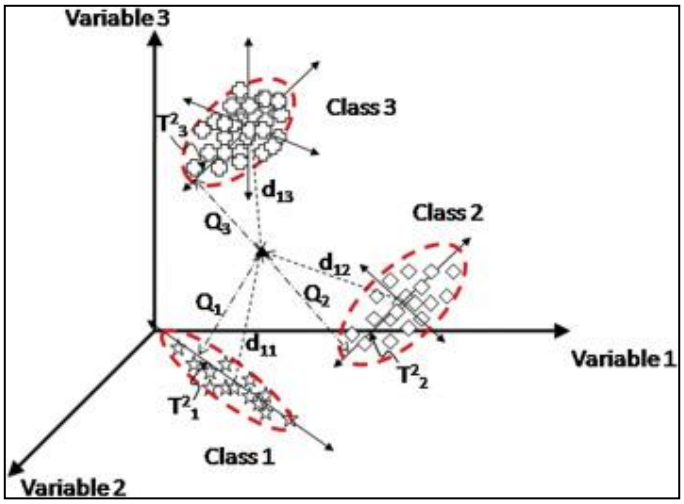


Figure 4.12 The Hotelling's T^2 and Q residual under the class variable condition

If a training data set $X_{mxn} = \{X^1, X^2, \ldots X^j\}$ consist of m samples of n variables with j number of classes. The training set is subdivided into j class matrices. A class matrix $X^j{}_{pxn}$ consist of p number of samples belonging to that class. Principal Component Analysis (PCA) is performed on all j class matrices creating a j number of PCA sub-models. Each j PCA sub-model consists of the score matrix T_{pxk}, where k is the number of retained principal components for model j and the loadings matrix P_{nxk}. Loading are the weights for each original variable when calculating the principal components. The Q residual is a measure of the variance in the data that is not accounted for by the principal components in the PCA model. The difference between measured and estimated sensor readings yields the residual E, which serves as the foundation for the Q statistic, which is expressed as follows:

$$E = X - TP^T \tag{4.28}$$

$$Q = E^T E = \sum_{i=1}^{N} E_i^2 \tag{4.29}$$

Hotelling's T^2 is a distance measure between the multivariate mean (the intersection of the Principal Components (PC) in the figure) and the sample's projection onto the two principal components. The T^2 limit of Hotelling determines an ellipse on the plane within which the data normally projects. It is the sum of normalized squared scores divided by their variance given by the following equation:

$$T^2 = t^T \lambda^{-1} t = \sum_{i=1}^{k} \frac{t_i^2}{\lambda_i} \tag{4.30}$$

where λ is a diagonal matrix with eigenvalue (λ_i) of the covariance matrix X in descending order, and $t_i$is the i[th] score. The nearest class to a sample is defined as the class model that results in a minimum distance of the sample *i* to model *j*,

$$\boldsymbol{d_i^j} = \sqrt{(Q_r)^2 + (T_r^2)^2} \tag{4.31}$$

where Q_r is the reduced Q residual and reduced Hotelling's T^2 , given by

$$Q_r = \frac{Q}{Q_{0.95}} \text{ and } \quad T_r^2 = \frac{T^2}{T^2{}_{0.95}}$$

$Q_{0.95}$ is the 95% confidence interval for the model under consideration. The above procedure is repeated for all the j classes [47] and based on the nearest distance a new sample is assigned to a class label.

In this work, SIMCA algorithm is used for classification of edible oils and detection of adulterations in edible oils.

4.7.4. Soft computing classification algorithms

Soft computing algorithms, as opposed to hard computing (precisely stated analytical model), deal with approximate models, and provide solutions to complex real-life problems using the human mind as a role model (nature-inspired systems). Imprecision, uncertainty, vagueness, partial truth, and approximations are tolerated by soft computing algorithms. Soft computing uses fuzzy logic, genetic algorithms, artificial neural networks, machine learning, and expert systems.

In this work, artificial neural networks, support vector machine algorithms, and deep learning algorithms such as convolution neural networks are used for edible oil analysis, described in the following section.

4.7.5. Support Vector Machines (SVM)

Deeply rooted in principles of statistics, ML and optimization, Support Vector Machine (SVM) was first introduced in 1992 by Boser, Guyon and Vapnik in a Computing Machinery workshop on Computational Learning Theory. SVM is a supervised learning method used in the classification applications that analyzes trends and patterns in the data. SVM was initially used for supervised classification but is now

also used for regression analysis. The goal of the support vector machine algorithm is to find a hyperplane in n-dimensional space (n is the number of features) that classifies the data points. There are many possible hyperplanes for separating the clusters, but the one with the greatest marginal distance, i.e., the greatest distance between data points of different classes, is chosen.

Support vectors are data points that are closer and opposite to the hyper plane and influence the position and orientation of the hyper plane as shown in the Figure 4.13. The input data is discriminated by a hyper plane that passes through this support vector. Using these support vectors, the classifier maximizes the margin [48].

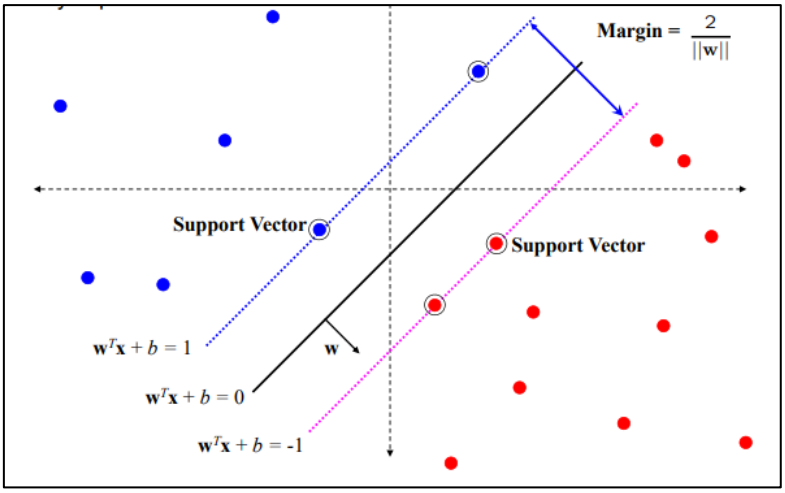


Figure 4.13 Support vector machines (SVM) Method

Hinge loss is used for “maximum -margin” classification, it is a loss function used to maximize the margin while training SVM classifiers. The hinge loss of the prediction y is defined for an intended output t = ±1 and a classifier score y as

$$L(y) = max\{0,\ \ 1 - t.y\}. \tag{4.32}$$

It is important to note that y should be the "raw" output of the classifier's decision function, rather than the predicted class label. For example, in linear SVM equation, $y = \boldsymbol{w}.x + b$,(w and b) are the parameters of hyperplane and x is the input variable.

When the input data is at the boundary, the hinge loss is one. When the data point lies on the other side the hinge loss is larger than one. In our work, Linear support vector machines model is trained for the classification of edible oil samples and classification of adulterated and non- adulterated edible oil samples.

4.7.6. Convolution neural networks (CNN)

Yann LeCun et al. first proposed the concept of convolution neural networks (CNN) in 1998[49]. CNN, a multi-layered feedforward hierarchical network is a subset of deep neural networks in which each layer employs a bank of convolution kernels which are widely used for computer vision tasks. CNN's ability to exploit spatial or temporal correlation in data is one of its most attractive aspects. CNN's topology is divided into multiple learning stages, each of which consists of a combination of convolution layers, nonlinear processing units, and sub-sampling layers [50]. CNNs are among the best learning algorithms for understanding image content, and they have demonstrated exceptional performance in numerous applications like image segmentation, classification, object detection and localization tasks [51], [52]. CNN has notable characteristics such as hierarchical learning, automatic feature extraction, multitasking, and weight sharing. In general, CNN architecture is comprised of three major layers.

1. Convolution layer
2. Pooling/Sub sampling layer,
3. Dense layer or fully connected layer

A convolution layer is a fundamental component in CNN architecture that performs feature extraction from locally correlated data points, with a combination of linear operations such as convolution and nonlinear operations with the help of activation function. In the convolution layer, the input image is convolved with a learnable feature extraction kernel. This process can be formulated as

$$map_{i,j}^{x,y} = f\left(\sum_m \sum_{h=0}^{H_i-1} \sum_{w=0}^{W_j-1} k_{i,j}^{h,w}\, map_{(i-1,m)}^{(x+h),(y+w)} + b_{i,j}\right) \tag{4.33}$$

where $k_{i,j}^{h,w}$ is the value at position (h, w) of kernel connected to the m^{th} feature in $(i-1)^{th}$ location. H_i and $W_i$are the height and width of the kernel, $b_{i,j}$ is the bias of jth feature in $(i-1)^{th}$ layer. Because of the weight sharing capability of convolutional operations, different sets of features within an image can be extracted by sliding kernel with the same set of weights on the image, making CNN parameters more efficient than fully connected networks. It reduces the output feature map's height and width in comparison to the input image, and the initial layers extract low-level image features as shown in Figure 4.14.

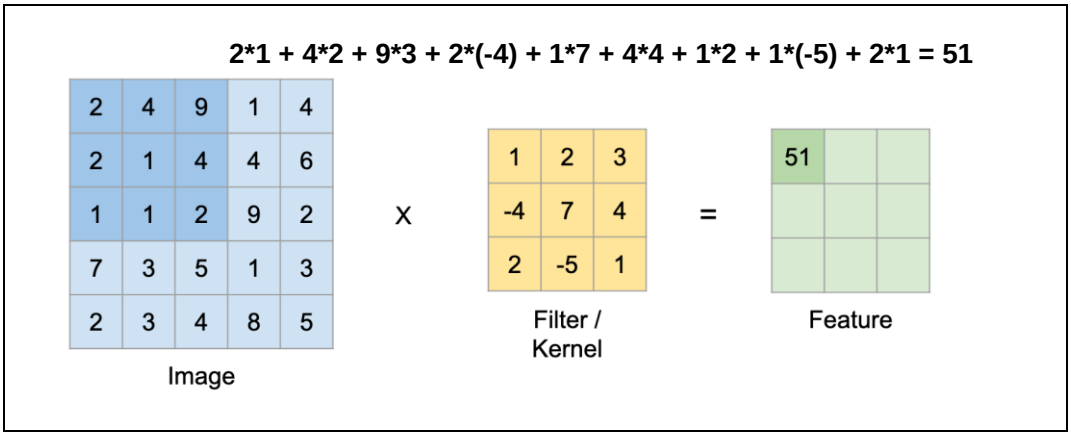


Figure 4.14 Convolution operation example with kernel size 3x3

The hyper parameters are the configuration parameters of a model which control the training process. The hyper parameters of CNN networks are given in Table 4.1 and explained in following sections

Table 4.1 Hyperparameters of CNN

Hyper parameter	Description
Kernel/Filter Size	Kernel size at each convolution layer
Kernel/Filter count	Number of kernels at convolution layer
Stride	The amount by which we slide filter in horizontal and vertical direction when performing a convolution operation
Padding	Hyperparameters used to preserve size and not to lose information of the training data
Epoch	Number of learning iterations
Learning rate	Amount of change in weight that is updated during training
Layer depth	Number of layers constituting an entire network
Batch size	Group size to divide the training data into several groups
Neuron count	Number of nodes in a fully connected layer
Loss function	Function to calculate error/loss
Activation function	Activation function at each node (Relu, sigmoid, softmax)

Padding, which is the process of adding zeros to the input image, is a technique for addressing the issue of keeping the image the same size after the convolution operation. Modern CNN architectures usually employ zero padding to retain dimensions to apply more layers so that more and better features are extracted.

The distance between a successive kernel position is called a stride. The convolution layer's output is passed through a nonlinear smooth activation function such as sigmoid, tanh, rectified linear unit (ReLU) [53]. This method not only assists in abstraction

learning but also embeds non-linearity in the feature space. Figure 4.15 shows some activation functions that are used in CNN.

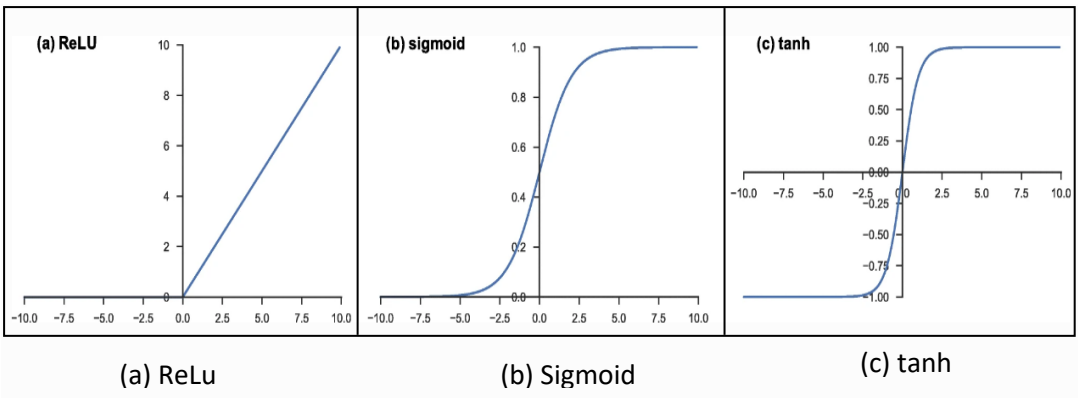


Figure 4.15 Activation functions used in CNN architecture

Following the convolution layer is a pooling layer (down sampling layer), which reduces the dimensionality of feature maps, aids in summarizing the results, and also makes the input insensitive to geometrical distortions [54], [55]. There are two types of pooling operation which are most used: average pooling and max pooling. For a 4x4 input matrix, it can be divided in to four 2x2 sub matrices, max pooling operation (2x2 size) consider the maximum value in each sub sample. Similarly, the average sampling method considers the average of elements in each subsample. The methodology of the max and average pooling operation is depicted in Figure 4.16. Apart from average and max pooling, there are several other pooling operations, including spatial global max pooling, global average pooling, pyramid pooling [56] stochastic pooling [57], and def-pooling [58].

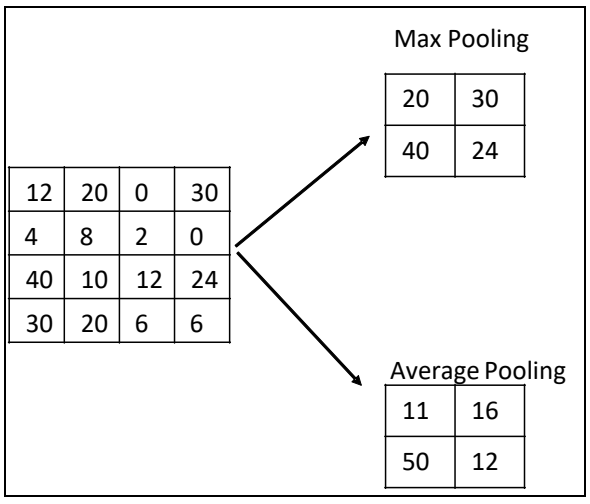


Figure 4.16 An example for max pooling and average pooling

A fully connected layer or dense layer is the same as a traditional neural network. The output maps of the last convolution layer or pooling layer are arranged into vectors, acting as the inputs to the first fully connected layer. But the fully connected layers have a risk of overfitting which can be reduced by using dropout method as a regularize which makes some elements of fully connected layer to zero and eliminates the overfitting.

Another technique to avoid overfitting is to replace the flatten layer in the CNN architecture with global max pooling or global average pooling layers. Global pooling creates one extracted feature for each relevant category of the classification model from the last convolution layer. In global averaging pooling layer, the average of each feature map from the convolution layer is computed and fed to the softmax layer, rather than building a fully connected layer on the top of feature maps. One benefit of global average pooling over fully connected layers is that it maintains correspondences between extracted features and classes, giving it more original to the convolution structure. Another benefit of global average pooling is that there are no parameters to optimise, therefore overfitting is prevented at this layer. Because global average pooling sums up the spatial information, it is more resistant to input spatial translations. We can think of global average pooling as a structural regularizer that drives feature maps to the corresponding classes. In global max pooling layer, the maximum of each feature map is considered and fed to the SoftMax layer.

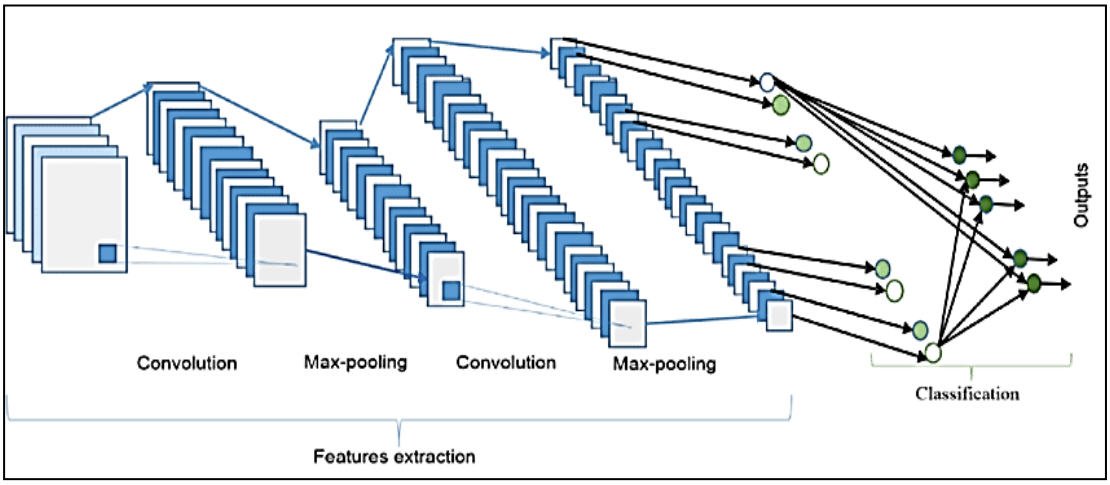


Figure 4.17 Architecture of 2D convolution neural network

The traditional deep CNN presented in the preceding section are intended to work on 2D data such as images and videos. Therefore, they are commonly referred to as 2D CNNs and the complete architecture is shown in Figure 4.17. As an alternative, 1D

Convolutional Neural Networks (1D CNNs) are a modified version of 2D CNNs that were recently developed [59][60], [61]. 1D CNNs have quickly attained state-of-the-art efficiency standards in a variety of applications such as customized biomedical data classification and early diagnosis, structural health monitoring, failure detection and identification in power electronics, and electrical motor troubleshooting. The architecture of 1-D CNN is shown in Figure 4.18.

The computational complexities of 1D and 2D convolutions differ significantly, i.e., an image with NxN dimensions convolved with KxK kernel will have a higher computational complexity whereas the corresponding 1D convolution (with the same dimensions N and K) will have a less computational complexity [59]. Compact 1D CNNs are well-suited for real-time and low-cost applications due to their low computational requirements, especially on mobile or hand-held devices.

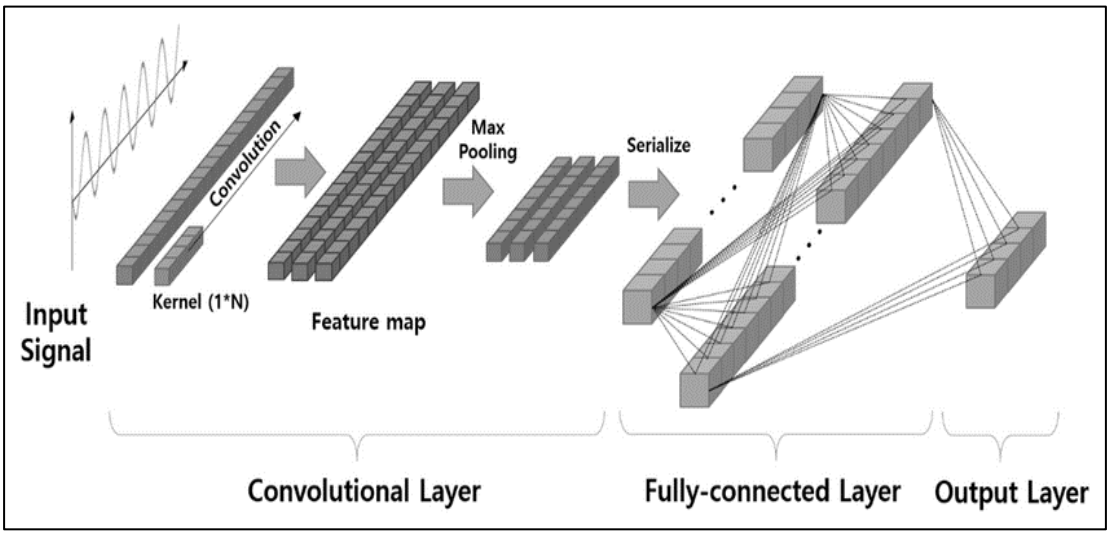


Figure 4.18 1-D convolution neural network architecture

The learning and working process of CNN has two stages: (i) network training (ii) feature extraction and classification. The first part has two tasks, a forward part, and a backward part. In the forward part, the input images are fed through the network to obtain an abstract representation, which will be used to compute the loss (cross-entropy) with respect to the given ground truth labels. Based on the loss function, the backward part computes the gradients of error with respect to each parameter of the network. Then all the parameters are updated in response to the gradients in preparation for the next forward computation cycle. The aim of the training is to minimize the loss and the process of this minimization is known as optimization. There are different types of optimization methods for the CNN learning to converge the desired output.

Optimization methods which are used in this research are explained in the following section

A. Stochastic gradient descent

The SGD algorithm is an adaptation of the Gradient Descent (GD) algorithm that addresses some of the drawbacks of GD algorithm. GD has the drawback of using a lot of memory to load the complete dataset of n-points at once in order to compute derivative of the loss function. The SGD algorithm computes the derivative one point at a time. The process of updating a parameter(θ) for each training example is given by

$$\theta = \theta - \alpha \frac{\partial}{\partial \theta} f(\theta; x_i, y_i) \tag{4.34}$$

where α is the learning rate, f is the cost function. $\frac{\partial}{\partial \theta}$ is the gradient of the cost function at point x_i, y_i. The size of the steps we take to reach a (local) minimum is determined by α.

B. Adaptive gradient

Adagrad is an algorithm for gradient-based optimization, it adjusts the learning rate, performing smaller updates (low learning rates) for parameters with often occurring features and bigger updates (high learning rates) for parameters with rarely features. Adagrad utilises a distinct learning rate for every parameter θ_i at every time step t, if g_t denotes the gradient at time step t, $g_{t,i}$ is the gradient of the cost function with respect to the θ_i and time step t is given by

$$g_{t,i} = \frac{\partial f(\theta_{t,i})}{\partial \theta} \tag{4.35}$$

Adagrad updates the learning rate(α) at each time step t, for every parameter θ_i based on the earlier gradient computed,

$$\theta_{t+1,i} = \theta_{t,i} - g_{t,i} \left(\frac{\alpha}{\sqrt{G_{t,ii} + \varepsilon}} \right) \tag{4.36}$$

Where G is the diagonal matrix of each diagonal elements (i,i) are the sum of squares of the gradient with respect to θ_i at time step t and ε is the smoothing term to avoid division by zero. The advantage of adagrad is that it eliminates the manual tuning of learning rate.

C. RMSprop

RMS prop optimization was proposed to resolve adaptive gradient (adagrad) radically diminishing learning rates. RMSprop adjusts the learning rate by an average of squared gradients that decays exponentially. The update process in PMSprop is given by the following equation

$$\theta_{t+1} = \theta_t - g_t \left(\frac{\alpha}{\sqrt{E[g^2]_t + \varepsilon}} \right) \tag{4.37}$$

Where $E[g^2]_t = 0.9E[g^2]_t + 0.1{g_t}^2$ is the is the running average at time step t.

D. Adaptive moment estimation

Adaptive moment estimation(adam) is another method for adaptive learning rates for each parameter. In addition to storing exponentially decaying average of past squared gradients (v_t), it also stores the decaying average of gradients (m_t). The decaying gradients and squared gradients are calculated by the equations

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \tag{4.38}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \tag{4.39}$$

Mt and vt are the estimates of mean and uncentered variance of gradients. They adjust for these biases by computing bias-corrected first and second moment estimates, as follows:

$$\widehat{m_t} = \frac{m_t}{1 - \beta_1^t} \tag{4.40}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \tag{4.41}$$

The parameter update by adam optimization is then given by

$$\theta_{t+1} = \theta_t - \widehat{m}_t \left(\frac{\alpha}{\sqrt{\hat{v}_t} + \varepsilon} \right) \tag{4.42}$$

Here the authors propose new hyperparameters and assign default values of 0.9 for β1, 0.999 for β2, and 10−8 for ϵ. After the model is trained, the trained network can be used to extract in-depth features and classify unknown images [62], [63]. In this work both 1D CNN and 2D-CNN are used for the classification of different types of edible oils and the identification of adulterated edible oils.

4.7.7. Performance indices of classification algorithms

Various types of performance measures can be used to evaluate the developed classification algorithms [64], [65]. The following are the key performance metrics:

(a) Confusion matrix
(b) Specificity
(c) Sensitivity(recall)
(d) Accuracy
(e) F-measure
(f) Area under the curve (ROC)

The following section is dedicated to explaining the above-mentioned performance measure.

(a) Confusion Matrix

The complete performance of a classification model is given in a matrix form called a confusion matrix., as shown in Figure 4.19. Each entry in a confusion matrix represents the number of predictions made by the model where it classified the classes correctly or incorrectly.

True Positive: The cases in which the predicted class is true and the actual class also true
True Negative: The cases in which the predicted class is not true as the actual class is also not true
False Positive: The cases in which the predicted class is not true while the actual class is true
False Negative: The cases in which the predicted class is true while the actual class is not true

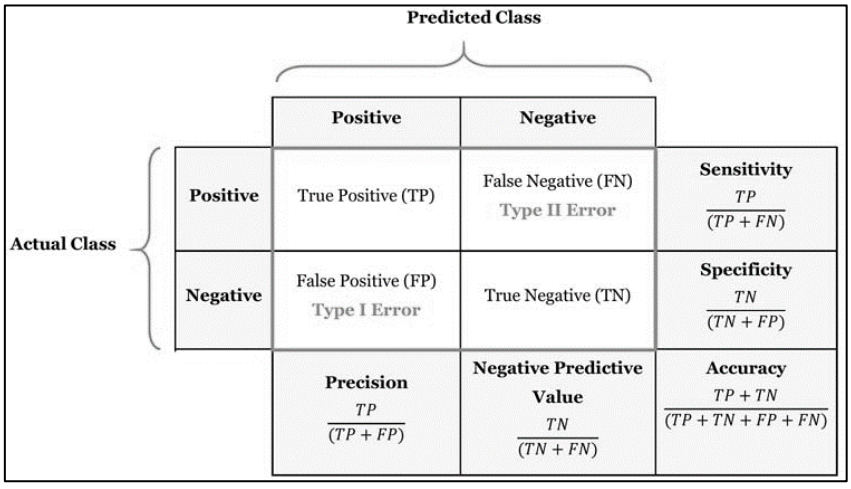


Figure 4.19 Confusion matrix for classification performance

Accuracy from the confusion matrix can be calculated using the following formula.

$$\text{Accuracy} = \frac{\text{True Positive(TP)+True Negative(TN)}}{\text{Total number of samples}} \tag{4.43}$$

Precision is defined as out of all tested positive samples how may are actually true positive.

$$\text{Precision} = \frac{\text{True Positive(TP)}}{\text{True Positive(TP)+False Positive}} \tag{4.44}$$

Sensitivity is defined as Out of all the actual real positive cases, how many were identified as positive.

$$\text{Sensitivity} = \frac{\text{True Positive(TP)}}{\text{True Positive(TP)+False Negative(FN)}} \tag{4.45}$$

Specificity is defined as out of all actual negative cases how many were identified as negative.

$$\text{Specificity} = \frac{\text{True Negative(TN)}}{\text{True Negative(TN)+False Positive(FP)}} \tag{4.46}$$

F1-score is defined as the harmonic mean of Precision and Sensitivity(recall). F1-score reaches its best value 1 and worst value at 0. A more general F1 score that employs a positive real factor β and this β is chosen so that recall is considered times as essential as precision.

$$\text{F}_\beta = \frac{(1+\beta^2)\text{Precision}*\text{Sensitivity}}{\beta*\text{Precision+Sensitivity}} \tag{4.47}$$

Commonly used β values are 0.5, 1 and 2. F1-score is a special case of F_β when β=1. β =0.5 gives more weight to precision than sensitivity. β =2 gives more weight to sensitivity than precision.

(b) Area Under Curve (AUC) and ROC Curve

The region under the receiver operating characteristic (ROC) is an output metric used to test a classification model. A ROC curve is a graph that depicts the output of a binary classification model by plotting the True Positive Rate (TPR) against False Positive Rate (FPR) at all classification thresholds. The AOC value will be 1 if the classification accuracy is 100 %.

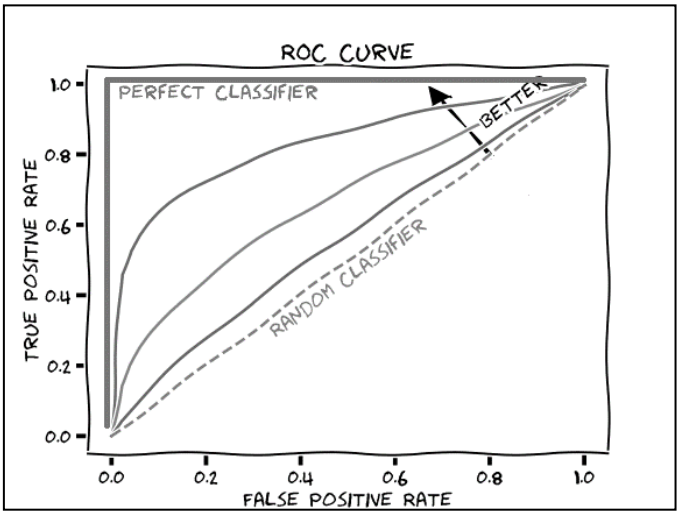


Figure 4.20 Visualization of ROC curve

4.8. Results and discussion of classification of edible oils

This section describes the application of the machine learning algorithms to data collected from e-tongue and NIR spectroscopy experiments for edible classification and ATR spectroscopy experiments for detection of adulteration and quantification in edible oils.

4.8.1. Electronic tongue (Voltammetry)-classification of edible oils

Experimental methodology, sample preparation, and data acquisition procedure of electronic tongue based on voltammetry are explained in chapter 3. Cyclic voltammetry experiment with a single electrode resulted in 1x8000 data points. For a three-electrode combination, the total number of variables is 1x24000. Eight edible oil samples with three working electrodes resulted in a total data matrix of 64x24000. The redox current response for three working electrode configurations with the application of linear sweep voltage is shown in Figure 4.21.

First, the data is pre-treated with the standardization process, which uses the z-score method. Standardization results in a data representation with a mean of zero and unity standard deviation. Figure 4.22 represents the standardized data for three electrodes of the electronic tongue. The data is smoothed using the Savitzy-Golay smoothing filter with a 5-point window to minimize noise and improve the signal-to-noise ratio. Figure 4.23 depicts a voltammogram, which is a plot of the applied excitation potential on a working electrode and the resulting redox current.

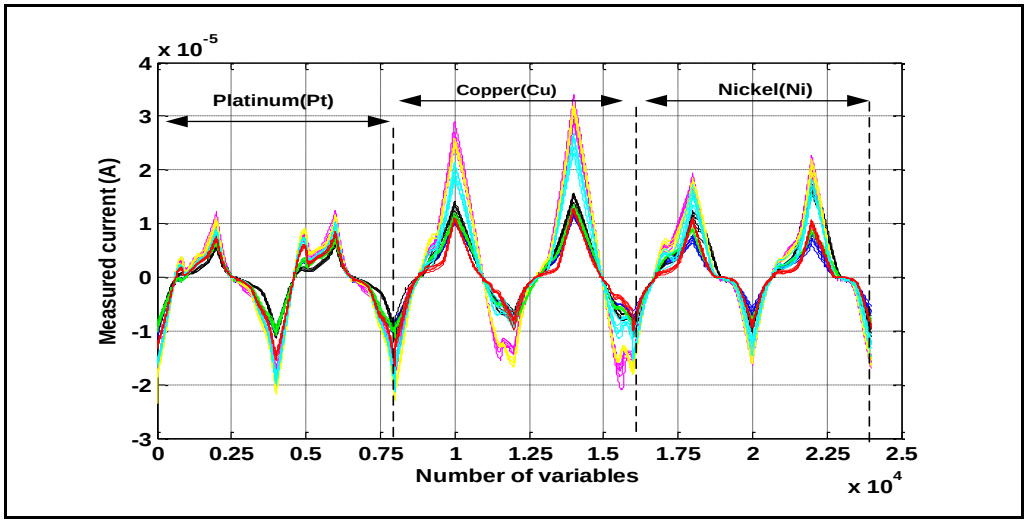


Figure 4.21 Current response of voltammetric electronic tongue with three working electrodes, platinum, copper, and nickel

The PCA is applied to the normalized data to observe the variance in the data and to indicate the data trend in visualizing dimension spaces based on the score plot of the two components. For PCA, the input data size was 64x24000. Figures 4.24 and 4.25 depict a two-dimensional scatter plot of PCA. Figure 4.26 shows a three-dimensional score plot for edible oil analysis using PCA.

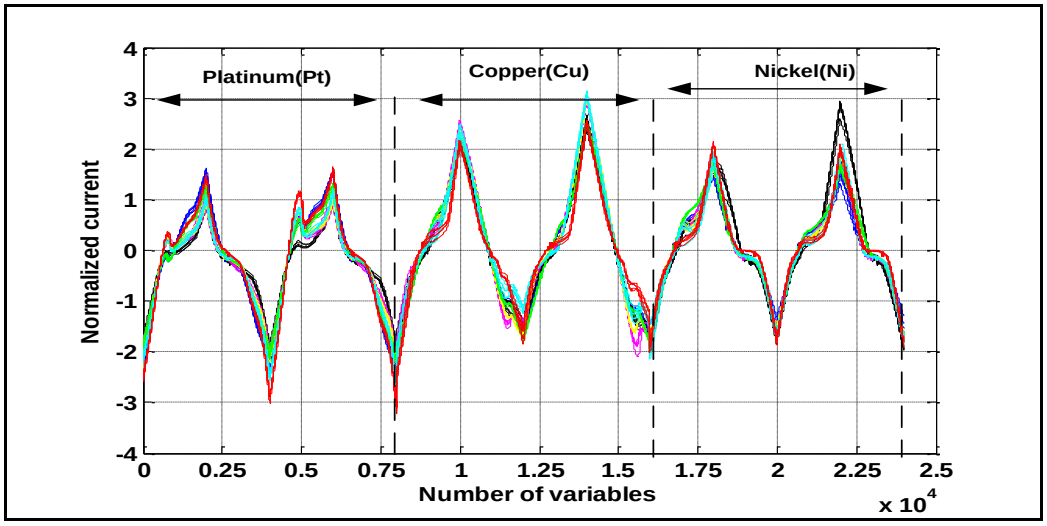


Figure 4.22 Z score normalized current response of voltammetric electronic tongue with three working electrodes, platinum, copper, and nickel

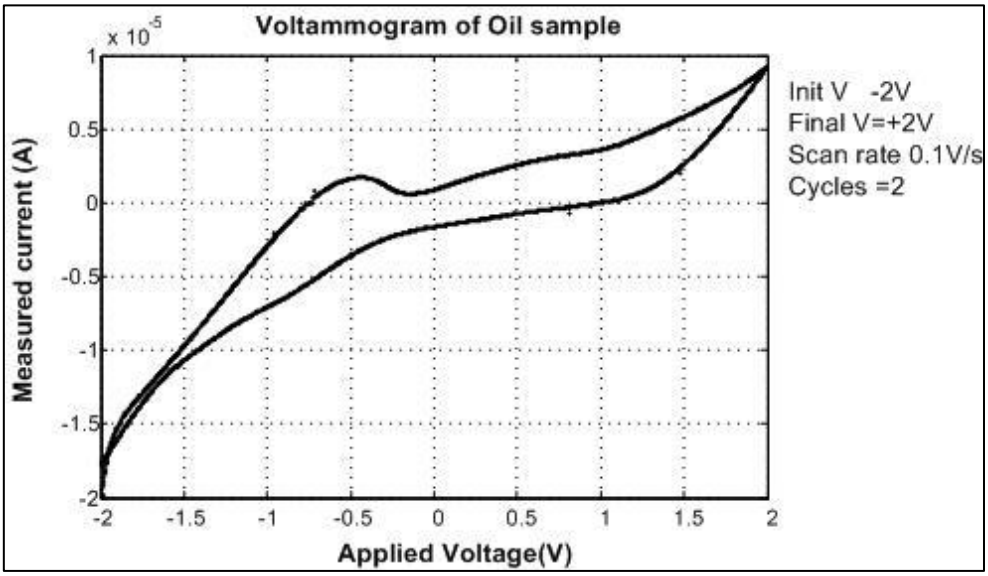


Figure 4.23 Voltammogram of applied voltage and current response of voltammetric electronic tongue for edible oil analysis

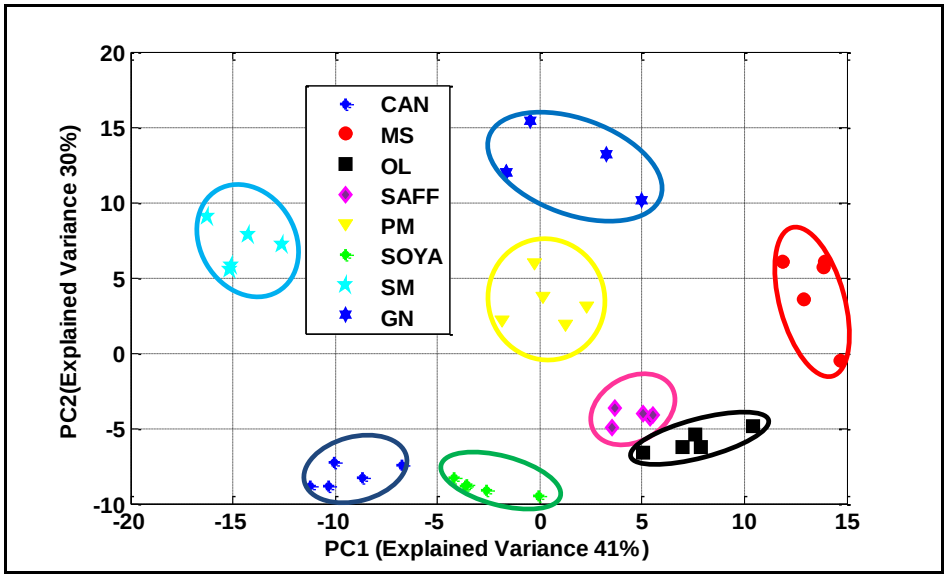


Figure 4.24 Score plot corresponding to PC1 and PC2 of E-tongue data for eight varieties of edible oils

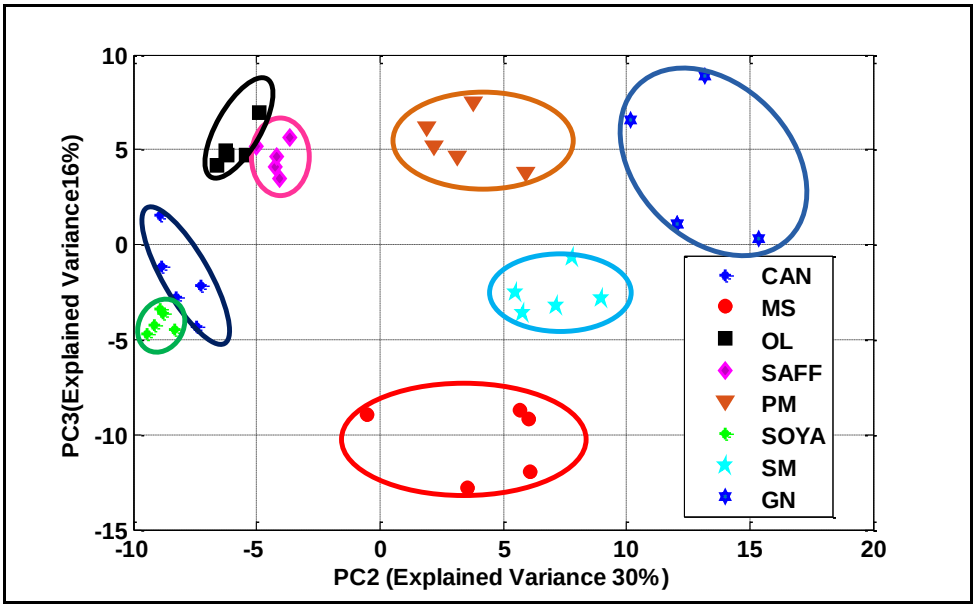


Figure 4.25 Score plot corresponding to PC2 and PC3 of E-tongue data for eight varieties of edible oils

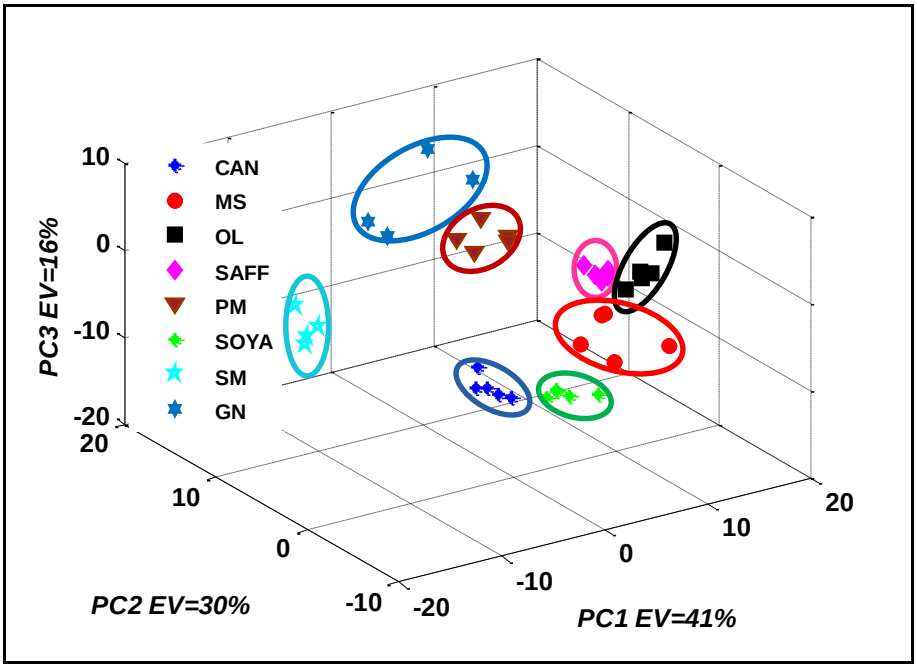


Figure 4.26 Three-dimensional score plot corresponding to PC1, PC2, and PC3 of E-tongue data for eight varieties of edible oils

The fraction of variance explained by a principal component is the ratio of that principal component's variance to the total variance. Add the variances of each principal component and divide by the total variance. Figure 4.27 depicts the explained

variance plot, which provides a comprehensive picture of the number of principal components (PCs) involved in explaining the data variance. The plot clearly shows that a maximum four to five PCs are sufficient to explain the variance in the data, accounting for more than 90% of the total variance.

The first two principal components contribute a total variance of 71%. (PC1 (41%) and PC2 (30%)). The explained variance with the five principal components is above 90% of the data set's total variance. From this, it is clear that the variance in the edible oil data set with total features of 24000 can be explained with the help of only five principal components. Hence the data reduction by PCA is observed in these results. All eight varieties of edible oils are distinguishable from the score PCA plots.

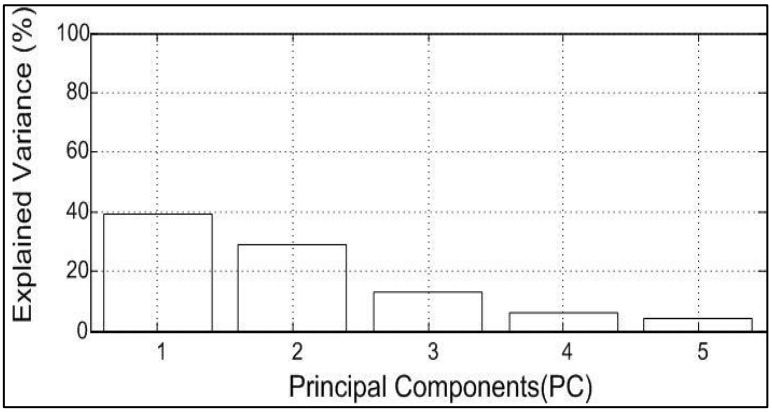
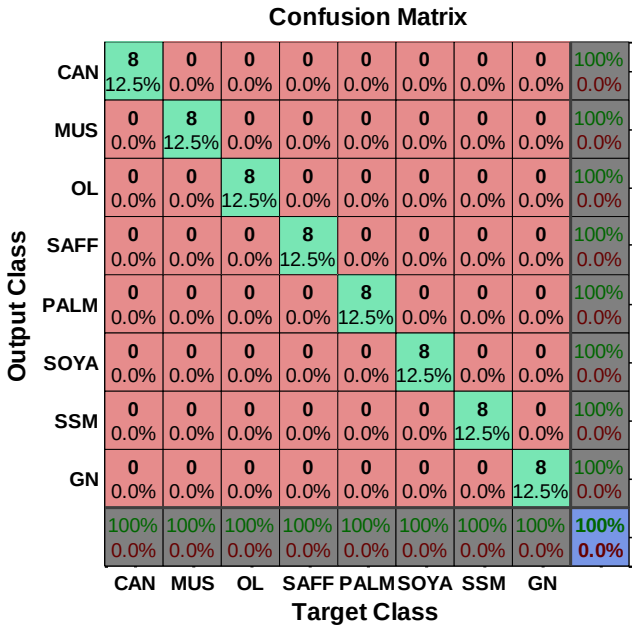


Figure 4.27 Explained variance plot of edible oil data set

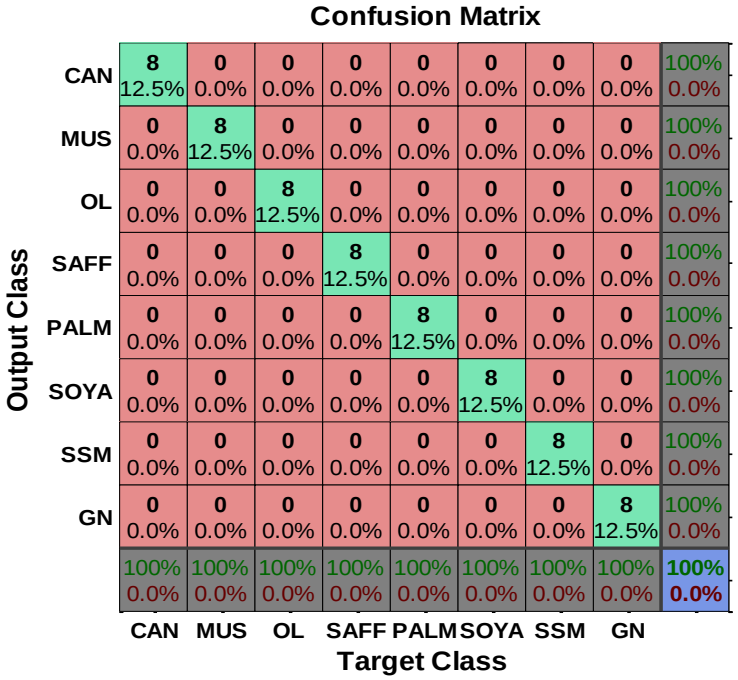
Table 4.2 Confusion matrix for classification of edible oils using PCA-DA



A classifier for the discrimination of eight types of edible oils is modelled using principal component analysis (PCA) followed by a discriminant algorithm (DA). PCA attempts to find the maximum variance in the given data, while DA minimizes variance within the cluster while maximizing variance between clusters.

As described in the previous section, PCA is applied to input data, and the first five principal components (PCs) are considered. These five PCs serve as the DA algorithm input. The number of variables in the data set should be less than the number of samples if the discriminant method is to be used. The DA algorithm generates a transformation function for each class in the training dataset. An unknown test data set is assigned to a class based on the minimum, distance, or maximum probability. The training set in this e-tongue application was 40x24000 in size. When the first five PCs were considered, PCA generated a 40x5 matrix. The testing set was 24x24000 in size. The confusion matrix for classification (discrimination) of edible oils using PCA-DA is shown in Table 4.2. As per the above confusion matrix, the sensitivity, specificity, and accuracy of PCA-DA for the classification of edible oils is 100 %.

Table 4.3 Confusion matrix for classification of edible oils using PLS-DA



For the classification of edible oils using e-tongue data, the partial least square discriminant analysis (PLS-DA) algorithm is also modelled. The PLS-DA model computes a weight matrix of size 24000x5, considering the five components and a

loading matrix of size 24000x5. The response matrix contains categorical data with a value of {0 and 1}. PLS-DA classification is based on predicted Y variable thresholding. If the predicted value is above 0, a corresponding object is considered as a member of a class, and if not, it is not a member of that class. The confusion matrix for the PLS-DA classification of edible oils is shown in Table 4.3. PLS-DA has a 100% sensitivity, specificity, and accuracy for the classification of edible oils.

PCA model serves as the foundation for the SIMCA modeling. PCA is calculated for each class in the training data set, and appropriate principal components are chosen for class modeling. Four principal components from each class have been used for SIMCA model. Using CAMO Unscrambler software SIMCA model is implemented on the data. The distance between the PCA models is calculated to define a threshold for a class assignment. Table 4.4 demonstrates the calculated distances between the PCA models. The classification table for the SIMCA algorithm as shown in Table 4.5 shows every sample is correctly classified to its actual class, (100% accuracy in edible oil classification)

Table 4.4 Distance between the PCA models used in SIMCA classification algorithm

	PCA (Canola)	**PCA (Mustard)**	**PCA (Olive)**	**PCA (Saff)**	**PCA (Palm)**	**PCA (Soya)**	**PCA (Sesame)**	**PCA (Gnut)**
PCA (Canola)	1	794.5981	371.0215	458.412	100.8238	73.22601	406.1363	299.2405
PCA (Mustard)	794.5981	1	561.75	612.2254	129.1936	1080.132	977.1089	220.3224
PCA (Olive)	371.0215	561.75	1	91.37238	53.78019	519.7866	702.4572	216.7497
PCA (Saff)	458.412	612.2254	91.37238	1	34.80748	775.5845	621.5262	160.3955
PCA (Palm)	100.8238	129.1936	53.78019	34.80748	1	97.85661	78.59394	39.44076
PCA (Soya)	73.22601	1080.132	519.7866	775.5845	97.85661	1	735.8607	330.2481
PCA (Sesame)	406.1363	977.1089	702.4572	621.5262	78.59394	735.8607	1	149.8653
PCA (Gnut)	299.2405	220.3224	216.7497	160.3955	39.44076	330.2481	149.8653	1

Table 4.5 Classification of edible oils using SIMCA

Sample Class membership (25%)	PCA (Canola)	PCA (Mustard)	PCA (Olive)	PCA (Saff)	PCA (Palm)	PCA (Soya)	PCA (Sesame)	PCA (Gnut)
Canola	*							
Canola	*							
Canola	*							
Canola	*							
Canola	*							
Mustard		*						
Mustard		*						
Mustard		*						
Mustard		*						
Mustard		*						
Olive			*					
Olive			*					
Olive			*					
Olive			*					
Olive			*					
Saff				*				
Saff				*				
Saff				*				
Saff				*				
Saff				*				
Palm					*			
Palm					*			
Palm					*			
Palm					*			
Palm					*			
Soya						*		
Soya						*		
Soya						*		
Soya						*		
Soya						*		
Sesame							*	
Sesame							*	
Sesame							*	
Sesame							*	
Sesame							*	
Gnut								*
Gnut								*
Gnut								*
Gnut								*
Gnut								*

.

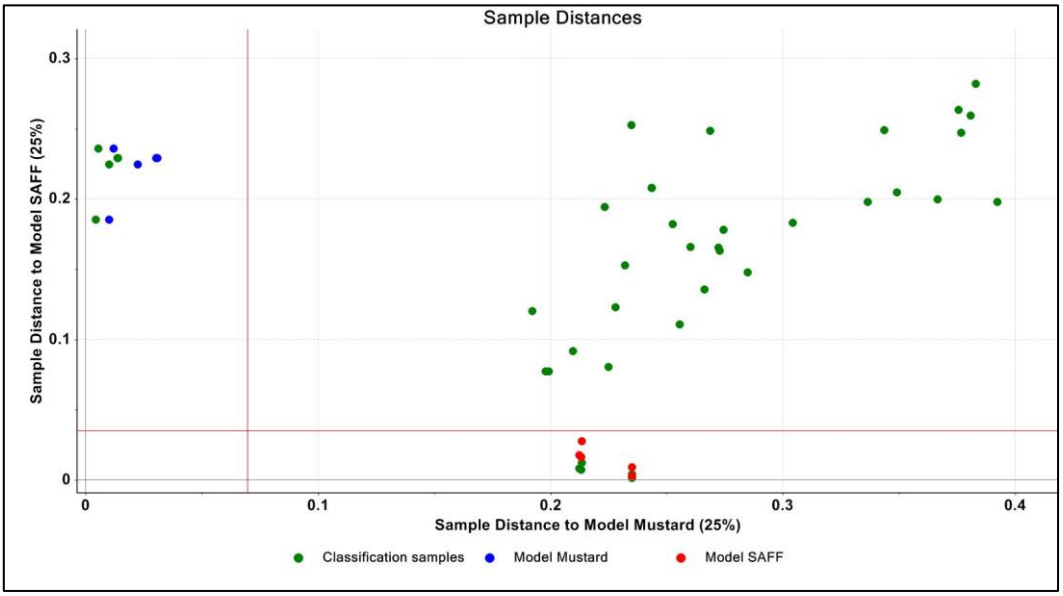


Figure 4.28 Cooman plot for two PCA models using SIMCA algorithm

The support vector machine (SVM) modeling produced good results for the classification of edible oils using electronic tongue data. Each sample in the training data set has 24000 feature vectors; the linear SVM classification algorithm generates hyperplanes for each class with 24000 coefficients and one bias value. Since the training data includes eight edible oil samples, solving eight hyperplane equations with 24000 attribute vectors is a computationally challenging task. SVM applied to 40x24000 input training data yielded 26x24000 support vectors. And the prediction of $20x24000$ test data achieved a classification accuracy of 100 percent.

PCA for feature selection followed by the SVM is trained to accurately discriminate eight edible oils using voltammetric e-tongue data. The PCA is applied to training data for dimensionality reduction to a 40x5 size, by considering the first five PCs. Using this data as input, a linear SVM classifier is trained for 1000 iterations using Python software (sklearn library). SVM model with a coefficient matrix of size 8x5 and a bias matrix of size 8x1 is obtained. Figure 4.29 shows the plot between the first two principal components with the support vectors. Table 4.6 below lists the linear classification equations coefficients and intercept for the hyper planes that separate the support vectors of each class.

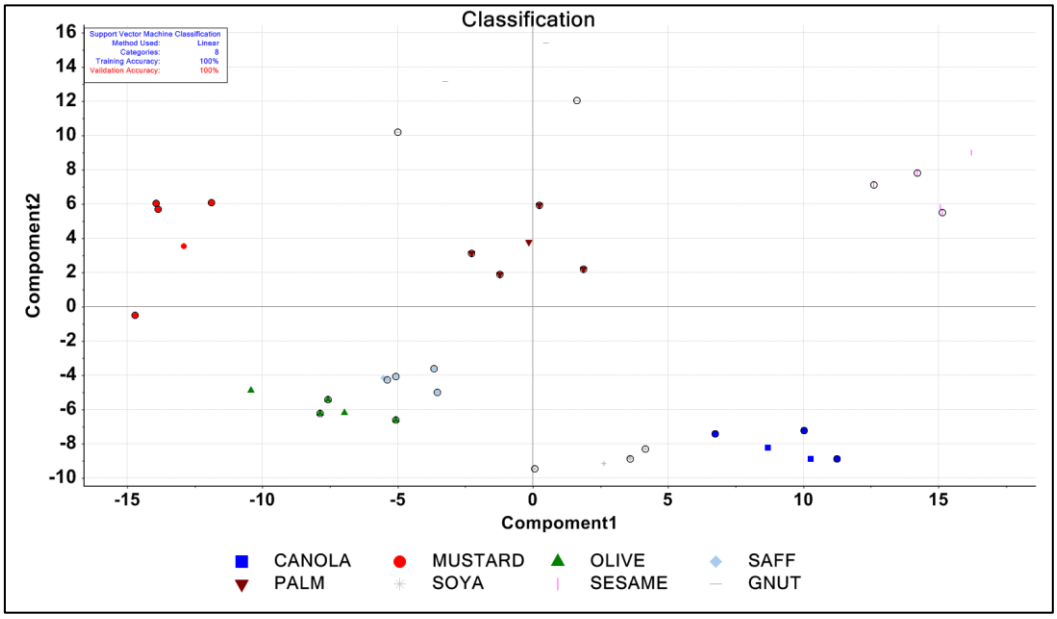


Figure 4.29 SVM classification and support vectors using PCA-SVM

Table 4.6 Linear SVM classification hyperplane's coefficients and intercept values

S. No	Coefficient1	Coefficient1	Coefficient1	Coefficient1	Intercept
1	0.23970844	-0.09178667	-0.10610768	0.40482797	-1,72912363
2	-0.10821971	0.04776101	0.17060967	−0.10838028	−0.97070584
3	0.31691518	−0.30479953	−0.21303589	0.50252771	−2.10594616
4	−0.03102418	−0.09050121	−0.17146761	−0.74549052	−1.94948233
5	0.10136706	0.03385491	−0.36656168	−0.09558325	−1.46670073
6	0.05640343	−0.4341491	0.26851092	−0.56292715	−2.22412899
7	0.10948559	0.04193459	0.06694059	−0.29455079	−0.91300369
8	0.0476876	0.20885354	−0.02118063	0.47125695	−1.5192361

During the testing of a new unknown sample data for a class assignment involving eight hyper plane equations, the outcome of the corresponding hyper plane will be positive (>0), indicating its class, while the other will be negative (<0). The results of PCA-SVM showed 100% classification accuracy

In summary, data analysis with voltammetry data for discrimination of eight types of edible oils revealed that the PCA-DA, PLS-DA, SIMCA, and Linear SVM yielded 100% sensitivity, specificity, classification accuracy, and F1-score. However, the results of PCA-SVM are found to be computationally simple.

4.8.2. Electronic tongue (EIS)-classification of edible oils

The experimental methodology for the EIS experiment for edible oil classification is explained in the previous chapter 3. The impedance and phase characteristics of the solution in the electrochemical cell are calculated using this experiment. In general, impedance and phase response are used to simulate an equivalent electrical circuit made up of basic electrical components (Resistor, Capacitor, etc). Instead of this equivalent electrical model, a novel method for studying trends in electrochemical response using AI algorithms is presented. AI algorithms were applied to magnitude and phase data from EIS experiments over a wide frequency range (0.01Hz to 30kHz). Figure 4.30 depicts phase data from an EIS experiment for edible oils. PCA is applied to phase data to observe variance and indicate data trends on the score plot of the two PCs. The input data size for PCA was 40x67. Figures 4.31 and 4.32 show a two-dimensional scatter plot of the Principal Components from PCA.

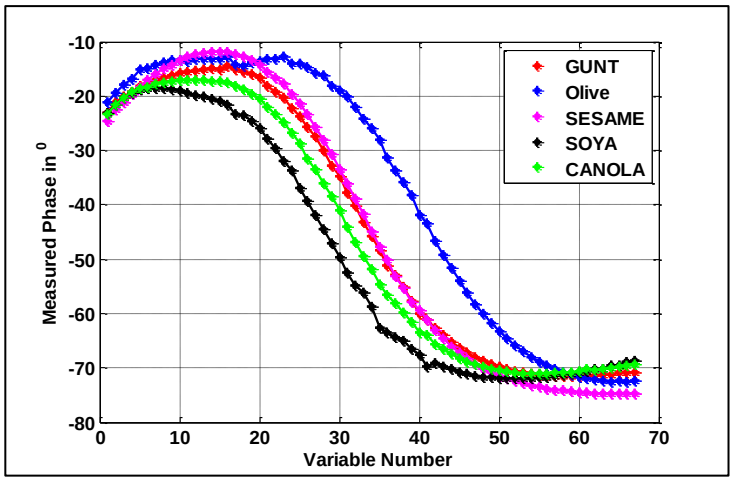


Figure 4.30 Electronic tongue-EIS experiment phase data using a platinum working electrode

From Figure 4.31 and 4.32, it is seen that explained variance of the first three Principal Components corresponding edible oil sample is above 99%. It is obvious from the plot that the utmost three Principal Components are sufficient to explain the variance in the data set of 67 features. The first two principal components contribute a total variance of 99.21%. (PC1 (91.68%) and PC2 (7.53%)). Five types of edible oils are distinguishable from PCA score plots.

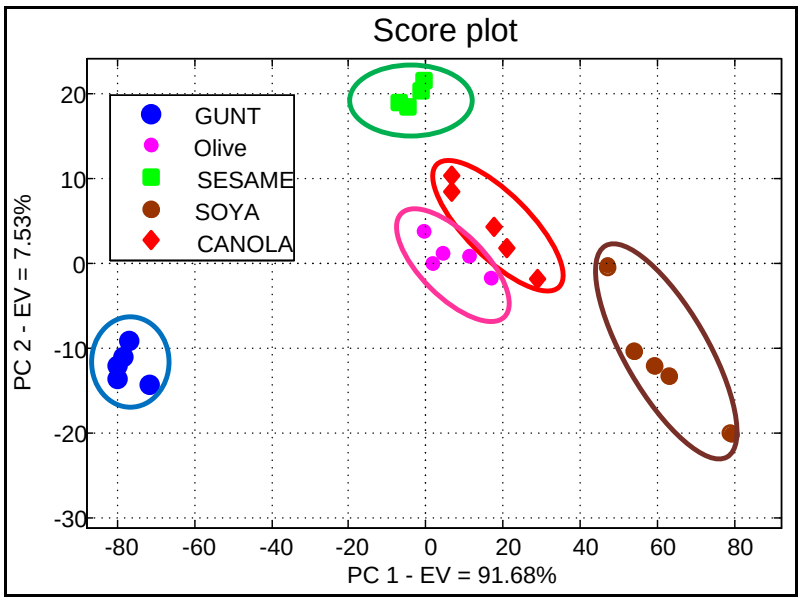


Figure 4.31 PC2 and PC3 of E-tongue data for five varieties of edible oils

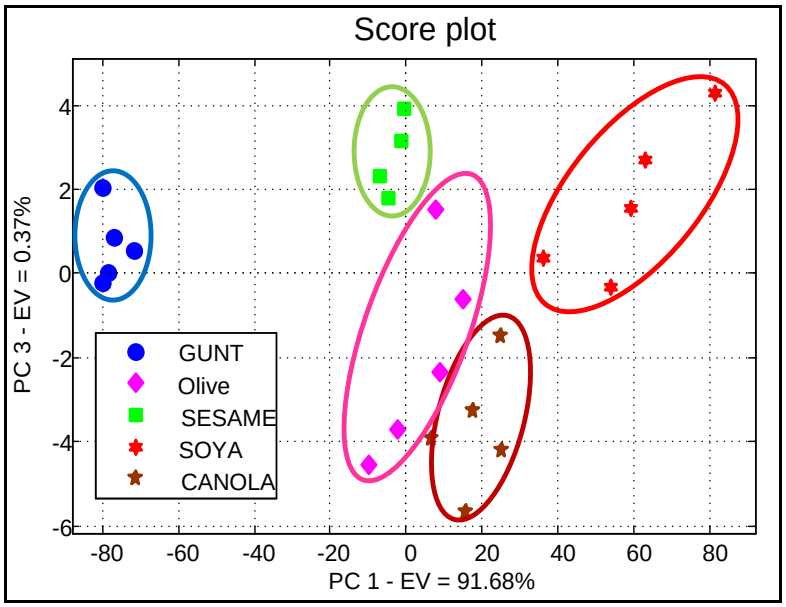


Figure 4.32 Score plot corresponding to PC1 and PC3 of E-tongue data for five varieties of edible oils

SIMCA classification was used for the classification of edible oils by EIS. For the SIMCA model development, PCA was performed on data of an individual class and the optimum number of Principal Components to use in class modelling is selected. The explained variance plot for five PCA models is shown in Figure 4.33.

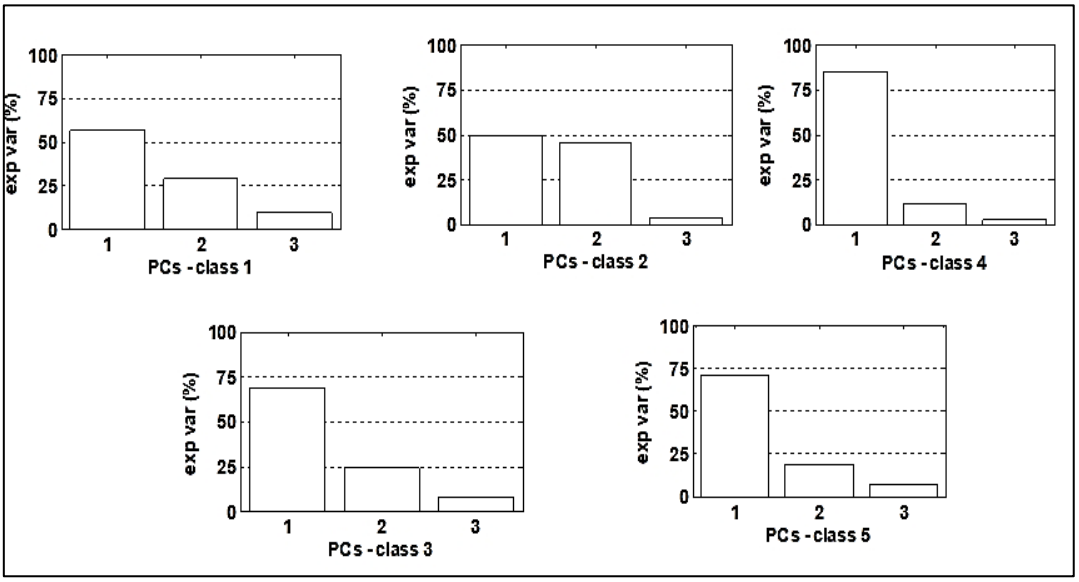


Figure 4.33 Score plot corresponding to PC1 and PC3 of E-tongue data for five varieties of edible oils

Three Principal Components from each sub class PCA model were selected for the SIMCA algorithm development. Data has been divided into training and testing set in (80%-20%) proportions, resulting in a training set of 20 samples and a testing set of 10 samples as shown in Table 4.7. Using the SIMCA algorithm with Classification toolbox in MATLAB, classification accuracy was 100% on the training and testing set. The results are shown in Table 4.8.

Table 4.7 Training and testing data set

Training Set			
Class	**Sample name**	**Identification**	**Number of samples**
Class I	Ground nut	GNUT	4
Class II	Olive	Olive	4
Class III	Sesame	SESAME	4
Class IV	Soya	SOYA	4
Class V	Canola	CANOLA	4
Testing Set			
Class I	Ground nut	GNUT	2
Class II	Olive	Olive	2
Class III	Sesame	SESAME	2
Class IV	Soya	SOYA	2
Class V	Canola	CANOLA	2

Table 4.8 Summary of the results for SIMCA model with EIS experiment (classification of edible oils)

Class	**Sample name**	**Number of factors**	**% of Cumulative variance**	**Classification accuracy**
I	Groundnut	3	87	100%
II	Olive	3	96	100%
III	Sesame	3	94	100%
IV	Soya	3	98	100%
V	Canola	3	97	100%

The classification accuracy of a PLS-DA algorithm developed with EIS experiment training and testing data was also 100%. Based on these results, the EIS method, in combination with chemometric algorithms, successfully classified all five types of edible oils.

4.8.3. NIR spectroscopy-classification of edible oils

The methodology of NIR spectroscopy for edible oil classification is presented in the preceding chapter 3. NIR spectra were recorded over 1351 wavelengths ($1095\ nm\ -2400nm$) for each edible oil. The total size of the NIR experimental data for five different edible oils is 140x1351. This NIR data is divided into two sets: training (100 samples) and testing (40 samples). Table 4.9 below shows the training and testing data set for edible oil analysis using NIR spectroscopy.

Table 4.9 Training and testing data set with NIR spectroscopy

Training Set			
Class	**Sample name**	**Identification**	**Number of samples**
Class I	Sesame	Sesame	20
Class II	Mustard	Mustard	20
Class III	Palm	Palm	20
Class IV	Groundnut	Groundnut	20
Class V	Soya	Soya	20
Testing Set			
Class I	Sesame	Sesame	8
Class II	Mustard	Mustard	8
Class III	Palm	Palm	8
Class IV	Groundnut	Groundnut	8
Class V	Soya	Soya	8

Figure 4.34 depicts the NIR spectra of five different edible oils with wavelengths ranging from 1095 to 2400 nm. The NIR spectra are pre-processed using techniques such as baseline correction and standard normal variate (SNV). The data is smoothed

using the Savitzy-Golay smoothing method with a five-point window and first-degree polynomial.

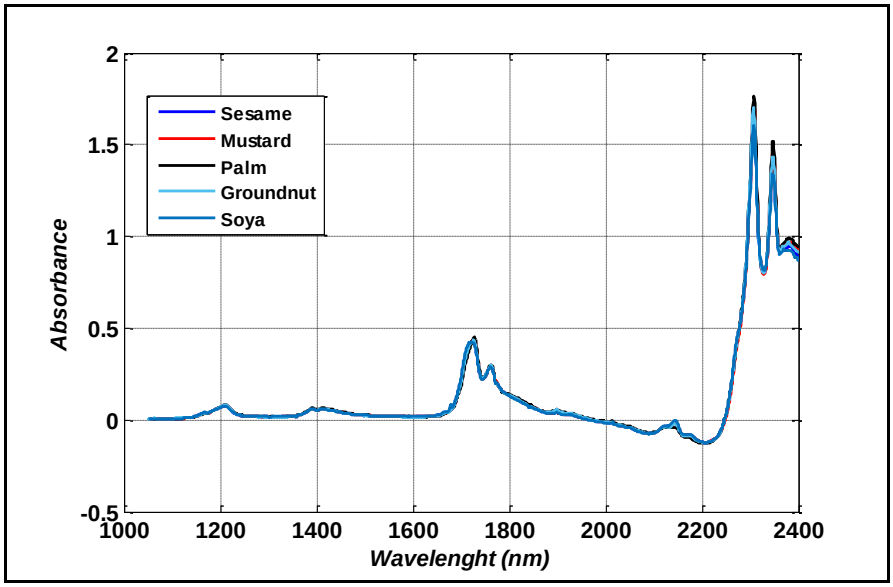


Figure 4.34 NIR spectra of edible oils in the wavelength range 1024 nm to 2400 nm

The PCA method is applied on pre-processed data to extract features, visualize the extracted features in lower-dimensional space, and test the discrimination of data into different clusters. The three-dimensional PCA score plot for visualizing edible oil clusters is shown in Figure 4.35.

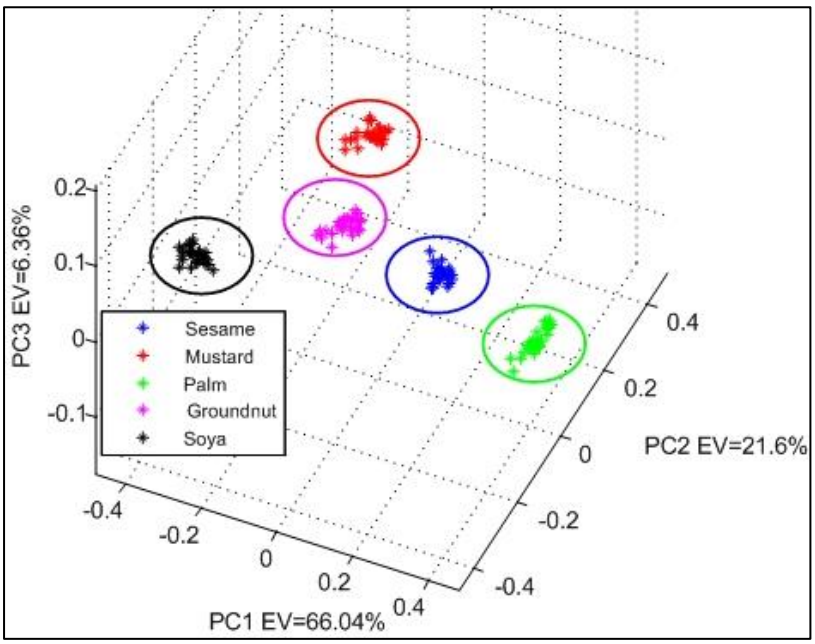


Figure 4.35 Three dimensional PCA score plot for NIR spectroscopy data

In the NIR spectra data, the first three Principal Components account for 94% of the total explained variance (PC1 (66.04 %), PC2 (21.6 %), and PC3 (6.36 %). The PCA score plots clearly distinguish five types of edible oil as separate clusters.

PLS-DA, PCA-DA, and SIMCA classifiers were modelled with the NIR training data set for discriminating edible oils. Classification toolbox version 3.1, in MATLAB, was used to build the classifiers. For selecting optimal components for the algorithms, the 2 fold Venetian blind sampling method (each test set is determined by selecting every i^{th}, $(i+2)^{th}$, $(i+4)^{th}$ ...$(i+2n)^{th}$)) was used. For SIMCA model, four components from PCA for each class were used. PLS-DA, PCADA, and SIMCA classifier models showed a 100% classification accuracy with the testing dataset. The SIMCA was performed with four Principal Components from each PCA model. Table 4.10 shows the classification accuracy of the models. The classification efficiency of developed models was calculated using performance indices like the sensitivity, accuracy, and specificity from the confusion matrix and the Table 4.11 depicts the performance indices for the developed classification models.

Table 4.10 Classification accuracy of developed classification models

Sample name	**Number of tested Samples**	**PCA-DA Classification**	**PLS-DA Classification**	**SIMCA Classification**	**Correct classification**	**Incorrect classification**
Groundnut	28	28	28	28	28 (100%)	0 (0%)
Olive	28	28	28	28	28 (100%)	0 (0%)
Sesame	28	28	28	28	28 (100%)	0 (0%)
Soya	28	28	28	28	28 (100%)	0 (0%)
Canola	28	28	28	28	28 (100%)	0 (0%)

Table 4.11 Confusion matrix for classification of edible oils

Samples	**Sensitivity**	**Precision**	**Specificity**	**False Positive value**	**F-measure**	**Accuracy**
Groundnut	1	1	1	0	1	1
Olive	1	1	1	0	1	1
Sesame	1	1	1	0	1	1
Soya	1	1	1	0	1	1
Canola	1	1	1	0	1	1
Overall	**1.0**	**1.0**	**1.0**	**0.0**	**1.0**	**1.0**

Five different types of edible oils were successfully classified using NIR spectroscopy and chemometrics algorithms. Pre-processing methods like baseline correction and standard normal variate followed by Savitzy-Golay smoothing on NIR data showed a good performance in the classification task. PLS-DA, PCA-DA, and SIMCA showed accurate classification results. The problem with NIR spectroscopy of edible oils was cuvette cleaning. Because of the sticky nature of edible oils, cleaning a 1mm pathlength cuvette was very difficult. As the residual oil stains affect the next sample reading, NIR spectroscopy was not considered for the adulteration detection method. Instead, ATR-based spectroscopy was used for the detection of adulteration in edible oils, and it is explained in the following sections.

4.8.4. Mid infrared spectroscopy with ATR sampling method-Classification of edible oils

Mid infrared spectroscopy with ATR sampling method is a straightforward but accurate analytical method for analyzing liquid and solid samples especially fats and lipids[58]. Experiment methodology is so simple that samples can be placed directly on ATR crystals for spectra acquisition. Cleaning the ATR crystal is also simple. Experimentation also does not involve sample preparation. As a result, this method is used for edible oil classification and adulteration detection.

Nine varieties of edible oil samples were taken for experimentation. The experimental methodology is explained in detail in the previous chapter 3. For a recap of the experimental method, edible oils are directly placed on ATR crystal, ensuring no bubble formation in the oil. The experiment is carried out on five sample readings, and its average is taken as one sample spectrum. A total of 15 such spectra for an edible oil sample were recorded (15x128). Nine varieties of edible oils produce a data matrix of size 105x120. The Savitzy-Golay smoothing filter with the first-degree polynomial fit on nine points windows was used as pre-processing method to smooth the spectrum data.

A. Cluster analysis results

Unsupervised clustering methods are applied to the ATR spectroscopy data. The following section explains the clustering analysis results. MATLAB R2014a commands for cluster analysis methods discussed below is given in Appendix A.1.

i. Hierarchical cluster analysis (HCA)

Hierarchical Cluster Analysis (HCA) was used to identify similarities and differences between the oil samples. The dissimilarity of different clusters was measured using the complete linkage method and described by a squared Euclidian distance. The outcomes were presented in the form of a dendrogram, which depicted the various groups of edible oils.

Figure 4.37 shows the dendrogram plot obtained from the nine varieties of pure oil samples, which produced a very satisfactory separation of edible oils. A total of nine clusters are observed with a cut-off of 5.2 dissimilarity units. The discovery of such natural groupings means that differentiation between edible oils might be possible.

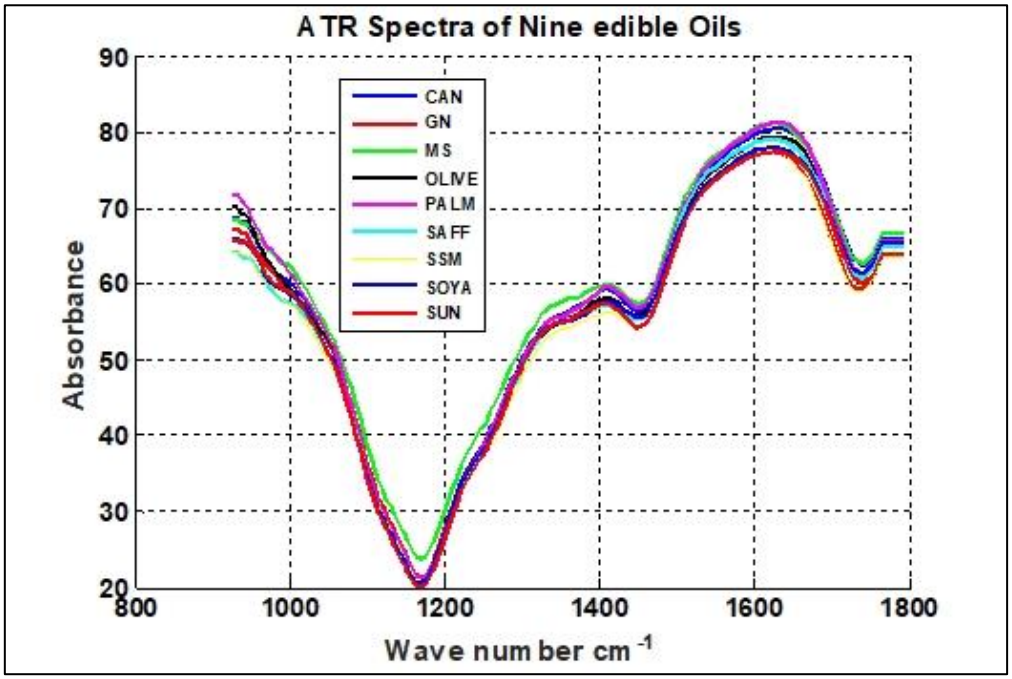


Figure 4.36 ATR spectra of edible oils after pre-processing with savizty Golay smoothing filter

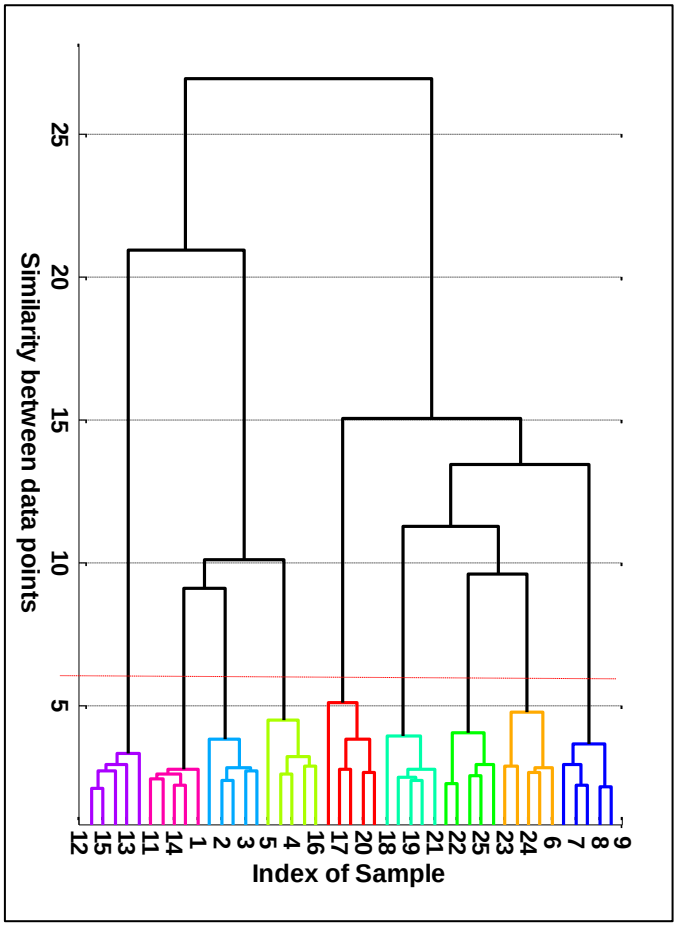


Figure 4.37 Dendrogram of nine types of pure edible oil samples

ii. K-means Clustering

K-means clustering was used to identify similar groups in the edible oil data set. To find the closest clusters among the data points, the squared Euclidian distance was used. The results were presented in the form of a two-dimensional scatter plot between variables 7 and 15 from the dataset, which depicted the various edible oil groups. Figure 4.38 shows the Scatter plot obtained from the nine varieties of pure oil samples, which produced a very satisfactory separation of edible oils. A total of eight clusters were observed. One edible sample group was completely mixed with another, this may be because of similar physicochemical properties of edible oils from this analysis, The discovery of such natural groupings means that differentiation between edible oils may be possible.

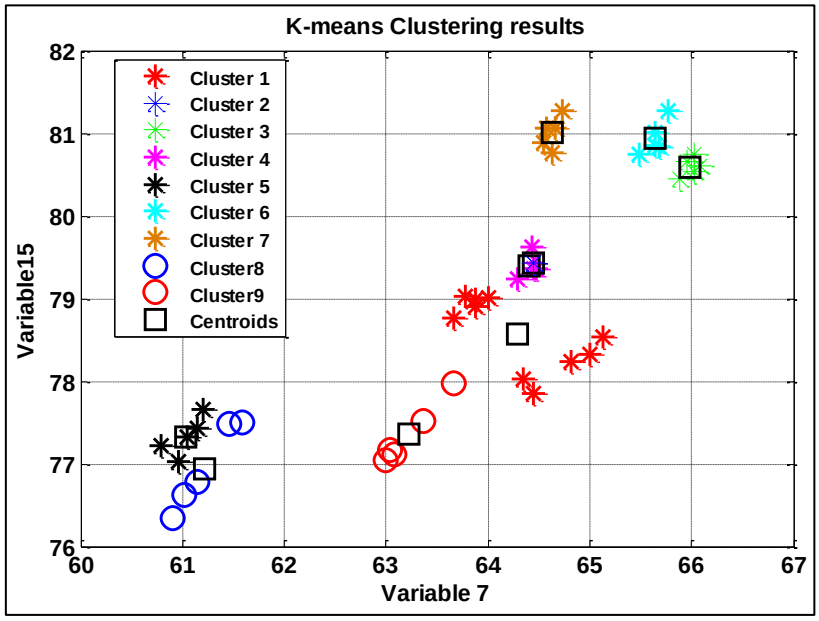


Figure 4.38 k-means clustering results

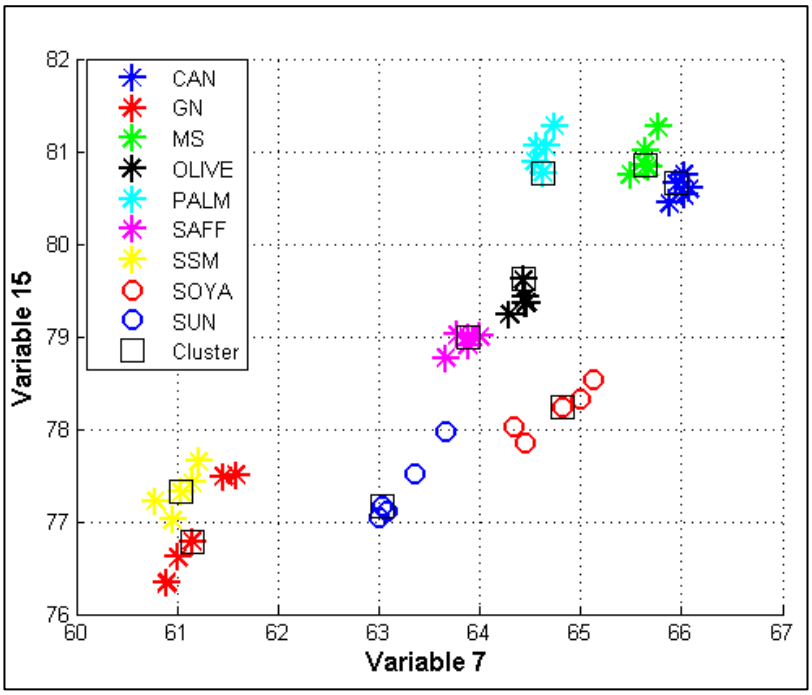


Figure 4.39 Subtractive clustering results

iii. Subtractive clustering

Subtractive clustering is a clustering approach that is based on data point density. This clustering algorithm is implemented in MATLAB. The data clusters are presented

in Figure 4.39. Each of the eight edible oils is clearly separated into its own cluster. In the illustration, the square box indicates the cluster centre for each class.

In summary to the clustering results, Subtractive clustering with a radius of 2 units and hierarchical clustering with a similarity index of 5.2 generated accurate grouping. When K-means clustering with squared Euclidian distance parameters was applied, two clusters were combined.

iv. Principal component analysis

To understand the dissimilarities in the dataset and to extract the features from the spectral data in a lower-dimensional space, PCA was applied to the pre-processed data. A two-dimensional score plot for visualizing the variation among the spectra of edible oils is shown in Figures 4.40 and 4.41. The first three principal components explain total variance of 85% PC1 (72.0%), PC2 (12.0%), PC3 (3%)). Score plots clearly show the nine varieties of edible oils as different clusters.

B. PLS-DA classifier

PLS-DA is used to model a classifier for the discrimination of edible oils using MIR spectroscopy with ATR sampling data. The total data matrix is divided into training, validating sets, and testing sets in (70%,15%, and 15% ratio). Table 4.12 shows the training and testing data information. PLS-DA classifier modelled using classification toolbox 3.1 in MATLAB R2014a. To find optimal components for PLS-DA, auto scaling of data, Venetian blind cross-validation, and Bayes class assignment method was used resulting in six (6) optimal components [78]. PLS-DA showed 100% classification accuracy and it is shown Table 4.13. as a confusion matrix.

Table 4.12 Training Sample and testing samples

Training Set and Testing Set					
Class	**Sample name**	**Identification**	**Number of samples (Training)**	**Number of samples (validation)**	**Number of samples (Testing)**
Class1	Canola	CAN	10	5	5
Class2	Groundnut	GN	10	5	5
Class3	Mustard	MS	10	5	5
Class4	Olive	OLIVE	10	5	5
Class5	Palm	PALM	10	5	5
Class6	Safflower	SAFF	10	5	5
Class7	Sesame	SSM	10	5	5
Class8	Soya	SOYA	10	5	5
Class9	Sunflower	SUN	10	5	5

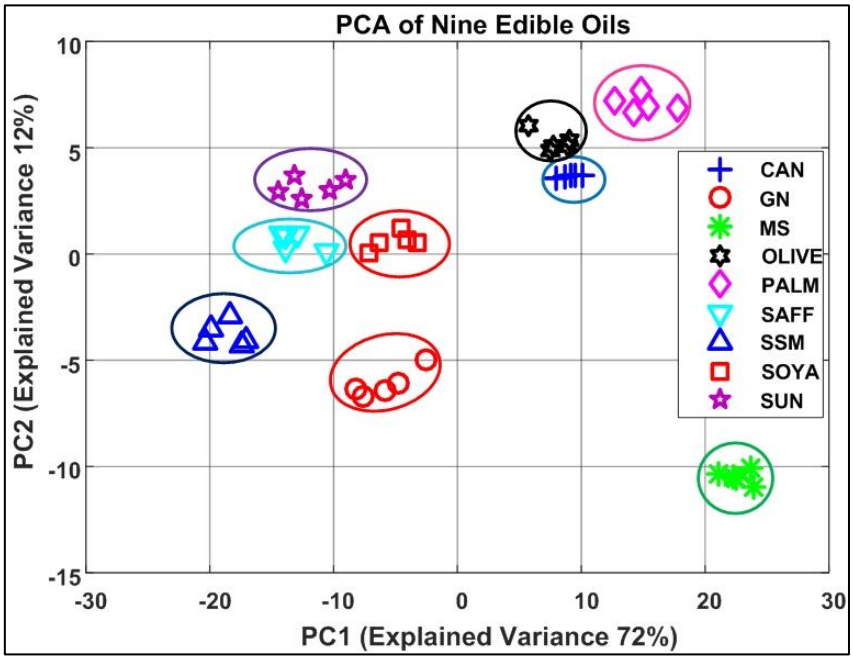


Figure 4.40 Score plot corresponding to PC1 and PC2 of E-tongue data for nine varieties of edible oils

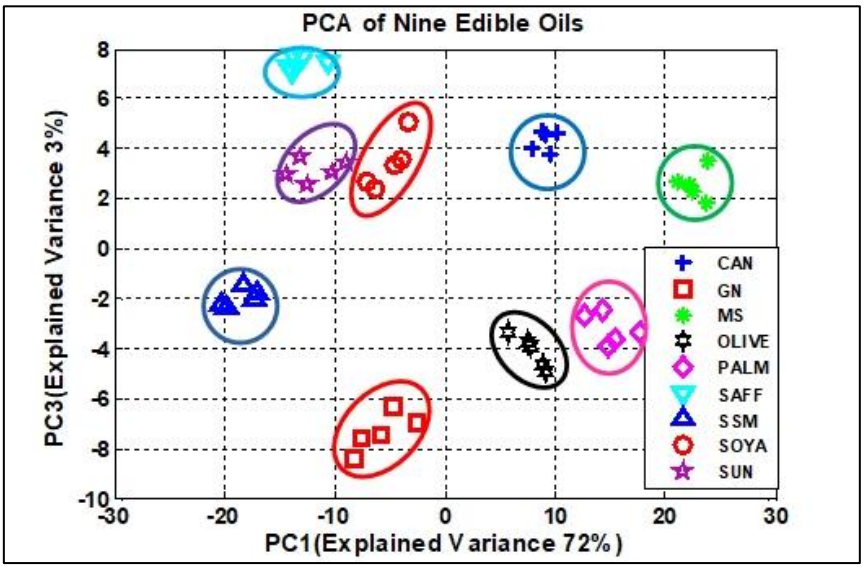


Figure 4.41 Score plot corresponding to PC1 and PC3 of E-tongue data for nine varieties of edible oils

Table 4.13 Confusion matrix for PLS-DA classification

	Canola	Groundnu	Mustard	Olive	Palm	Safflower	Sesame	Soya	Sunflowe
Canola	15	0	0	0	0	0	0	0	0
Groundnut	0	15	0	0	0	0	0	0	0
Mustard	0	0	15	0	0	0	0	0	0
Olive	0	0	0	15	0	0	0	0	0
Palm	0	0	0	0	15	0	0	0	0
Safflower	0	0	0	0	0	15	0	0	0
Sesame	0	0	0	0	0	0	15	0	0
Soya	0	0	0	0	0	0	0	15	0
Sunflower	0	0	0	0	0	0	0	0	15

Table 4.14 Measured performance metrics for PLS-DA classification

Samples	**Sensitivity**	**Precision**	**Specificity**	**False Positive value**	**F1-score**	**Accuracy**
Canola	1	1	1	0	1	1
Groundnut	1	1	1	0	1	1
Mustard	1	1	1	0	1	1
Olive	1	1	1	0	1	1
Palm	1	1	1	0	1	1
Safflower	1	1	1	0	1	1
Sesame	1	1	1	0	1	1
Soya	**1**	**1**	**1**	**0**	**1**	**1**
Sunflower	**1**	**1**	**1**	**0**	**1**	**1**
Overall model parameters	**1.0**	**1.0**	**1.0**	**0.0**	**1.0**	**1.0**

C. Linear SVM classifier

To discriminate the nine varieties of edible oils, features from the PCA were used to model a SVM classifier with a linear kernel. SVM with full spectrum features was also modelled. However, because the support vectors are higher dimensional (128 variables), predicting a new sample for a class assignment becomes computationally challenging. For SVM classifier modelling, reduced feature vectors (principal components) are used. Training data set size (70x128) after feature extraction resulted in a feature vector of 70x5 considering first five principal components. Linear SVM classifier (finding the hyperplanes) developed in Python 3.6 using scikit-learn version 0.24.2 library with the feature vector is used to classify the edible oil samples. Linear SVM model is trained with max iterations of 10000, with squared Hinge loss function.

The hyperplane equations coefficients and intercept for classifying edible oil samples are given in Table 4.16. The sensitivity, specificity and classification accuracy of this model is observed to be 100%. The trained SVM model parameters have been saved and used for the implementation of inference algorithms on an ARM-based platform. This inference implementation is covered in Chapter 5.

Table 4.15 Linear SVM coefficients and intercepts for classification of oils

S No	Coefficient1	Coefficient2	Coefficient3	Coefficient4	Intercept
1	0.08613	-0.18129	-0.31389	0.11392	-2.88724
2	0.00019	0.11536	0.25148	0.13434	-1.55323
3	0.03447	0.11175	-0.05266	-0.04909	-1.0517
4	0.09419	-0.07569	0.23069	0.47705	-1.8727
5	0.05191	-0.05541	0.02216	-0.4720	-1.51718
6	-0.02149	0.03897	-0.23487	-0.18379	-1.53501
7	-0.0997	-0.11633	0.1761	-0.33228	-1.8844
8	-0.02177	0.06957	-0.33472	0.5629	-2.32266
9	-0.09691	-0.18935	0.12611	0.23519	-1.7582

D. CNN based deep neural network

Convolutional neural networks have a wide range of applications in computer vision, object recognition, and classification. A novel 1-Dimensional and 2-Dimensional CNN approach is proposed for edible oil classification. The usage of 1D CNN with NMR spectroscopy is discussed in the literature, but not with Infrared spectroscopy. The spectrum of each edible oil is used as input for a 1D-Convolutional layer in 1D CNN for edible oil classification. The CNN model was developed with Python 3 and the Keras python deep learning API with the TensorFlow backend. The idea of this model is to study the correlation of absorbance of edible oil at one wavelength to absorbance at another wavelength.

The edible oil sample spectrum is of the size 1x128 (captured at 128 frequencies). Min max normalization is applied on this captured data. For 1-D CNN this sequence of normalized 128 variables were used as input. 1-D CNN model with two convolutional layers, two pooling layers and two dense layers has been trained. The architecture of 1D CNN is shown in Figure 4.42. The hyperparameters for CNN algorithms are shown in Table 4.16

Table 4.16 Hyper parameters used in 1-D CNN and 2D CNN

S. No	Hyperparameter	1-D CNN	2-D CNN
1	Input Size	1x128	128x128
2	Kernel Size	1x3,1x2	3x3,2x2
3	Kernel Count	10,10,	10
4	Stride	1	1x1
5	Zero-padding	None	None
6	Learning rate	0.001	0,001
7	Batch Size	1	1
8	Loss Function	Cross entropy	Cross entropy
9	Optimization	Adam	Adam
10	Activation	Relu, Softmax, sigmoid	Relu, Softmax, sigmoid

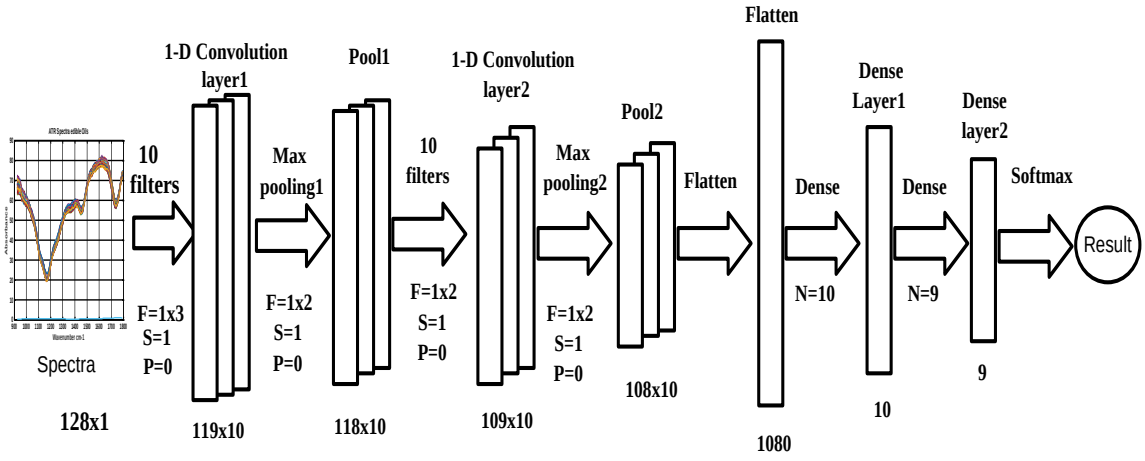


Figure 4.42 Proposed1D CNN sequential model architecture for classification of edible oils

Data is divided into training, validation, and testing data sets in proportions of 70%, 15%, and 15% respectively resulting in 70 training samples 15 validation, and 15 testing samples. The model was trained with 1000 epochs and a batch size of 1. The classification results were 100% accurate demonstrating the feasibility of 1-D CNN algorithms for the classification of edible oils.

Another novel methodology for classifying edible oils was proposed, this time using the 2D CNN deep learning method. A correlation matrix for each edible oil spectrum (X^TX) was calculated, and CNN was applied to the correlation matrix of size (128x128). The CNN algorithm, which consists of two convolution layers, two max pooling layers, and two dense layers, has been developed. In the convolution layer, the Relu activation function was utilised, and in the final dense (fully connected) layer,

softmax activation was used. The CNN method using training data (70x128x128) was successfully implemented, and 100 % classification accuracy was observed. In the training, a batch size of one and the Adam optimization technique were used. Figure 4.43 depicts a summary of the proposed CNN model architecture.

The developed 1-D and 2D CNN model and its each layer kernels, weights and bias parameter were saved in a readable CSV format. These parameters were used in the development of the inference algorithm in C and its implementation on an embedded platform. CNN algorithm for edible oil classification is provided in Appendix 3. The C language and embedded implementation is discussed in the next chapter 5.

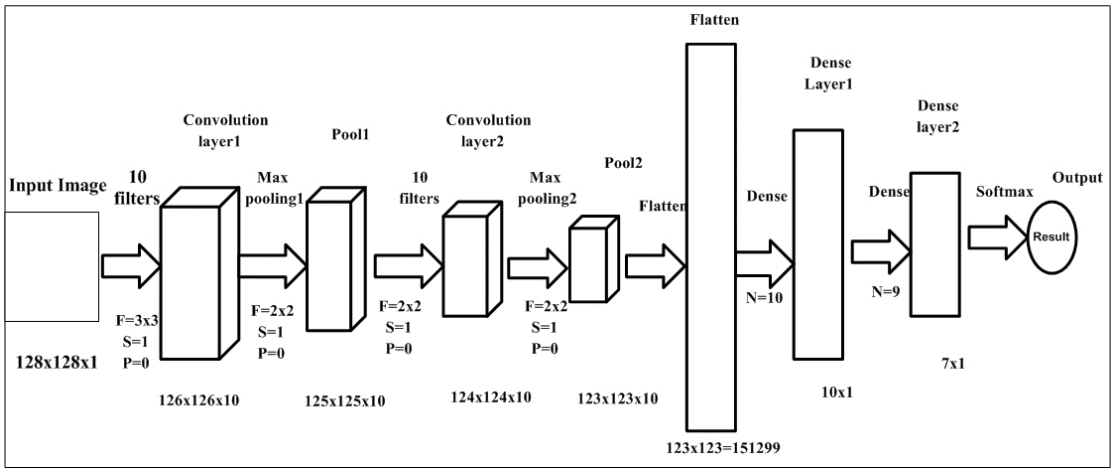


Figure 4.43 Proposed 2D CNN sequential model architecture summary for classification of edible oils

4.9. Qualitative detection of adulteration in edible oils using MIR spectroscopy with ATR sampling

Attenuated total reflection spectroscopy has been used to design an experiment for the detection of adulterations in edible oils. Detection of adulterations using an electronic tongue based on voltammetry and EIS requires the use of harmful chemicals in sample preparation, and the experiments are time-consuming. As discussed, the drawbacks of electronic tongue and NIR spectroscopy alternatively, ATR spectroscopy offers the flexibility of direct contact with samples for spectra acquisition. ATR does not require any sample preparation and suitable for solid and liquid sample analysis. As discussed in the previous chapter lab-made adulterated samples of proportions 5%, 10%, 25%,50%, 75% were used for experimentation. AI models for detection of adulteration (PLS-DA, PCA-DA, SVM 1-D CNN) have been developed and the results have been discussed in the following sections. Three adulteration case studies i.e. (i)

Sunflower oil adulteration with palm oil, (ii)Groundnut oil adulteration with cottonseed oil and (iii) Sesame oil adulteration with cottonseed oil have been presented.

Case 1: Palm oil adulteration in Sunflower oil

From the literature, the functional group associated with -C=O (ester) carbonyl group from ester linkage of triacylglycerol is attributed to the wavenumber of 1745 cm^{-1}. The other functional groups/ bonds in fatty acids, for example -C-H-bending, -C-H-, -C-O-CH2-, =C-H-(cis), and -C-H-(CH2, CH3), have been reported to be attributed for wavenumbers 1097, 1117, 1161, 1377, and1460 cm^{-1} respectively [79], [80], [81]. In the present work, the spectrum observed in the regions of 1786-1680 cm^{-1} could be because of the ester group of the triglycerides present in the edible oils, while the 1490-915 cm^{-1} could be because of fatty acid functional groups. 1400 to 1097 cm^{-1} region is also represented as fingerprint regions [82][83].

The data obtained in the mid-infrared region (1751-900 cm^{-1}) with sunflower oil adulteration with palm oil (in proportions of 5%, 10%, 15%, 25%, 50%, and 75(v/v) %) is divided into three sets of spectra variables corresponding to wavenumbers marked R1 and R2 in Figure 4.44. The third wavenumber region, R3 was chosen based on the contribution of variance by variables from the PCA correlation loading plot. of the entire data set. Table 4.17 shows the selected wavenumber regions for the analysis.

Table 4.17 Spectral regions selected for analysis of adulteration in edible oils

S. No	**Selected Wavenumber range (cm^{-1})**	**Indicator**
1	1492-937	R1
2	1781 to 1635	R2
3	{1717 to1581, 1501 to 1447, 1372 to 1334, 1263 to 937}	R3

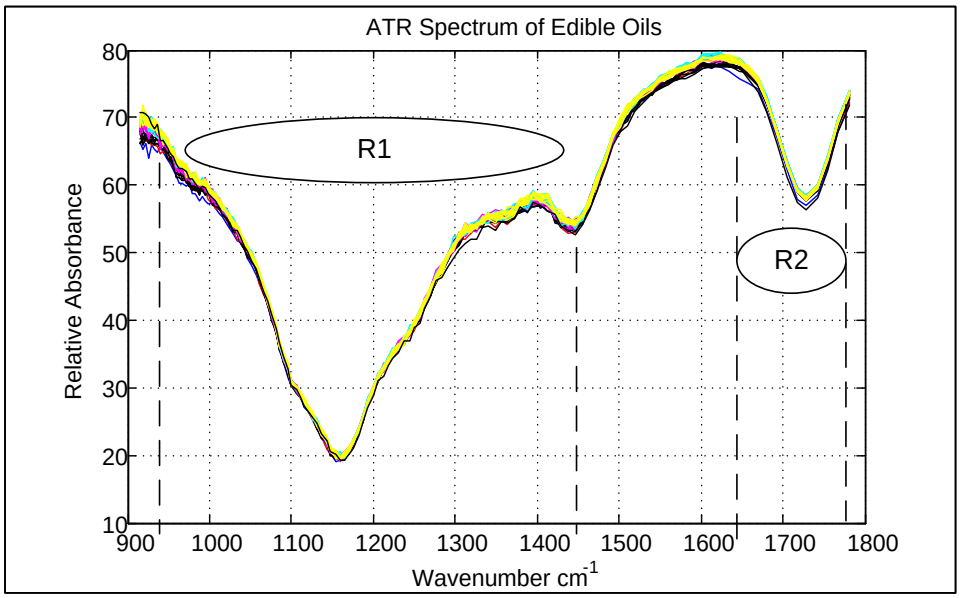


Figure 4.44 ATR spectra of adulterated sunflower oil with palm oil

PCA was applied on the selected data (R1, R2, R3) to understand variance and to visualize data in lower dimensional space. Score plots of PCA on selected wavenumbers are shown in Figures 4.45, 4.46 and 4.47. For R1 (1781 to 1635cm^{-1}), the total explained variance with three principal components (PCs) is above 99% (PC1 (93%.0%), PC2 (6.0%), PC3 (0.6%)). Similarly, for R2 (1492-937 cm^{-1}) the explained variance with three PCs is 97% (PC1 (81%.0%), PC2 (16.3%), PC3 (1%)). For the R3 (1717-1581, 1501-1447,1372-1334,1263-937 cm^{-1}) the explained variance is 95% (PC1 (75.1%.0%), PC2 (17.72.0%), PC3 (2.35%)). From the three-dimensional score plots, it is observed that the adulterated samples are clearly clustered.

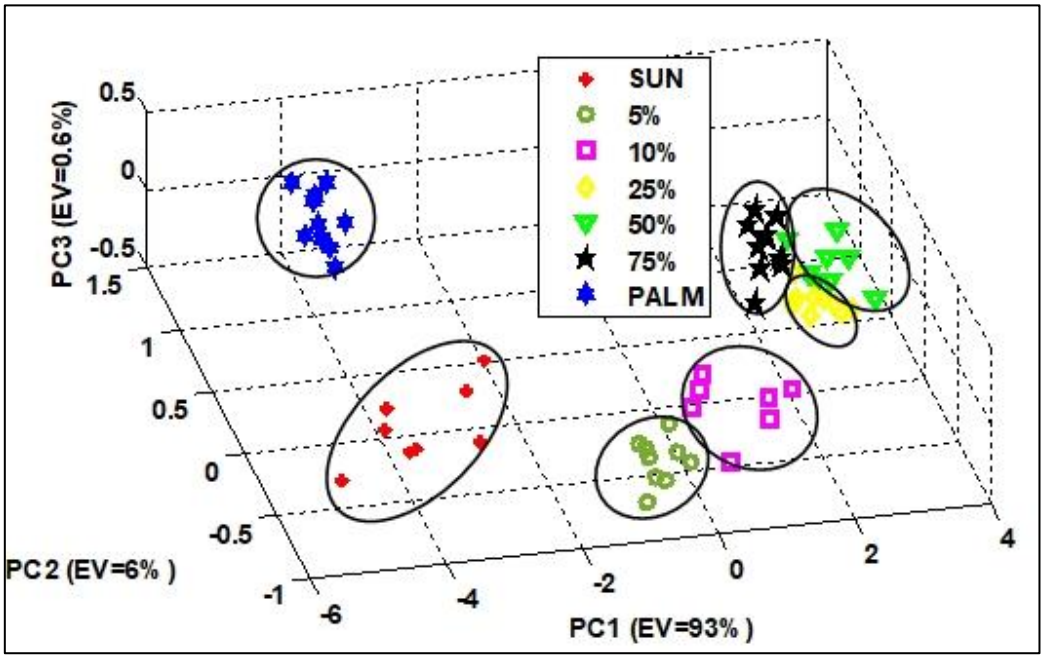


Figure 4.45 Three dimensional PCA score plot for sunflower oil adulteration with palm oil in R1 (1492-937 cm^{-1})

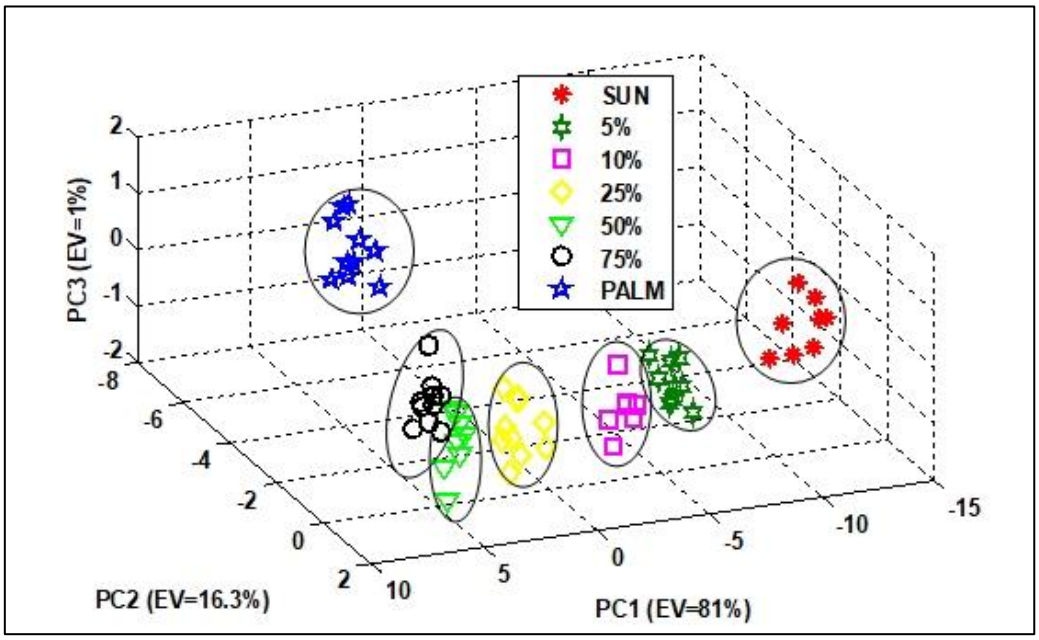


Figure 4.46 Three dimensional PCA score plot for sunflower oil adulteration with palm oil in the spectral range of R2 (1781 to 1635cm^{-1})

For the classification of adulterated and unadulterated oil samples, classification algorithms such as PCA-DA and PLS-DA, as well as SVM algorithms were developed. Training data consists of 70 samples $(70x128)$, 10 samples for each class. PCA-DA and PLS-DA algorithms

were developed by mean centering of the data and considering 5 components. For the cross-validation, the Venetian blind method with 4 cross-validation groups was used.

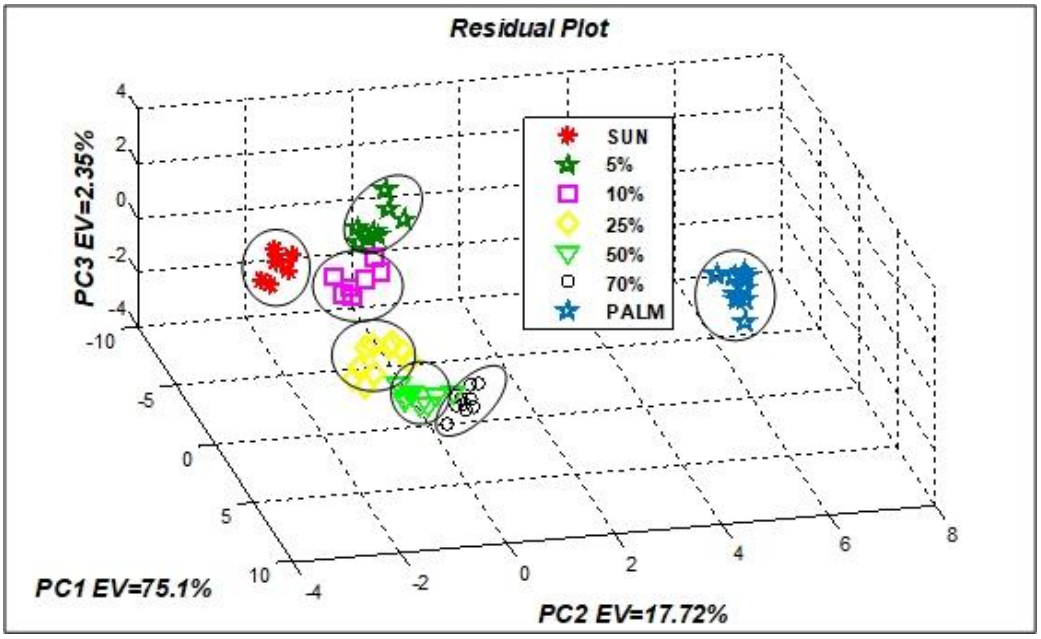


Figure 4.47 Three dimensional PCA score plot for sunflower oil adulteration with palm oil in R3

A. Results of developed models for qualitative detection of adulteration in edible oils

PCA-DA: The PCA-DA classification results of these algorithms for three spectral regions (R1, R2 and R3) data are shown in Table 4.18 as a confusion matrix. Overall PCA-DA classification accuracy is found to be 97.1% and validation accuracy is 96.3%. With the data selected for PCA loading plot (R3), 100% classification accuracy is obtained. The adulterated sunflower oil samples with palm oils are clearly classified as separate classes from pure oil samples. Among the adulterated samples, one 50% adulterated sample was listed as part of a 75% class of adulterated samples. However, a pure sample is never labeled as an adulterated sample, demonstrating the effectiveness of developed algorithms for adulteration detection. For PCA-DA and PLS-DA models, the classification performance indices namely sensitivity, specificity, accuracy have been calculated from confusion matrix and shown in Table 4.20, 4.21 and 4.22.

Table 4.18 Classification results of adulterated edible oils using PCA-DA

Wavenumber (cm^{-1})	**Total Samples**	**% Adulteration Real/Predicted**	**Number of classes (PCA-DA)**							**Accuracy (%)**
			SUN	**5%**	**10%**	**25%**	**50%**	**75%**	**PALM**	
R1 1781-1635	10	SUN	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	5%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	10%	0 0%	0 0%	10 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	25%	0 0%	0 0%	0 0%	9 12.9%	1 1.4%	0 0%	0 0%	90.0% 10.0%
	10	50%	0 0%	0 0%	0 0%	1 1.4%	8 11.4%	0 0%	0 0%	88.9% 11.1%
	10	75%	0 0%	0 0%	0 0%	0 0%	1 1.4%	10 14.3%	0 0%	90.9% 9.1%
	10	PALM	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
			100 0%	100 0%	100 0%	90% 10%	80% 20%	100% 0%	100% 0%	95.7% 4.3%
R2 1492-937	**10**	**SUN**	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**5%**	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**10%**	0 0%	0 0%	10 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**25%**	0 0%	0 0%	0 0%	10 12.9%	0 0%	0 0%	0 0%	100.0% 10.0%
	10	**50%**	0 0%	0 0%	0 0%	0 0%	9 11.4%	1 1.4%	0 0%	90% 10.0%
	10	**75%**	0 0%	0 0%	0 0%	0 0%	1 1.4%	9 14.3%	0 0%	90% 10.0%
	10	**PALM**	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
			100 0%	100 0%	100 0%	100% 10%	90% 20%	90% 0%	100% 0%	97.1% 2.9%
R3 (1717-1581, 1501-1447, 1372-1334, 1263-937)	**10**	**SUN**	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**5%**	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**10%**	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**25%**	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	100% 0.0%
	10	**50%**	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	100% 0.0%
	10	**75%**	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	100% 0.0%
	10	**PALM**	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
			100 0%	100 0%	100 0%	100 0%	100 0%	100% 0%	100% 0%	100% 0%

Table 4.19 Classification results of adulterated edible oils using PLS-DA

Wavenumber (cm^{-1})	Total Samples	% Adulteration Real/Predicted	Number of classes (PLS-DA)							Accuracy (%)
			SUN	5%	10%	25%	50%	75%	PALM	
R1 1781-1635	10	**SUN**	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**5%**	0 0%	10 14.3%	2 2.9%	0 0%	0 0%	0 0%	0 0%	83.3% 16.7%
	10	**10%**	0 0%	0 0%	8 11.4%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**25%**	0 0%	0 0%	0 0%	6 8.6%	1 1.4%	0 0%	0 0%	85.7% 14.3%
	10	**50%**	0 0%	0 0%	0 0%	4 5.7%	9 11.4%	4 5.7%	0 0%	88.9% 11.1%
	10	**75%**	0 0%	0 0%	0 0%	0 0%	0 0%	6 8.6%	0 0%	90.9% 9.1%
	10	**PALM**	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
			100 0%	100 0%	80% 20%	60% 40%	90% 10%	60% 40%	100% 0%	84.3% 15.7%
R2 1492-937	**10**	**SUN**	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**5%**	0 0%	8 11.4%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**10%**	0 0%	2 2.9%	10 0%	0 0%	0 0%	0 0%	0 0%	83.3% 16.7%
	10	**25%**	0 0%	0 0%	0 0%	8 11.4%	0 0%	0 0%	0 0%	88.9.0% 11.1%
	10	**50%**	0 0%	0 0%	0 0%	2 2.9%	7 10.0%	2 2.9%	0 0%	63.6% 36.4%
	10	**75%**	0 0%	0 0%	0 0%	0 0%	2 2.9%	8 11.4%	0 0%	80% 20.0%
	10	**PALM**	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
			100 0%	80% 20%	100 0%	80% 20%	70% 30%	80% 20%	100% 0%	87.1% 12.9%
R3 (1717-1581, 1501-1447, 1372-1334, 1263-937)	**10**	**SUN**	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**5%**	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**10%**	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
	10	**25%**	0 0%	0 0%	0 0%	7 10.0%	1 1.4%	2 2.9%	0 0%	70% 30%
	10	**50%**	0 0%	0 0%	0 0%	2 2.9%	9 12.9%	1 1.4%	0 0%	75% 25.0%
	10	**75%**	0 0%	0 0%	0 0%	1 1.4%	0 0%	7 10.0%	0 0%	87.5% 12.5%
	10	**PALM**	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
			100 0%	100 0%	100 0%	70% 30%	90% 10%	70% 30%	100% 0%	90.0% 10%

Table 4.20 PCA-DA Classification performance parameters in region R1

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
Sunflower	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	0.900	0.9000	0.9834	0.0167	0.9000	0.900
50%	0.800	0.8889	0.9834	0.0167	0.8421	0.800
75%	1	0.9091	0.9834	0.0167	0.9523	1
Palm	1	1	1	0	1	1
Overall	**0.9571**	**0.9569**	**0.9929**	**0.0071**	**0.9564**	**0.9571**

Table 4.21 PCA-DA Classification performance parameters in region R2

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
Sunflower	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	1	1	1	0	1	1
50%	0.9000	0.900	0.9834	0.0167	0.9000	0.9000
75%	0.9000	0.900	0.9834	0.0167	0.9000	0.9000
Palm	1	1	1	0	1	1
Overall	**0.9714**	**0.9714**	**0.9952**	**0.0048**	**0.9714**	**0.9714**

Table 4.22 PCA-DA Classification performance parameters in region R3

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
Sunflower	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	1	1	1	0	1	1
50%	1	1	1	0	1	1
75%	1	1	1	0	1	1
Palm	1	1	1	0	1	1
Overall	**1.0**	**1.0**	**1.0**	**0.0**	**1.0**	**1.0**

Table 4.23 PLS-DA Classification performance parameters in region R1

Samples	**Sensitivity**	**Precision**	**Specificity**	**False Positive value**	**F1-score**	**Accuracy**
Sunflower	1	1	1	0	1	1
5%	1	0.8334	0.9667	0.0333	0.9090	1
10%	0.80000	1	1	0	0.8889	0.8000
25%	0.6000	0.8571	0.9834	0.0166	0.7059	0.6000
50%	0.900000	0.5294	0.8667	0.1333	0.6667	0.9000
75%	0.600000	1	1	0	0.7500	0.6000
Palm	1	1	1	0	1	1
Overall	**0.8429**	**0.8886**	**0.9738**	**0.0262**	**0.8458**	**0.8429**

Table 4.24 PLS-DA Classification performance parameters in region R2

Samples	**Sensitivity**	**Precision**	**Specificity**	**False Positive value**	**F1-score**	**Accuracy**
Sunflower	1	1	1	0	1	1
5%	0.8000	1	1	0	0.8889	0.8000
10%	1	0.8334	0.9667	0.0334	0.9090	1
25%	0.8000	0.8889	0.9834	0.0167	0.8421	0.8000
50%	0.7000	0.6364	0.9333	0.0667	0.6667	0.7000
75%	0.8000	0.80	0.9667	0.0334	0.8000	0.8000
Palm	1	1	1	0	1	1
Overall model parameters	**0.8714**	**0.8798**	**0.9786**	**0.0214**	**0.8724**	**0.8714**

Table 4.25 PLS-DA Classification performance parameters in region R3

Samples	**Sensitivity**	**Precision**	**Specificity**	**False Positive value**	**F1-score**	**Accuracy**
Sunflower	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	0.7000	0.70	0.9500	0.05	0.7000	0.7000
50%	0.9000	0.75	0.9500	0.005	0.8182	0.9000
75%	0.7000	0.87	0.9833	0.0167	0.7778	0.7000
Palm	1	1	1	0	1	1
Overall model parameters	**0.9000**	**0.9036**	**0.9833**	**0.0167**	**0.8994**	**0.900**

PLS-DA: PLS-DA classification model for R1, R2 and R3 data is modelled with 70 training samples, 15 testing samples and 15 validation sample data. It is observed that overall performance measures for PCA-DA results are better than PLS-DA results in all selected regions. Among the selected regions, variables selected from principal components loading plots have shown good sensitivity and specificity and accuracy as shown in Table 4.23,4.24 and 4.25. Using PLS-DA adulterated samples are clearly classified as adulterated samples are never classified into a pure class. In adulterated samples, 25% adulterated sample mixed with 50% sample in R1. For wavelength region selected from PCA loading plot showed good classification accuracy (90%) compared to the three selected ranges (R1(84.3%) and R2(87%)).

SVM: For the discrimination of adulterated edible oil samples, a linear SVM model was trained using Python 3.6. The input training data was pre-processed with mean cantering before applying PCA. The first four eigenvectors corresponding to dominant eigenvalues were chosen. These eigenvectors were multiplied with the training data set to calculate principal components. The first four Principal Components are taken as input for the linear SVM model. SVM model is trained with a squared Hinge loss function and 10000 epochs generated seven hyperplane equations. Table 4.26 shows the classification accuracy and Table 4.27 shows the hyperplane equation coefficients and bias for the trained SVM model. The linear SVM model's accuracy has been determined to be 100%.

Table 4.26 Confusion matrix for classification of adulterated edible oil samples using Linear SVM model

Predicted Class	**Target Samples class**							**Accuracy (%)**
	SUN	**5%**	**10%**	**25%**	**50%**	**75%**	**PALM**	
SUN	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
5%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
10%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
25%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	100% 0.0%
50%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	100% 0.0%
75%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	100% 0.0%
PALM	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
Precision	100 0%	100 0%	100 0%	100 0%	100 0%	100% 0%	100% 0%	100% 0%

Table 4.27 Linear SVM model hyperplane coefficients and intercept

S No	Coefficient 1	Coefficient 2	Coefficient 3	Coefficient 4	Intercept
1	0.09256	-0.03833	0.43285	0.05943	-0.8626
2	0.22968	0.09251	-0.79402	-0.16649	-2.2768
3	0.02759	-0.40636	-0.37644	0.17041	-1.4674
4	-0.0332	-0.55604	-0.13142	-0.35934	-1.61062
5	-0.24371	-0,51953	-0.00474	0.10486	-2.67728
6	-0.26685	0.09502	0.63567	0.09748	-2.30176
7	0.000045	0.23088	0.01537	0.02536	-0.87057

From the above results, it is observed that classification results of adulterated edible oils are obtained accurately with variables selected from PCA loading plot. The overall classification performance of PLS-DA, PCA-DA, and Linear SVM is shown in Table 4.28.

Table 4.28 Overall classification performance of developed models for sunflower oil adulteration

Model	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
PCA-DA	**1**	**1**	**1**	**0**	**1**	**1**
PLS-DA	**0.9000**	**0.9036**	**0.9833**	**0.0167**	**0.8994**	**0.900**
SVM	**1**	**1**	**1**	**0**	**1**	**1**

Case 2: Groundnut oil adulterated with cottonseed oil

Cottonseed oil is used as adulteration in groundnut oil and adulterated samples were prepared in the lab in proportions of 5%, 10%, 25%,50%, and 75(v/v)%. Pure groundnut oil is considered as 0% adulterated sample and pure cottonseed oil is as 100% adulterated sample while training a classification model. MIR spectroscopy with ATR sampling with these edible oil samples resulted in 105x128 data with 15 samples for each class. First unsupervised clustering analysis was carried out on the acquired data to understand the similarity in the data. The results of hierarchical and subtractive clustering are shown in Figure 4.48 and Figure 4.49.

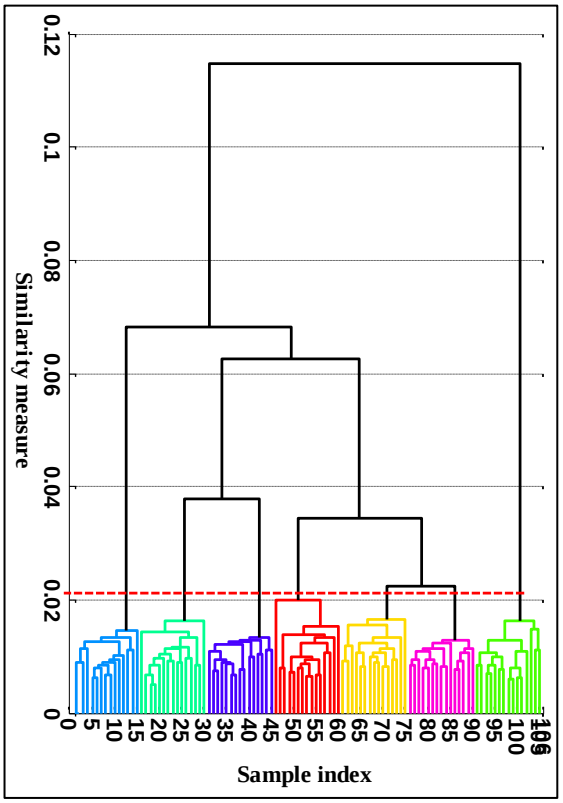


Figure 4.48 Hierarchical clustering of adulterated groundnut oil with cottonseed oil

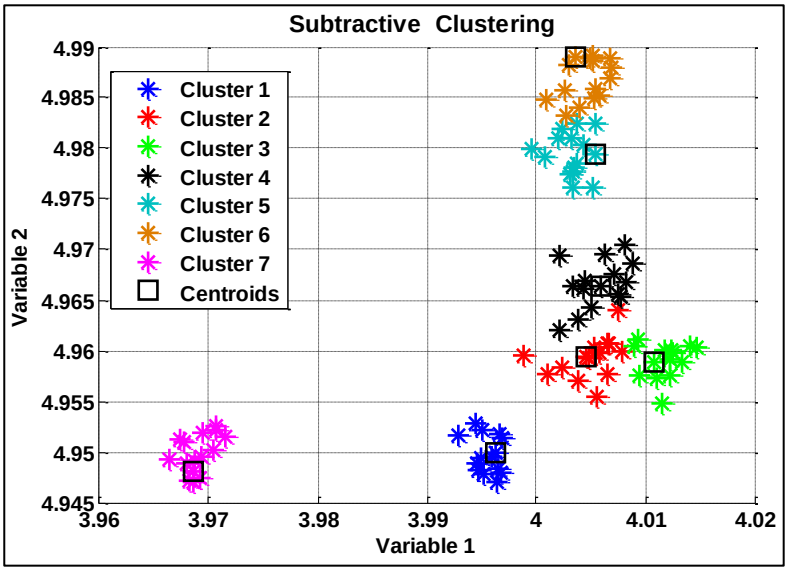


Figure 4.49 Subtractive clustering of adulterated groundnut oil with cottonseed oil

Hierarchical clustering showed a distinctive separation at a squared Euclidean distance similarity measure of 0.02. Similarly, subtractive clustering showed distinctive

clusters with squared Euclidean distance and radius of 1.5 units around the cluster centers.

Score plots of PCA on selected spectral regions are shown in Figures 4.50, 4.51 and Figure 4.52. For R1 (1492-937 cm ^{1}), the total explained variance with three principal components is above 97.1% (PC1 (66%.0%), PC2 (27%), PC3 (4.1%)). Similarly, for spectral R2 (1781 to 1635cm^{-1}) the explained variance with three principal components is 96.3% (PC1 (71.3%), PC2 (19%), PC3 (6%)). For the region selected from correlation loading plot (1717-1581,1501-1447, 1372-1334, 1263-937 cm^{-1}) the explained variance is 99.3% (PC1 (87.0%), PC2 (8.9%), PC3 (3.04%)). From the three-dimensional score plots, adulterated samples are clearly clustered.

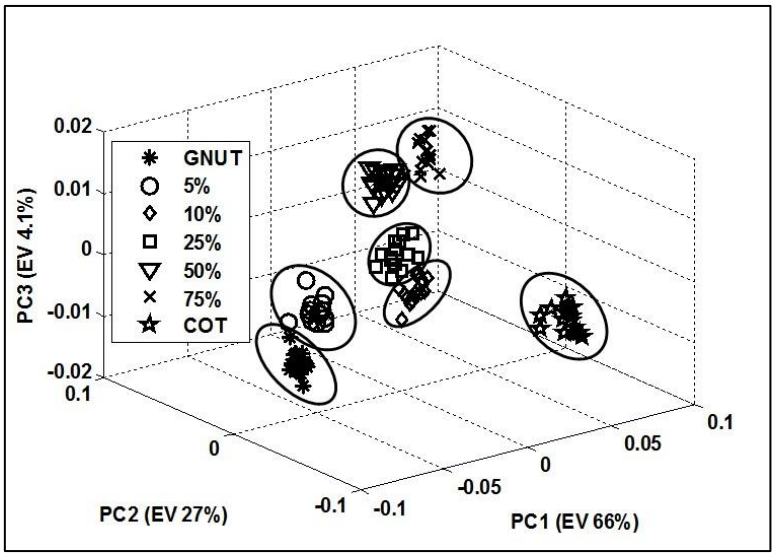


Figure 4.50 Three dimensional PCA score plot for groundnut oil adulteration with cottonseed oil in the spectral range of R1 (1492-937cm^{-1})

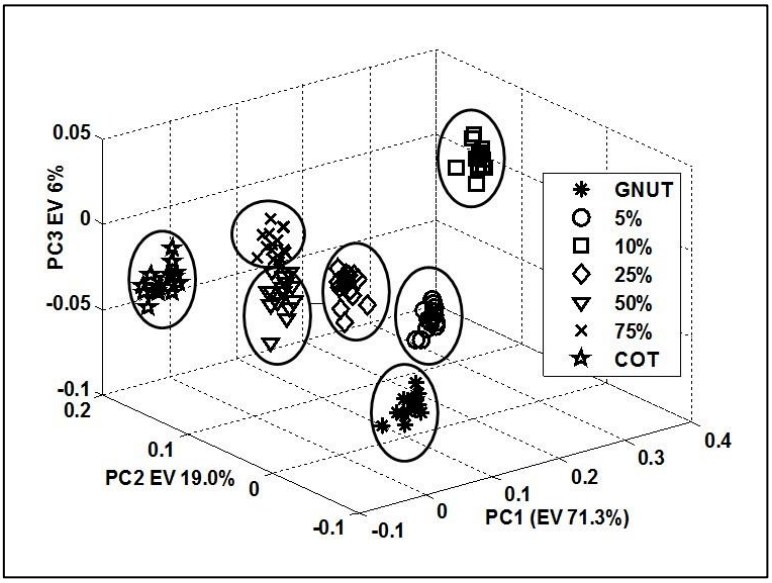


Figure 4.51 Three dimensional PCA score plot for groundnut oil adulteration with cottonseed oil in the spectral range of R2(1781 to 1635cm^{-1})

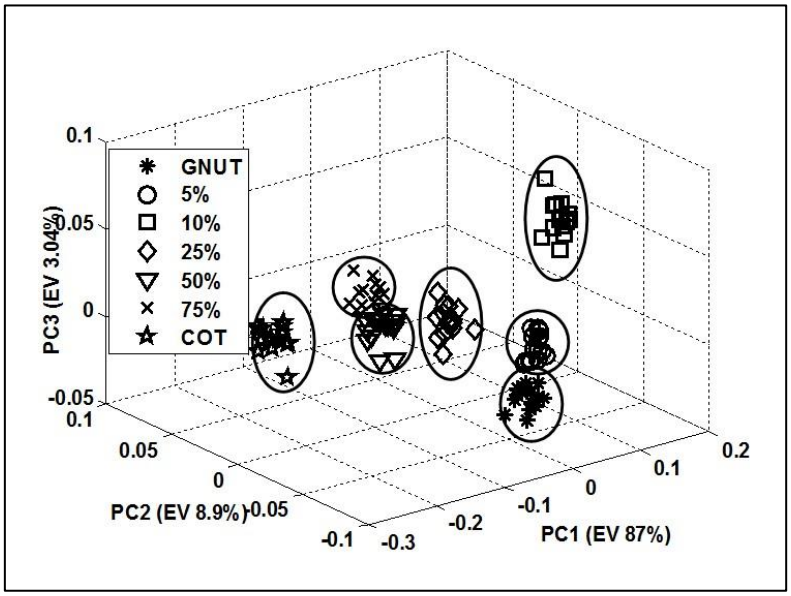


Figure 4.52 Three dimensional PCA score plot for groundnut oil adulteration with cottonseed oil in the spectral range R3

PCA-DA: PCA-DA classification results for selected spectral regions (R1 R2 and R3) data are calculated in terms of sensitivity, specificity, precision, and accuracy and presented in Tables 4.29 and 4.30 For region R1 the accuracy is observed as 99%,for R2 and R3 the PCA-DA classification accuracy is found to be 100% and validation accuracy is 100%.

Table 4.29 Classification performance of PCA-DA for R1 region

Samples	Sensitivity	Precision	Specificity	False Positive	F1-score	Accuracy
GNUT	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	1	1	1	0	1	1
50%	1	0.9375	0.9889	0.0111	0.9677	1
75%	0.9334	1	1	0	0.9655	0.9334
COT	1	1	1	0	1	1
Overall	**0.9905**	**0.9911**	**0.9984**	**0.0016**	**0.9905**	**0.9905**

Table 4.30 Classification performance of PCA-DA for R3 data

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
GNUT	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	1	1	1	0	1	1
50%	1	1	1	0	1	1
75%	1	1	1	0	1	1
COT	1	1	1	0	1	1
Overall	**1.0**	**1.0**	**1.0**	**0.0**	**1.0**	**1.0**

PLS-DA: PLS-DA classification algorithm was developed with 70 training samples, 30 testing samples data. From the PLS-DA results, the adulterated groundnut oil samples with cottonseed oils are always clearly classified as separate classes from pure oil samples. For R1, among the adulterated samples, one (50%) adulterated sample was listed as part of a 75% class of adulterated samples. However, a pure sample is never labeled as an adulterated sample. For R2 and R3 regions all adulterated samples are classified with 100% accuracy.

It is observed that PCADA results are better than PLSDA results. Using PLS-DA also adulterated samples were never classified into a pure class. But within adulterated samples, five number of 50% adulterated samples were mixed with 75% sample in R2. Overall, the PCADA results are better than PLSDA classification results (98.1%) in all selected regions. The classification performance metrics calculated for R1, R2, and R3 using PLS-DA are shown in Tables 4.31, 4.32, and 4.33 respectively.

Table 4.31 Classification performance of PLS-DA for R1 region

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
GNUT	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	1	1	1	0	1	1
50%	0.9334	0.9334	0.9889	0.0111	0.9334	0.9334
75%	0.9334	0.9334	0.9889	0.0111	0.9334	0.9334
Palm	1	1	1	0	1	1
Overall	**0.9810**	**0.9810**	**0.9968**	**0.0032**	**0.9810**	**0.9810**

Table 4.32 Classification performance of PLS-DA for R2 region

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
GNUT	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	1	1	1	0	1	1
50%	0.8666	1	1	0	0.9286	0.8666
75%	1	0.8824	0.9778	0.0223	0.9375	1
COT	1	1	1	0	1	1
Overall	**0.9810**	**0.9832**	**0.9968**	**0.0031**	**0.9905**	**0.9810**

Table 4.33 Classification performance of PLS-DA for R3 region

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
GNUT	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	0.9334	1	1	0	0.9655	0.9334
50%	1	0.9375	0.9889	0.0111	0.9677	1
75%	1	1	1	0	1	1
Palm	1	1	1	0	1	1
Overall	**0.9905**	**0.9911**	**0.9984**	**0.0016**	**0.9905**	**0.9905**

SVM: Linear SVM model was developed for discrimination of adulterated edible oil samples. Input training data is pre-processed with mean cantering and followed by PCA first four PCs are taken as input for the linear SVM model. Linear SVM model developed using squared hinge loss function with 10000 epochs resulting in seven hyperplane equations. Linear SVM model accuracy has been observed as 100% and it

is shown in a confusion matrix in Table 4.34. The hyperplane equation coefficients and bias for the developed SVM model are given in Table 4.35.

Table 4.34 Confusion matrix for classification of adulterated samples using Linear SVM model

% Adulteration Predicted Class	**Target Samples class**							**Accuracy (%)**
	GNUT	**5%**	**10%**	**25%**	**50%**	**75%**	COT	
GNUT	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
5%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
10%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
25%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	100% 0.0%
50%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	100% 0.0%
75%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	100% 0.0%
COT	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
Precision	100 0%	100 0%	100 0%	100 0%	100 0%	100% 0%	100% 0%	100% 0%

Table 4.35 Linear SVM model hyperplane coefficients and bias

S. No	**Coefficient 1**	**Coefficient 2**	**Coefficient 3**	**Coefficient 4**	**Intercept**
1	-0.01675216	1.11668004,	0.0534307	2.46962496	0.13060277,
2	-0.01517757	-1.76253115	-0.55015032	1.36581228	0.03127781,
3	-0.01241064	-1.88930905	-3.08122723	-2.14686272	-0.09447107,
4	-0.01365588	-0.29275527	0.4825773	-0.2460055	-0.02484984,
5	-0.01351692	0.27539625	2.60241024	-0.52866573	-0.03184446,
6	-0.01238014	0.35926036	2.70387398	-1.34882483	- 0.08891927
7	-0.01593669	4.13340811	-1.91764149	0.34766949	0.06743734

CNN: Deep neural networks based on 1D CNN and 2D CNN models were developed for the detection of adulterated samples. CNN model derives features from input data at different levels of abstraction. CNN are invariant to spatial transformations whilst lowering the overall computational cost. The implementation of inference algorithms in general C language and on embedded systems is also possible. CNN architecture which is explained in the classification of edible oils section, spectroscopy is also used for detection of adulterations in groundnut oil using ATR sampling data.

1D CNN model with two convolution layers and two dense layers was developed in Python 3.6(Keras library with TensorFlow backend). The model was trained on train data of 70 samples, validation data is 16 samples. Relu activation function is used in convolution layer followed by a max pool layer. The final dense layer activation function was softmax. Model hyperparameters include batch size as 1, stride 1, learning rate as 0.001. 1D CNN model was trained with Different optimization techniques. The Flatten layer was replaced with global averaging and global max pooling layer to eliminate overfitting and to reduce the trainable parameters in 1D CNN. The maximum training epochs is 1000. Trained model architecture is shown in Figure 4.53.

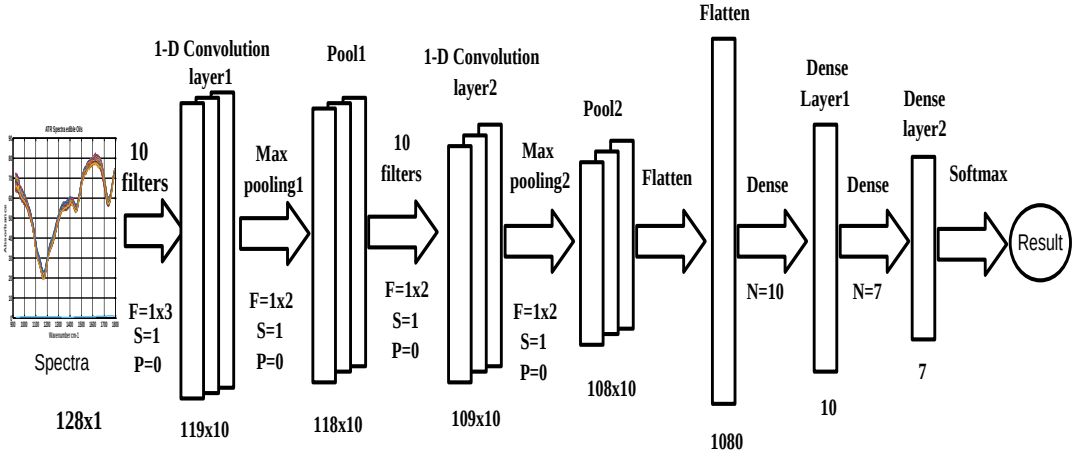


Figure 4.53 Proposed 1D CNN sequential model architecture summary for detecting adulteration of edible oils

Table 4.36 Confusion matrix for classification of adulterated samples using 1D CNN

% Adulteration Predicted Class	Target Samples class							Accuracy (%)
	GNUT	5%	10%	25%	50%	75%	COT	
GNUT	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
5%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
10%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
25%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	100% 0.0%
50%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	100% 0.0%
75%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	100% 0.0%
COT	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
	100 0%	100 0%	100 0%	100 0%	100 0%	100% 0%	100% 0%	100% 0%

Table 4.37 CNN results comparison with different optimization methods

Layer structure at Output	Optimization Method	Epochs to converge	Accuracy (%)	Trainable Parameters
Flatten Layer	Adam	75	100	12637
	SGD	415	100	
	Adagrad	330	100	
	RMSprop	325	100	
Global max pooling	Adam	400	94	427
	SGD	1000	91	
	Adagrad	1000	88	
	RMSprop	400	94	
Global average pooling	Adam	480	94.0	427
	SGD	1000	89	
	Adagrad	600	85	
	RMSprop	450	91	

The results of CNN model with global pooling layers and different optimization methods have been presented in Table 4.37. The trained model showed 100% classification accuracy in case of Flatten layer with all optimization methods , all adulterated edible oils samples were classified into adulterated class and pure sample into pure. The classification results are shown in a confusion matrix in Table 4.36. These trained model parameters like kernel filters, weights and bias have been saved to a CSV file, the 1-D CNN inference algorithm implementation on an embedded platform is explained in chapter 6.

2D CNN was applied to the correlation matrix of an ATR spectrum to study the correlation of correlation data of an ATR spectrum at different wavenumbers. 2D CNN was trained on 70 correlation spectra data with a learning rate of 0.001, batch size of 1. For training the CNN model, various optimization methods such as stochastic gradient descent (SGD), RMSprop, Adaptive gradient(adagrad), and Adam optimization algorithm have been utilised. The Flatten layer was replaced by global max pooling and global average pooling layer in the other investigation using these optimization approaches. These global max poling and global average pooling methods not only significantly reduces the computation parameters but also very helpful in the embedded implementation of these algorithms. The architecture of 2D CNN with Flatten layer and dense layer at the output stage is shown in Figure 4.54. The classification accuracy was 100%. The results of CNN model with global pooling layers and different optimization methods have been presented in Table 4.38.

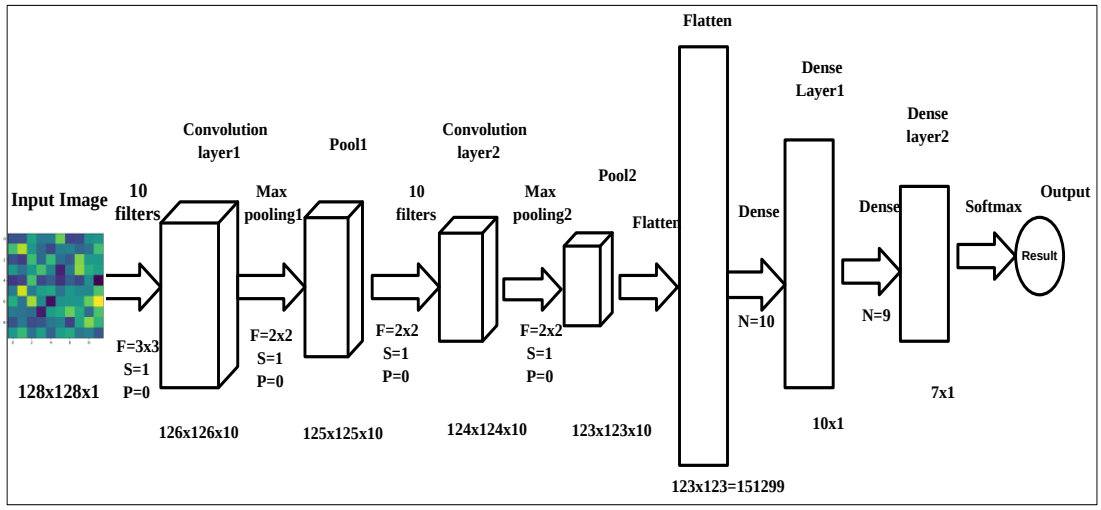


Figure 4.54 Proposed 2D CNN sequential model architecture for detecting adulteration of edible oils

. It is observed from the obtained results with generic CNN model with flatten and dense layer at the output stage, the total number of trainable parameters are 1489197. Adagrad optimization converged with a smaller number of epochs than other optimizations. The classification accuracy with all optimization methods is 100%. In case of global average pooling, the total trainable parameters are 1087. All four optimization methods have shown 99% of classification accuracy, although they converged at different training epochs. In case of global max pooling only Adam and RMSprop optimization methods have shown 99% of classification accuracy. From these results it is observed that with replacing the flatten layer with global averaging layer the results are on par with the regular CNN with flatten layer results. Hence for less computation, these models can be used.

Table 4.38 CNN results comparison with different optimization methods

Layer structure	**Optimization Method**	**Epochs to converge**	**Accuracy (%)**	**Trainable Parameters**
Flatten Layer	Adam	665	99.05	1489197
	SGD	100	99.05	
	Adagrad	250	99.05	
	RMSprop	1000	99.05	
Global max pooling	Adam	400	99.05	1087
	SGD	1000	85%	
	Adagrad	1000	90.3	
	RMSprop	400	98.05	
Global average pooling	Adam	480	99.0%	1087
	SGD	1000	95.2%	
	Adagrad	600	98.6%	
	RMSprop	450	97.2%	

Case 3: Sesame oil adulterated with cottonseed oil

Cottonseed oil is used as adulteration in sesame oil. The adulterated samples were prepared in the lab in proportions of 5%, 10%, 25%,50%, and 70%. Pure sesame oil is considered as 0% adulterated sample and pure cottonseed oil is a 100% adulterated sample while developing a calibration model. MIR spectroscopy with ATR sampling with these edible oil samples resulted in 105x128 data with 15 samples for each class. Unsupervised clustering analysis was carried out on the acquired data to understand the similarity in the data. The results of hierarchical and subtractive clustering are shown in Figures 4.55 and 4.56, respectively.

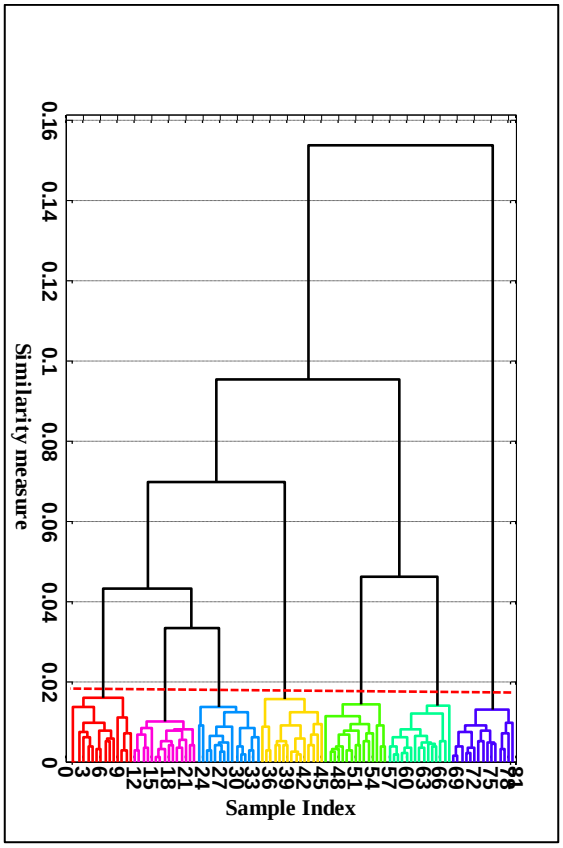


Figure 4.55 Hierarchical clustering of adulterated Sesame oil with cottonseed oil

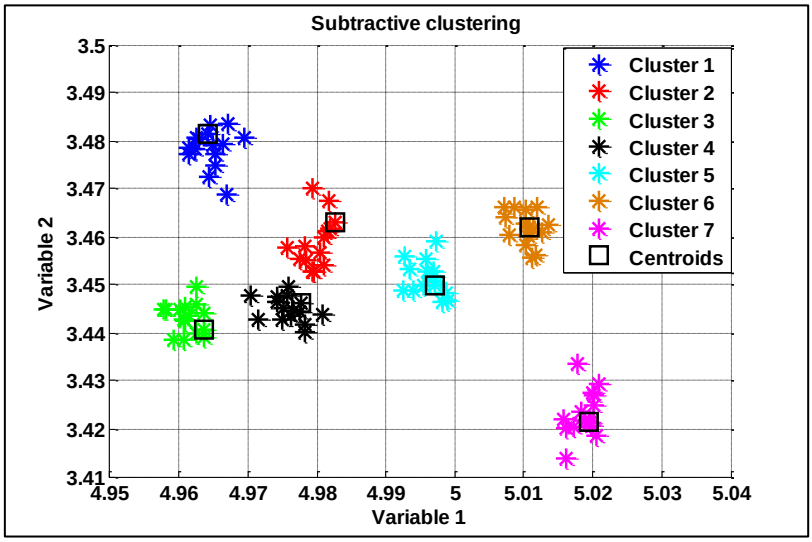


Figure 4.56 Subtractive clustering of adulterated sesame oil with cottonseed oil

Score plots of PCA on selected spectral regions are shown in Figures 4.57, 4.58, and Figure 4.59. For region 1 (1781 to 1635cm^{-1}), the total explained variance with three principal components is above 97.1% (PC1 (66%.0%), PC2 (27%), PC3(4.1%)). Similarly, for spectral region 2 (1492-937 cm^{-1}), the explained variance with three principal components is 96.3% (PC1(71.3%), PC2 (19%), PC3(6%)). For the region selected from correlation loading plot the explained variance is 99.3% (PC1 (87.0%), PC2 (8.9%), PC3 (3.04%)). From the three-dimensional score plots, adulterated samples are seemed to be clearly clustered.

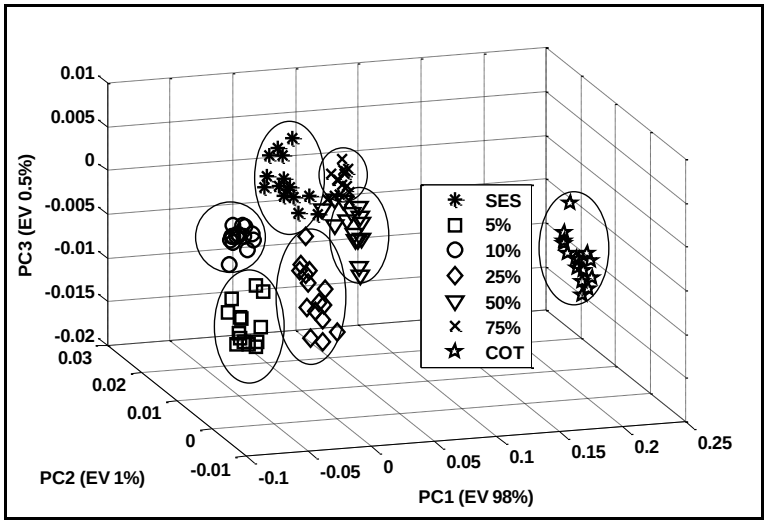


Figure 4.57 Three dimensional PCA score plot for sesame oil adulteration with cottonseed oil in R1 (1781 to 1635cm^{-1})

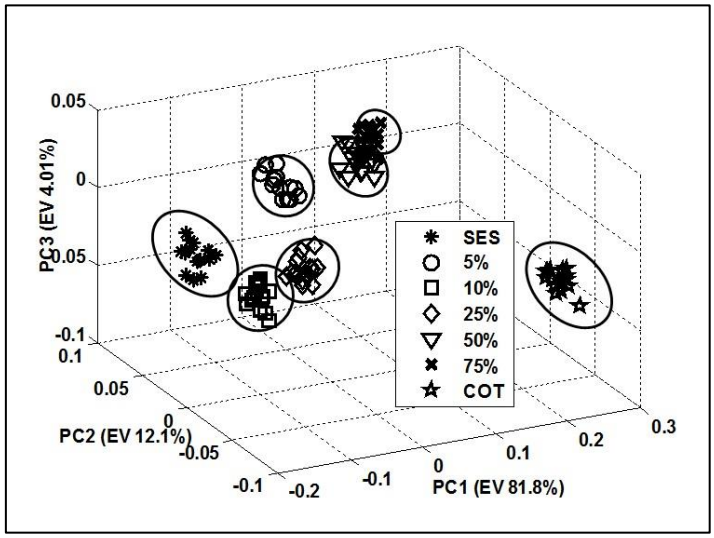


Figure 4.58 Three dimensional PCA score plot for sesame oil adulteration with cottonseed oil in the spectral range R2 (1492-937 cm^{-1})

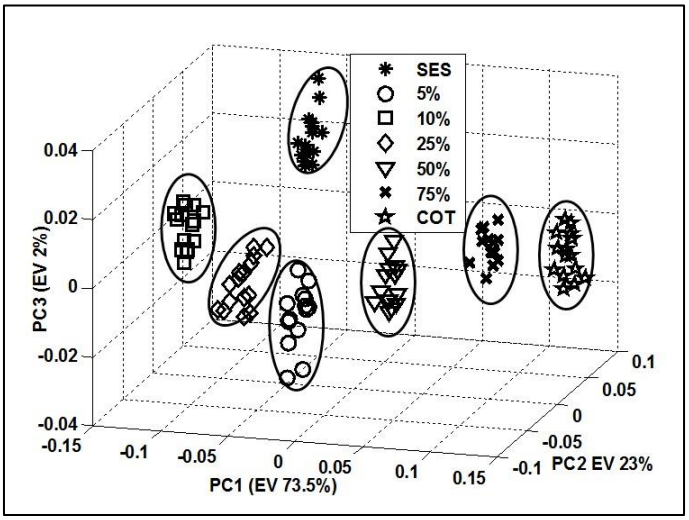


Figure 4.59 Three-dimensional PCA score plot for sesame oil adulteration with cottonseed oil in the spectral range selected from correlation loading plot

PCA-DA model was trained in Matlab 2014 using classification toolbox 3.1. Data is mean centered, and discriminant algorithm was implemented with first four Principal Components. Venetian blind cross validation used with four validation groups from the data. The PCA-DA classification results for three selected spectral regions (R1, R2 and R3) in terms of classification performance metrics sensitivity, specificity and precision calculated using a confusion matrix are shown in Table 4.39. Overall PCA-DA the classification accuracy is found to be 100%. The adulterated sesame oil samples with cottonseed oils are clearly classified as separate classes from pure oil samples. No adulterated sample is classified into a pure sample. The accuracy of the model is

estimated using the classification performance measures which are indicated in Table 4.39 below.

Table 4.39 Classification performance of PCA-DA for R1, R2 and R3 region

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
SES	1	1	1	0	1	1
5%	1	1	1	0	1	1
10%	1	1	1	0	1	1
25%	1	1	1	0	1	1
50%	1	1	1	0	1	1
75%	1	1	1	0	1	1
COT	1	1	1	0	1	1
Overall	**1.0**	**1.0**	**1.0**	**0.0**	**1.0**	**1.0**

PLS-DA classification algorithm was also trained using classification toolbox 3.1, with 70 training samples, 30 testing samples data. Data mean centering and Venetian blind cross validation with four validations sets from the data was used. Using PLS-DA adulterated samples are clearly classified as adulterated never classified into a pure class. within adulterated samples, three samples of 25% adulteration sample were classified as 10% sample in R1. Overall, the PCADA results are better than PLSDA classification results (96.1%) in all selected regions. Tables 4.40, 4.41 and 4.42 show the performance of PLS-DA model using sensitivity, specificity, and accuracy values for the detection of sesame oil adulteration with cottonseed oil in various proportions.

Table 4.40 Classification performance of PLS-DA for R1

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
SES	1	1	1	0	1	1
5%	1	0.9375	0.9889	0.0111	0.9677	1
10%	0.934	0.8235	0.9667	0.0333	0.8750	0.934
25%	0.800	1	1	0	0.8889	0.800
50%	1	1	1	0	1	1
75%	1	1	1	0	1	1
COT	1	1	1	0	1	1
Overall	**0.9619**	**0.9659**	**0.9937**	**0.0063**	**0.9617**	**0.9619**

Table 4.41 Classification performance of PLS-DA for R2

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
SES	1	1	1	0	1	1
5%	1	0.8824	0.9778	0.0223	0.9375	1
10%	0.8667	0.8125	0.9667	0.0334	0.8387	0.8667
25%	0.8000	1	1	0	0.8889	0.8000
50%	1	1	1	0	1	1
75%	1	1	1	0	1	1
COT	1	1	1	0	1	1
Overall model parameters	**0.9524**	**0.9564**	**0.9921**	**0.0079**	**0.9522**	**0.9524**

Table 4.42 Classification performance of PLS-DA for R3

Samples	Sensitivity	Precision	Specificity	False Positive value	F1-score	Accuracy
SES	1	1	1	0	1	1
5%	0.9375	1	1	0	0.9677	0.9375
10%	0.9375	1	1	0	0.9677	0.9375
25%	1	0.9334	0.9891	0.0108	0.9655	1
50%	0.93333	0.9334	0.9890	0.0109	0.9333	0.93333
75%	1	0.9375	0.9890	0.0109	0.9677	1
COT	1	1	1	0	1	1
Overall	**0.9333**	**0.9385**	**0.9889**	**0.0111**	**0.9312**	**0.9333**

SVM model with linear kernel was trained using Scikit-learn 0.24.1 in Python 3.6 for discrimination of adulterated edible oil samples. Input training data is pre-processed with mean centering and followed by PCA. First four PCs are taken as input for the linear SVM model. Linear SVM model developed using squared hinge loss function with 10000 epochs resulting in seven hyperplane equations. The hyperplane equation coefficients and bias for the developed SVM model are given in Table 4.44. Linear SVM model accuracy has been observed as 100% and it is shown in a confusion matrix in Table 4.43.

Table 4.43 Confusion matrix for classification of adulterated samples using Linear SVM model

%Adulteration Predicted	Target Samples class							Accuracy (%)
	SES	**5%**	**10%**	**25%**	**50%**	**75%**	COT	
SES	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
5%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
10%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
25%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	100% 0.0%
50%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	100% 0.0%
75%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	100% 0.0%
COT	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
	100 0%	100 0%	100 0%	100 0%	100 0%	100% 0%	100% 0%	100% 0%

Table 4.44 Linear SVM model hyperplane coefficients and bias

S. No	Coefficient 1	Coefficient 2	Coefficient 3	Coefficient 4	Intercept
1	-0.0147014	-3.0893858	2.5167813	1.98972347	0.02148934
2	-0.01499028	-1.19850194	-1.92774751	-1.38903158	0.03265076
3	-0.01734954	-1.0306039	0.88219812	-1.65226855	0.15872041
4	-0.01628359	-0.24101061	-0.6853277	1.85786236	0.10543043
5	-0.01387322	0.31686112	-2.14755502	1.2536795	-0.01512479
6	-0.01128729	0.19432243,	-1.88509151	1.56001177	0.14479947
7	-0.01297922,	4.59535668,	1.31664822,	-0.06431792	-0.09886905

Deep neural networks based on 1D CNN and 2D CNN models were trained for the detection of adulterated samples. CNN architecture which is shown in Figure 4.53 and 4.54 are used for the detection of adulterations in sesame oil. 1D CNN and 2D CNN algorithms were successful in the discrimination of adulterated sesame oil with cottonseed oil.

Different optimization methods like stochastic gradient descent (SGD), RMSprop, Adaptive gradient(adagrad) and Adam optimization algorithm have been used for optimization of parameters. The other analysis with these optimization methods have been carried out with replacing the Flatten layer with global max pooling and global average pooling layer. These global max poling and global average pooling methods not only reduces the computation parameters but also very helpful in the embedded implementation of these algorithms.

Table 4.45 1D-CNN results comparison with different optimization methods

Layer structure at Output	**Optimization Method**	**Epochs to converge**	**Accuracy (%)**	**Trainable Parameters**
Flatten Layer	Adam	70	100,0	12637
	SGD	340	100.0	
	Adagrad	700	100.0	
	RMSprop	480	100.0	
Global max pooling	Adam	950	97.0	427
	SGD	1400	97.0	
	Adagrad	1500	97.0	
	RMSprop	963	94.2	
Global average pooling	Adam	980	97.0	427
	SGD	1500	98.0	
	Adagrad	1290	98,0	
	RMSprop	1300	98%	

Table 4.46 2D-CNN results comparison with different optimization methods

Layer structure at Output	**Optimization Method**	**Epochs to converge**	**Accuracy (%)**	**Trainable Parameters**
Flatten Layer	Adam	200	100	1489197
	SGD	270	100	
	Adagrad	120	100	
	RMSprop	175	100	
Global max pooling	Adam	500	99.05	1087
	SGD	1200	96.04	
	Adagrad	1200	90	
	RMSprop	750	99.05	
Global average pooling	Adam	480	99.05%	1087
	SGD	100	98%	
	Adagrad	500	99.05	
	RMSprop	490	98.5%	

The classification accuracy was 100%. The results of CNN model with global pooling layers and different optimization methods have been presented in Table 4.45 and Table 4.46.

. It is observed from the obtained results with generic CNN model with flatten and dense layer at the output stage, the total number of trainable parameters are 1489197. Adagrad optimization converged with a smaller number of epochs than other

optimizations. The classification accuracy with all optimization methods is 100%. In case of global average pooling, the total trainable parameters are 1087. All four optimization methods have shown 99% of classification accuracy, although they converged at different training epochs. In case of global max pooling only Adam and RMSprop optimization methods have shown 99% of classification accuracy. From these results it is observed that with replacing the flatten layer with global averaging layer the results are on par with the regular CNN with flatten layer results. Hence for less computation, these models can be used.

4.10. Summary

The fundamentals of statistics, as well as an overview of machine learning and artificial intelligence was covered at the outset of this chapter. The methodology of data analysis algorithms for edible oil classification and adulteration detection was presented. The principal component analysis was used as a data reduction and feature selection method. PLSDA, PCADA, SIMCA, SVM, and CNN supervised classification algorithms were developed for edible oil classification employing electronic tongue data, NIR spectroscopy data, and ATR spectroscopy data. In the case of voltammetry experiment data, the developed classification algorithms PLS-DA and Linear SVM exhibited 100% classification accuracy for distinguishing eight types of edible oils. Similarly, with EIS data, PLS-DA demonstrated 100% classification accuracy. Experiments with NIR spectroscopy on five different types of edible oils revealed that PLSDA had a 100% classification accuracy. PLS-DA, SVM models on ATR spectroscopy data for edible oil classification, on the other hand, resulted in accurate classification of nine types of edible oils. A novel approach for developing deep neural network algorithms for ATR spectroscopy data for edible oil classification was also presented. The 1D CNN algorithm was developed with 70% of the training data set and tested with 30% of the test data, resulting in 100% classification accuracy. Another novel method of developing 2D CNN algorithms based on the correlation spectrum of each sample also produced accurate results.

Data from ATR spectroscopy with adulterated samples was used to develop detection and quantification algorithms. Data analysis was performed at three different wavenumber regions. These regions were chosen from the literature to cover the dominant functional groups absorbance wavelengths in edible oils. PCA-DA performed better in terms of classification accuracy. The classification accuracy of the linear SVM, 1D CNN, and 2D CNN algorithms was 100 percent. In all cases, the accuracy of Linear SVM and CNN models outperformed all other developed algorithms. These developed

models, as well as the model parameters, were saved in readable Comma Separated Value (CSV) file format. These files were used for the embedded implementation of inference algorithms for edible oil classification that is to be covered in chapter 6. This implementation will assist in the development of intelligent portable instrumentation for the analysis of edible oils. MIR spectroscopy with ATR sampling has been identified as the simplest and perhaps most accurate method for edible oil analysis in this research. The developed algorithm results show the algorithm's capability in classification and qualitative adulteration detection in edible oils.

Bibliography

[1] P. Singham, P. Birwal, and B. K. Yadav, "Importance of Objective and Subjective Measurement of Food Quality and their Inter-relationship," J Food Process Technol, vol. 6, p. 488, 2015, doi: 10.4172/2157-7110.1000488.

[2] T. Zhang, T. Wang, R. Liu, M. Chang, Q. Jin, and X. Wang, "Chemical characterization of fourteen kinds of novel edible oils: A comparative study using chemometrics," LWT, vol. 118, p. 108725, Jan. 2020, doi: 10.1016/j.lwt.2019.108725.

[3] H. Karami, M. Rasekh, and E. Mirzaee – Ghaleh, "Comparison of chemometrics and AOCS official methods for predicting the shelf life of edible oil," Chemom. Intell. Lab. Syst., vol. 206, p. 104165, Nov. 2020, doi: 10.1016/j.chemolab.2020.104165.

[4] W. G. Hunter, "Statistics and Chemistry, and the Linear Calibration Problem," in Chemometrics, Springer Netherlands, 1984, pp. 97–114.

[5] S. Wold et al., "Multivariate Data Analysis in Chemistry," in Chemometrics, Springer Netherlands, 1984, pp. 17–95.

[6] X. J. R. Avula, "Mathematical Modeling," in Encyclopedia of Physical Science and Technology, Elsevier, 2003, pp. 219–230.

[7] J. P. Haton, "A brief introduction to artificial intelligence," in IFAC Proceedings Volumes (IFAC-PapersOnline), Jan. 2006, vol. 9, no. PART 1, pp. 8–16, doi: 10.3182/20060522-3-fr-2904.00003.

[8] T. C. Guetterman, "Basics of statistics for primary care research," Fam. Med. Community Heal., vol. 7, no. 2, p. 67, Mar. 2019, doi: 10.1136/fmch-2018-000067.

[9] B. G. Tabachnick and L. S. Fidell, Using Multivariate Statistics Title: Using multivariate statistics. 2019.

[10] R. Shi and J. W. McLarty, "Descriptive statistics," Annals of Allergy, Asthma and Immunology, vol. 103, no. 4 SUPPL. Apress, Berkeley, CA, pp. 125–174, 2009, doi: 10.1016/s1081-1206(10)60815-0.

[11] M. Titus, "Using Descriptive Statistics and Graphs," Springer, Cham, 2021, pp. 79–102.

[12] K. Potter, J. Kniss, R. Riesenfeld, and C. R. Johnson, "Visualizing summary statistics and uncertainty," Comput. Graph. Forum, vol. 29, no. 3, pp. 823–832, Jun. 2010, doi: 10.1111/j.1467-8659.2009.01677.x.

[13] T. Cleff, "Univariate Data Analysis," in Applied Statistics and Multivariate Data Analysis for Business and Economics, Springer International Publishing, 2019, pp. 27–70.

[14] A. Bertani, G. Di Paola, E. Russo, and F. Tuzzolino, "How to describe bivariate data," J. Thorac. Dis., vol. 10, no. 2, pp. 1133–1137, Feb. 2018, doi: 10.21037/jtd.2018.01.134.

[15] L. Wang and B. Mizaikoff, "Application of multivariate data-analysis techniques to biomedical diagnostics based on mid-infrared spectroscopy," Analytical and Bioanalytical Chemistry, vol. 391, no. 5. Springer, pp. 1641–1654, Jul. 01, 2008, doi: 10.1007/s00216-008-1989-9.

[16] O. F.Y, A. J.E.T, A. O, H. J. O, O. O, and A. J, "Supervised Machine Learning Algorithms: Classification and Comparison," Int. J. Comput. Trends Technol., vol. 48, no. 3, pp. 128–138, 2017, doi: 10.14445/22312803/ijctt-v48p126.

[17] N. Burkart and M. F. Huber, "A survey on the explainability of supervised machine learning," Journal of Artificial Intelligence Research, vol. 70. AI Access Foundation, pp. 245–317, Jan. 19, 2021, doi: 10.1613/JAIR.1.12228.

[18] R. Gentleman and V. J. Carey, "Unsupervised Machine Learning," in Bioconductor Case Studies, Springer New York, 2008, pp. 137–157.

[19] J. E. van Engelen and H. H. Hoos, "A survey on semi-supervised learning," Mach. Learn., vol. 109, no. 2, pp. 373–440, Feb. 2020, doi: 10.1007/s10994-019-05855-6.

[20] V. Francois-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An Introduction to Deep Reinforcement Learning," Found. Trends Mach. Learn., vol. 11, no. 3–4, pp. 219–354, Nov. 2018, doi: 10.1561/2200000071.

[21] S. A. Alasadi and W. S. Bhaya, "Review of data preprocessing techniques in data mining," J. Eng. Appl. Sci., vol. 12, no. 16, pp. 4102–4107, 2017, doi: 10.3923/jeasci.2017.4102.4107.

[22] B. Tudu, B. Kow, N. Bhattacharyya, and R. Bandyopadhyay, "Normalization techniques for gas sensor array as applied to classification for black tea," Int. J. Smart Sens. Intell. Syst., vol. 2, no. 1, pp. 176–189, Nov. 2009, doi: 10.21307/ijssis-2017-344.

[23] S. G. K. Patro and K. K. Sahu, "Normalization: A Preprocessing Stage," IARJSET, pp. 20–22, Mar. 2015, doi: 10.17148/iarjset.2015.2305.

[24] W. Windig, J. Shaver, and R. Bro, "Loopy MSC: A simple way to improve multiplicative scatter correction," Appl. Spectrosc., vol. 62, no. 10, pp. 1153–1159, Oct. 2008, doi: 10.1366/000370208786049097.

[25] A. Savitzky and M. J. E, "Smoothing and Differentiation of Data by Simplified Least Squares Procedures," 1951. Accessed: Apr. 03, 2021. [Online]. Available: https://pubs.acs.org/sharingguidelines.

[26] P. A. Gorry, "General Least-Squares Smoothing and Differentiation by the Convolution (Savitzky-Golay) Method," Anal. Chem., vol. 62, no. 6, pp. 570–573, Mar. 1990, doi: 10.1021/ac00205a007.

[27] G. T. Reddy et al., "Analysis of Dimensionality Reduction Techniques on Big Data," IEEE Access, vol. 8, pp. 54776–54788, 2020, doi: 10.1109/ACCESS.2020.2980942.

[28] X. Huang, L. Wu, and Y. Ye, "A Review on Dimensionality Reduction Techniques," Int. J. Pattern Recognit. Artif. Intell., vol. 33, no. 10, Sep. 2019, doi: 10.1142/S0218001419500174.

[29] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," Chemom. Intell. Lab. Syst., vol. 2, no. 1–3, pp. 37–52, Aug. 1987, doi: 10.1016/0169-7439(87)80084-9.

[30] S. Mannor et al., "K-Means Clustering," in Encyclopedia of Machine Learning, Springer US, 2011, pp. 563–564.

[31] I. Sota-Uba, M. Bamidele, J. Moulton, K. Booksh, and B. K. Lavine, "Authentication of Edible Oils Using Fourier Transform Infrared Spectroscopy and Pattern Recognition Methods," Chemom. Intell. Lab. Syst., vol. 210, p. 104251, Jan. 2021, doi: 10.1016/j.chemolab.2021.104251.

[32] C. Fredes et al., "A model based on clusters of similar color and NIR to estimate oil content of single olives," Dec. 2020, doi: 10.20944/preprints202012.0405.v1.

[33] F. Murtagh and P. Contreras, "Algorithms for hierarchical clustering: an overview," WIREs Data Min. Knowl. Discov., vol. 2, no. 1, pp. 86–97, Jan. 2012, doi: 10.1002/widm.53.

[34] K. Sasirekha and P. Baby, "Agglomerative Hierarchical Clustering Algorithm-A Review," Int. J. Sci. Res. Publ., vol. 3, no. 3, 2013, Accessed: Apr. 12, 2021. [Online]. Available: www.ijsrp.org.

[35] S. L. Chiu, "Fuzzy model identification based on cluster estimation," J. Intell. Fuzzy Syst., vol. 2, no. 3, pp. 267–278, Jan. 1994, doi: 10.3233/IFS-1994-2306.

[36] N. R. Pal and D. Chakraborty, "Mountain and subtractive clustering method: Improvements and generalizations," Int. J. Intell. Syst., vol. 15, no. 4, pp. 329–341, Apr. 2000, doi: 10.1002/(SICI)1098-111X(200004)15:4<329::AID-INT5>3.0.CO;2-9.

[37] K. McGarigal, S. Stafford, and S. Cushman, "Discriminant Analysis," in Multivariate Statistics for Wildlife and Ecology Research, New York, NY: Springer New York, 2000, pp. 129–187.

[38] A. Tharwat, T. Gaber, A. Ibrahim, and A. E. Hassanien, "Linear discriminant analysis: A detailed tutorial," AI Commun., vol. 30, no. 2, pp. 169–190, Jan. 2017, doi: 10.3233/AIC-170729.

[39] A. J. Izenman, "Linear Discriminant Analysis," Springer, New York, NY, 2013, pp. 237–280.

[40] S. Balakrishnama and A. Ganapathiraju, "INSTITUTE FOR SIGNAL AND INFORMATION PROCESSING LINEAR DISCRIMINANT ANALYSIS-A BRIEF TUTORIAL."

[41] M. Bassbasi, M. De Luca, G. Ioele, A. Oussama, and G. Ragno, "Prediction of the geographical origin of butters by partial least square discriminant analysis (PLS-DA) applied to infrared spectroscopy (FTIR) data," J. Food Compos. Anal., vol. 33, no. 2, pp. 210–215, Mar. 2014, doi: 10.1016/j.jfca.2013.11.010.

[42] M. Barker and W. Rayens, "Partial least squares for discrimination," J. Chemom., vol. 17, no. 3, pp. 166–173, Mar. 2003, doi: 10.1002/cem.785.

[43] M. Haenlein and A. M. Kaplan, "A Beginner's Guide to Partial Least Squares Analysis," Underst. Stat., vol. 3, no. 4, pp. 283–297, Nov. 2004, doi: 10.1207/s15328031us0304_4.

[44] A. A. Saleh, "A NEW VARIABLES SELECTION AND DIMENSIONALITY REDUCTION TECHNIQUE COUPLED WITH SIMCA METHOD FOR THE CLASSIFICATION OF TEXT DOCUMENTS," pp. 583–591.

[45] S. Wold, "Pattern recognition by means of disjoint principal components models," Pattern Recognit., vol. 8, no. 3, pp. 127–139, Jul. 1976, doi: 10.1016/0031-3203(76)90014-5.

[46] I. C. Yang et al., "Integration of SIMCA and near-infrared spectroscopy for rapid and precise identification of herbal medicines," J. Food Drug Anal., vol. 21, no. 3, pp. 268–278, Sep. 2013, doi: 10.1016/j.jfda.2013.07.008.

[47] K. Vanden Branden and M. Hubert, "Robust Classification in High Dimensions based on the SIMCA Method," 2005.

[48] D. A. Pisner and D. M. Schnyer, "Support vector machine," in Machine Learning: Methods and Applications to Brain Disorders, Elsevier, 2019, pp. 101–121.

[49] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proc. IEEE, vol. 86, no. 11, pp. 2278–2323, 1998, doi: 10.1109/5.726791.

[50] K. Jarrett, K. Kavukcuoglu, M. Ranzato, and Y. LeCun, "What is the best multi-

stage architecture for object recognition?," in Proceedings of the IEEE International Conference on Computer Vision, 2009, pp. 2146–2153, doi: 10.1109/ICCV.2009.5459469.

[51] D. C. Cires¸ancires¸an, A. Giusti, L. M. Gambardella, and J. ¨ Urgen Schmidhuber, "Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images," 2012. Accessed: Mar. 30, 2021. [Online]. Available: http://www.idsia.ch/.

[52] X. Liu, Z. Deng, and Y. Yang, "Recent progress in semantic image segmentation," Artif. Intell. Rev., vol. 52, no. 2, pp. 1089–1106, Aug. 2019, doi: 10.1007/s10462-018-9641-3.

[53] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for Activation Functions," arXiv, Oct. 2017, Accessed: Mar. 30, 2021. [Online]. Available: http://arxiv.org/abs/1710.05941.

[54] Y. LeCun, K. Kavukcuoglu, and C. Farabet, "Convolutional networks and applications in vision," in ISCAS 2010 - 2010 IEEE International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems, 2010, pp. 253–256, doi: 10.1109/ISCAS.2010.5537907.

[55] D. Scherer, A. Müller, and S. Behnke, "Evaluation of pooling operations in convolutional architectures for object recognition," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2010, vol. 6354 LNCS, no. PART 3, pp. 92–101, doi: 10.1007/978-3-642-15825-4_10.

[56] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 8691 LNCS, no. PART 3, pp. 346–361, Jun. 2014, doi: 10.1007/978-3-319-10578-9_23.

[57] M. D. Zeiler and R. Fergus, "LNCS 8689 - Visualizing and Understanding Convolutional Networks," 2014.

[58] W. Ouyang et al., "DeepID-Net: multi-stage and deformable deep convolutional neural networks for object detection," Sep. 2014, Accessed: Mar. 12, 2021. [Online]. Available: http://arxiv.org/abs/1409.3505.

[59] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabbouj, and D. J. Inman, "1D convolutional neural networks and applications: A survey," Mech. Syst. Signal Process., vol. 151, p. 107398, Apr. 2021, doi: 10.1016/j.ymssp.2020.107398.

[60] S. Kiranyaz, T. Ince, and M. Gabbouj, "Real-Time Patient-Specific ECG Classification by 1-D Convolutional Neural Networks," IEEE Trans. Biomed. Eng., vol. 63, no. 3, pp. 664–675, Mar. 2016, doi: 10.1109/TBME.2015.2468589.

[61] S. Kiranyaz, T. Ince, R. Hamila, and M. Gabbouj, "Convolutional Neural Networks for patient-specific ECG classification," in Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, Nov. 2015, vol. 2015-November, pp. 2608–2611, doi: 10.1109/EMBC.2015.7318926.

[62] H.-J. Yoo, "Deep Convolution Neural Networks in Computer Vision: a Review," IEIE Trans. Smart Process. Comput., vol. 4, no. 1, pp. 35–43, Feb. 2015, doi: 10.5573/ieiespc.2015.4.1.035.

[63] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in Proceedings of 2017 International Conference on Engineering and Technology, ICET 2017, Mar. 2018, vol. 2018-January, pp. 1–6, doi: 10.1109/ICEngTechnol.2017.8308186.

[64] A. Kaur and I. Kaur, "An empirical evaluation of classification algorithms for fault prediction in open source projects," Journal of King Saud University - Computer and Information Sciences, vol. 30, no. 1. King Saud bin Abdulaziz University, pp. 2–17, Jan. 01, 2018, doi: 10.1016/j.jksuci.2016.04.002.

[65] A. Rácz, D. Bajusz, and K. Héberger, "Multi-Level Comparison of Machine Learning Classifiers and Their Performance Metrics," Molecules, vol. 24, no. 15, Aug. 2019, doi: 10.3390/molecules24152811.

Chapter 5

Data Analysis: Quantitative Detection of Adulteration in Edible oils

5.1. Preamble

Quantitative analysis of edible oil adulterants is a critical health and economic concern that necessitates a quick, accurate, and comprehensive solution. This chapter describes the quantitative analysis of data acquired from analytical experiments such as MIR Spectroscopy with ATR sampling using chemometrics and artificial intelligence (AI) algorithms. For quantitative detection of adulteration in edible oils lab made adulterated samples have been used as reference. The first section of the chapter discusses the foundations of statistics and multivariate data analysis, as well as the methodology of several regression algorithms. This chapter's later sections focus on the development of AI based chemometric algorithms for the calibration of adulterations in edible oils using MIR spectroscopy with ATR sampling data. The performance of these algorithms in adulteration detection and quantification, as well as a suitable algorithm for embedded implementation, are also presented.

5.2. Quantitative analysis -regression methods

Regression analysis on acquired data is a major part of chemometric data analysis. The goal of a regression model is to predict the relationships between a set of p output variables $y = \{y_1, y_2 \ldots y_p\}$ to a set of q predictor variables $x = \{x_1 \ldots x_q\}$ for a given set of N training samples.

$$\{y_i, x_i\}_{i=0}^{N} = \{y_{1i}, y_{2i} \ldots y_{pi}, x_{1i} \ldots x_{qi}\}_{i=0}^{N} \quad (5.1)$$

A regression model is used for interpreting the data and as a prediction rule for calculating the likely values of the output variables for unknown input samples [1], [2]. The structural form of the linear predictive relationship is given by

$$y_i = b_0 + \sum_{i=1}^{q} a_{ki}\, x_i \qquad for\ k = 0\ to\ p \quad (5.2)$$

The input training data is used to calculate the coefficients b_0 and a_{ki}. in matrix notation, the regression model for predicting a response vector y is given as

$$y_i = \boldsymbol{a}_i^T \mathbf{x} \text{ for } i = 1 \; to \; p \tag{5.3}$$

$$\mathbf{y} = A\mathbf{x} \tag{5.4}$$

$A_{\mathrm{p\,x\,q}}$ is the matrix of regression coefficients. There are two popular regression methods: (i) Principal Component Regression and (ii) Partial Least Square Regression. A soft-computing based regression models are also useful in predicting the response variable with input predictor variables. In the following sections, Partial least square regression and ANN-based regression methods are discussed.

5.2.1. Principal components regression (PCR)

Principal component regression (PCR) is an alternative to multiple linear regression (MLR). PCR is based on principal component analysis that is used to determine the unknown regression coefficients in the linear regression process [3]. In MLR the regression equation can be written in a matrix form as

$$\mathrm{Y} = \mathrm{XB} \tag{5.5}$$

where Y is the dependent variable or response variable matrix, X represents the matrix of the independent variables, B is the regression coefficients to be estimated.

$$\mathrm{B} = (\mathrm{X^T X})^{-1}\,\mathrm{X^T Y} \tag{5.6}$$

The main drawbacks of MLR are that it cannot handle strongly correlated variables in X, missing values in the data set, and assumes X is noise free, which is practically not possible. Figure 5.1 shows a pictorial representation of MLR and PCR regression.

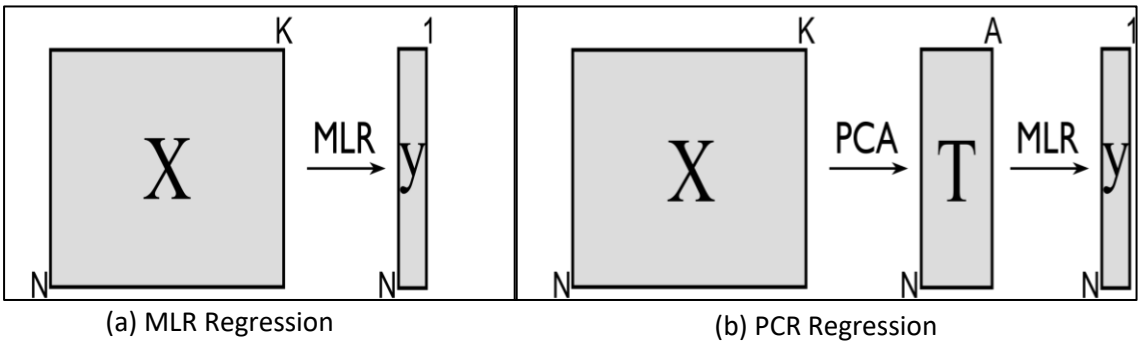


Figure 5.1 MLR and PCR Regression Overview

In PCR, PCA is employed for transforming the independent variable matrix (X) into principal loading matrix (P) (eigenvectors sorted in decreasing order of eigenvalues),

uncorrelated score matrix (T). The score matrix is then used as an independent variable for developing the regression model.

$$T = XP \tag{5.7}$$

$$Y = TB + E \tag{5.8}$$

In the ordinary least square's method, the coefficients are calculated as

$$B = (T^T T)^{-1} T^T Y \tag{5.9}$$

The columns in T (the scores from PCA) are orthogonal to each other, obtaining independence for the least-squares step. These T scores can be computed even if X includes missing data.

5.2.2. Partial least square regression (PLSR)

PLS regression is a generalization and combination of features from PCA and MLR algorithms. This idea was first introduced in the social sciences by Herman Wold et al. in 1966 but gained popularity first in chemometrics due to Herman's son Svante [4]. The principal components of X are used as independent variables in PCR to predict the response variable Y. This approach focuses solely on variables that describe X. Unlike PCR, PLSR identifies X components that are significant for Y also. PLSR decomposes the independent and dependent matrix into orthogonal factors and loadings.

$$\mathrm{X} = \mathrm{TP}^{\mathrm{T}} + \mathrm{E} \tag{5.10}$$

$$\mathrm{Y} = \mathrm{TQ}^{\mathrm{T}} + \mathrm{F} \tag{5.11}$$

Here T called score matrix, P and Q are the loading vectors (eigen vectors) of X and Y, respectively. The response vector Y can be expressed as

$$\mathrm{Y} = \mathrm{XB} + \mathrm{F} \tag{5.12}$$

where B is the regression coefficient matrix calculated as

$$B = XW(P^T W)^{-1} \tag{5.13}$$

In the present research work, the PLSR algorithm is used to predict the percentage of adulteration in edible oils. The regression model performance is evaluated by

calculating the Mean Squared Error (MSE), Coefficient of determination (R^2), and Root Mean Square of Error (RMSE). These terms are discussed in the next section.

5.2.3. ANN-based regression (ANNR)

Artificial neural networks (ANN), inspired by the biological human brain neural system to process information are a family of nonlinear computational and mathematical models [5]. The ANN paradigm, on a very simplistic and abstract basis, is based on the cognitive process of the human brain and, of course, is much simpler than the human brain system. A neural network is made up of a set of interacting processing neurons that operate in parallel. As shown in Figure 5.2, neurons in ANNs have weights that are assigned randomly during initialization. These weights are adjusted by means of an iterative or 'learning' process until the desired outputs are obtained in the network output. The final trained set of weights and functions are then saved as a 'neural network' in a supervised mode of learning process.

An ANN's fundamental structure consists of three layers: the input layer, the hidden layer, and the output layer. Each layer is comprised of several neurons. The input neuron layer receives the input data. The data for input will come from either an external source or from the sensor. The response from the network is expressed by the output neuron layer. One or more hidden layers may be used between the input and output layers.

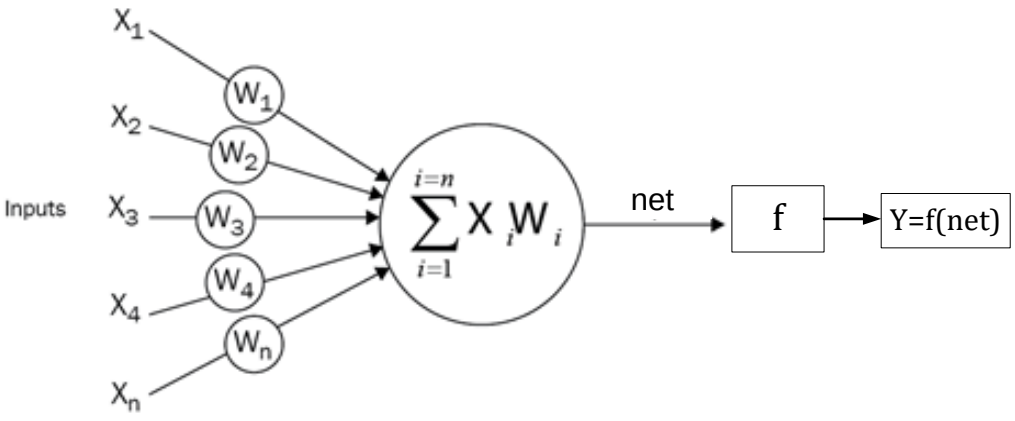


Figure 5.2 Architecture of neuron in neural network

In general, ANN algorithms are widely used for classification and regression applications [6], [7]. The outcomes in classification applications are categorical, while the response variables in regression applications are numerical. In this analysis, ANN

is used for regression purposes. Finding the best ANN model that can accurately predict the target while optimising many factors such as processing speed, numerical precision, and memory requirements is difficult. An optimization problem like this exists in the learning process of a neural network and can be solved by using an appropriate training algorithm. During the algorithm's training phase, ANN begins to identify patterns in the input data. It then compares the produced output to the target value. The difference between the two results is adjusted using a backward working process until it is less than a predefined criterion. As a result, choosing an appropriate training algorithm is critical when training a neural network. There are several types of training algorithms, but the most commonly used ones include Levenberg–Marquardt (LM), quasi-Newton (QN), conjugate gradient (CG), and gradient descent (GD). A particular training algorithm may be excellent for one situation but fail in another. GD training method is the slowest of the other mentioned training algorithms, but it requires less memory. LM is the fastest algorithm, but it uses the most memory. As a result, determining the best training algorithm in general and predicting percentage of adulteration requires a thorough investigation. Different metrics are used in the field of machine learning to evaluate the performance of models. The correlation coefficient (R) or coefficient of determination (R^2), mean absolute error (MAE), and root mean square error (RMSE) as well as the mean absolute percentage error (MAPE) are all examples for this metrics. The Training algorithms are discussed in the next sections.

A. Gradient Descent Algorithm (GD)

Gradient descent is an adaptive optimization technique used in machine learning and deep learning applications to determine a collection of internal variables for model optimization. Here "gradient" refers to the rate of inclination or declination of slope, while "descent" refers to decreasing. Gradient descent is performed in three steps: (1) variable initialization, (2) model evaluation using the variable and loss function, and (3) updating variables in the direction of finding optimal locations. The gradient descent method employs iteration by the following equation

$$Y_{j+1} = Y_i - \nabla(f(Y_i) * \alpha \tag{5.14}$$

Where (Y_i) variables that are to be updated according to the loss function $\nabla(f(Y_i)$ and α is the learning rate. The nature of the loss function optimization method is to locate optimal places to reduce or maximize the loss function. The GD method's objective is to identify such global minimum locations. The stop criterion for the training process might be when maximum number of training epochs reached, or the loss function's

value is modest enough, or the model's accuracy is sufficient, or the value of the loss function stays stable after several iterations.

B. Levenberg–Marquardt Algorithm (LM)

Levenberg–Marquardt (LM) algorithm, also known as the damped least-squares method, is used to train nonlinear least-squares problems. This algorithm determines the optimization with the gradient vector and the Jacobian matrix. The loss function of the LM method is expressed as a sum of squared errors, expressed below, where p is the number of instances in the data set and e is the vector of all error terms.

$$loss\ function(f) = \sum_{i=0}^{p} {e_i}^2 \tag{5.15}$$

The Jacobian matrix for the loss function is defined as

$$J_{i,j} = \frac{\partial e_i}{\partial w_i}\ for\ i = 1,2, \dots p\ and\ j = 1,2,3, \dots q \tag{5.16}$$

J is the Jacobian matrix with size (p, q),p is the number of instances in the data set and q is the number of parameters in the network. For this loss function, the gradient loss is calculated as

$$\nabla f = 2.J^T.e \tag{5.17}$$

The parameter improving process in the LM method is given by the following equation,

$$Y_{i+1} = Y_i - \left({J_i}^T J_i + \beta_i I\right)^{-1}(2.{J_i}^T.e) \tag{5.18}$$

here the β is the damping factor. Initially, β is taken as a large value, and if there is an error in the iterations, its value is increased by some factor. If the loss decreases, then β value will be decreased.

C. Quasi-Newton Method

The quasi-Newton technique has the benefit of being computationally cheap since it does not require many operations to assess the Hessian matrix and calculate the associated inverse. At each step, an approximation value to the inverse Hessian matrix is constructed. It is calculated only on the basis of the loss function’s first derivatives.

The Hessian matrix is made up of the loss function's second partial derivatives. The quasi-Newton formula is given by

$$Y_{i+1} = Y_i - (G_i g_i)\alpha \tag{5.19}$$

here the G represents the inverse Hessian approximation and α is the learning rate. The Newton method is faster than GD and conjugate gradient method, so it is frequently used for training ANN.

D. Conjugate Gradient Method

The conjugate gradient algorithm, which is midway between gradient descent and Newton's technique, might be regarded one of the methods to enhance the convergence rate of the artificial neural network. The benefit of this technique is that there is no need to calculate, preserve, or reverse the Hessian matrix. The search in this technique is conducted in conjunction with conjugate directions, which generate usually quicker convergence than gradient descent directions. These training instructions are conjugated with respect to the Hessian matrix. The series of training directions in this method is created using the following formula:

$$Y_{i+1} = v_{i+1} + Y_i c_i \tag{5.20}$$

Where the Y_0 is the training direction vector with initial conditions $Y_0 = v_0$ and c is the conjugate parameter. The parameter improvement method in the conjugate gradient algorithm is defined by the following equation.

$$W_{i+1} = W_i + Y_i \alpha \tag{5.21}$$

where i =0, 1,…. , and α is the learning rate. To detect the percentage of adulteration in edible oils, a neural network architecture with one input layer, one hidden layer, and one output layer was used. Figure 5.3 depicts the regression model architecture. For training the ANN regression, neural network fitting tool with LM, conjugate gradient and quasi-Newton algorithms have been employed. The results of these algorithms for the quantification of percentage of adulterations in edible oils (ground nut and sesame oils have been presented in the results section.

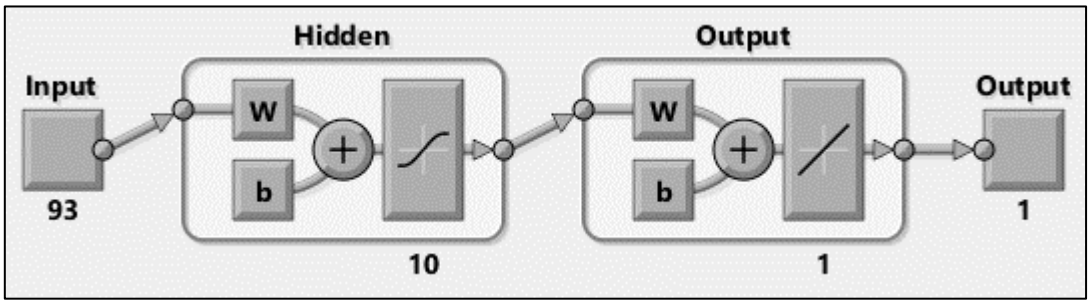


Figure 5.3. Architecture of ANN regression model

5.2.4. Successive Projective Algorithm-Regression Method

The Successive Projections Algorithm (SPA) is a useful tool for variable selection in the multivariate calibration and classification task. The goal of SPA is to find a small representative set of spectral variables with an aim of minimizing collinearity [8]. SPA employs a calibration (Xc) and validation (Xv) set consisting of instrumental response data (X) and parameter values measured by a reference method in multivariate calibration problems (Y). The projection operations performed on the calibration matrix Xc (N_{cal} x K)), whose rows correspond to N_{cal} calibration samples and columns correspond to K spectral variables, respectively, are at the core of SPA. SPA_MLR has three phases. These column vectors are subjected to a series of projection operations, resulting in the formation of K variable chains. The k^{th} chain is started with variable x_k and is progressively accompanied with variables that have the least collinearity with the previous ones.

For a simple case with N_{cal} = 3 samples and K = 5 variables, such collinearity is evaluated in terms of the associated column vectors, as shown in Figure 5.4 . In this case, matrix Xc comprises five-column vectors $\{x_1, x_2, x_3, x_4, x_5, \}$. Taking x_3 as starting point remaining vectors are projected on to the plane perpendicular to the x_3 resulting in projections Px_1, Px_2, Px_4, Px_5. Among them Px_1 has the largest projection and the second-largest projection is Px_5 as shown in Figure 5.4. Hence x_1 and $x_5$are added to the chain. The resulting chain of variables starting from x_3 is, therefore (x_3, x_1, x_5). Similarly, four other chains of variables can be constructed.

In the second phase of SPA, candidate subsets of variables extracted from the chains created in Phase 1 are evaluated. By taking the L first variables of each chain, candidate subsets with L variables are obtained. The best subset of variables is chosen based on the validation set's smallest root-mean-square error (RMSEV). This performance metric is calculated as follows,

$$RMSEV = \sqrt{\frac{1}{N_{val}} \sum_{i=1}^{N_{val}} \left(y_j - \hat{y}_j \right)^2} \tag{5.22}$$

The third phase entails a backward elimination procedure aimed at removing uninformative variables and thus improving the model's parsimony.

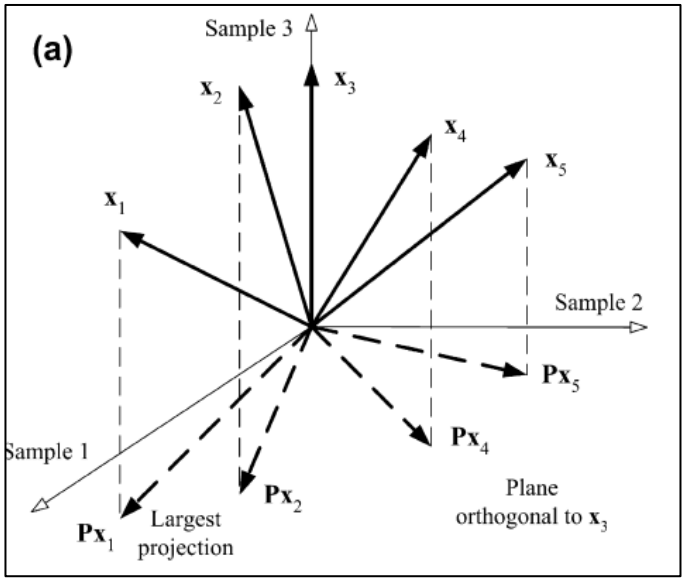


Figure 5.4 Illustration of successive projection methodology

For classification task, Y data consist of class index of each sample and Linear discriminant analysis's cost function from the average risk of misclassification is employed [8] . The cost function is given by

$$G = \frac{1}{K} \sum_{k=1}^{K} g_k \tag{5.23}$$

where g_k (risk of misclassification of the k^{th} validation object x_k) is defined as

$$g_k = \frac{r^2(x_k, \mu_k)}{\min_{j \neq k} r^2(x_k, \mu_j)} = \frac{(x_k - \mu_k) \sum^{-1} (x_k - \mu_k)^T}{\min_{j \neq k} (x_k - \mu_j) \sum^{-1} (x_k - \mu_j)^T} \tag{5.24}$$

A successive projection technique is employed for variable selection from MIR spectroscopy data with ATR sampling, and the quantity of adulterations in edible oil has been calibrated.

5.2.5. Performance indices for a regression model

The performance of a regression model is assessed using the following parameters.

(a) Coefficient of determination (R^2)

The coefficient of determination explains the variation of one variable with the variation of another, which can be explained by the regression equation [9], [10]. This correlation, known as the "goodness of fit," is represented as a value between 0.0 and 1.0.

$$R^2 = 1 - \frac{\sum_{i=0}^{n} (y_i - f_i)^2}{\sum_{i=0}^{n} (y - \bar{y})^2} \tag{5.25}$$

(b) Mean squared error (MSE)

Mean squared error is the average squared difference between the estimated values and the actual value.

$$MSE = \frac{\sum_{i=0}^{n} (y_{actual} - y_{predicted})^2}{n} \tag{5.26}$$

where n is the number of data points.

(c) Root mean squared error (RMSE)

It is the square root of the average squared difference between predicted and actual values, and it has the same unit as actual/predicted values.

$$RMSE = \sqrt{\frac{\sum_{i=0}^{n} (y_{actual} - y_{predicted})^2}{n}} \tag{5.27}$$

5.3. Results and discussion: quantitative analysis of adulteration

This section presents how the aforementioned machine learning techniques have been applied to data acquired from MIR spectroscopy with ATR sampling experiments to quantify adulterations in edible oils.

An experiment for detecting adulterations in edible oils was designed using attenuated total reflection sampling in the MIR region. The ATR sampling method requires no sample preparation and is suited for both solid and liquid sample analysis. As mentioned in the preceding chapter, lab-made adulterated samples in the proportions of 5%(v/v), 10%, 25%, 50%, and 75% were utilised for experimentation. AI algorithms like Principal component Regression (PCR), Partial Least Square Regression (PLSR), Successive Projection Algorithm-Multiple Linear Regression (SPA-MLR), and Artificial Neural Networks Regression (ANNR) for adulteration quantification have been developed, and the results are reported in the sections that follow. Three

adulteration case studies have been presented: I sunflower oil adulteration with palm oil, (ii) groundnut oil adulteration with cottonseed oil, and (iii) sesame oil adulteration with cottonseed oil.

5.3.1. Palm oil adulteration in Sunflower oil

As discussed in the previous chapters the functional group associated with -C=O (ester) carbonyl group from ester linkage of triacylglycerol is attributed to the wavenumber of 1745 cm^{-1}. The other functional groups/ bonds in fatty acids, for example -C-H-(CH2, CH3), =C-H-(cis), -C-O-CH2-, -C-H-, and -C-H-bending have been reported to be attributed for wavenumbers 1460, 1377, 1161, 1061, 1117 and 1097 cm^{-1} respectively [11], [12], [13]. In the present work, the spectrum observed in the regions of 1786-1680 cm^{-1} could be because of the ester group of the triglycerides present in the edible oils, while the 1490-915 cm^{-1} could be because of fatty acid functional groups. 1400 to 1097 cm^{-1} region is also represented as fingerprint regions [14], [15].

The data collected in the mid-infrared region (1751-900 cm-1) using sunflower oil adulteration with palm oil (in proportions of 5%, 10%, 15%, 25%, 50%, and 75%) is split into three sets of spectra variables corresponding to wavenumbers designated R1 and R2. R3 was chosen as the third wavenumber region based on the contribution of variance by variables from the PCA correlation loading plot of the complete data set. Table 5.1 illustrates the wave number locations chosen for quantitative analysis.

Table 5.1 Spectral regions selected for analysis of adulteration in edible oils

S. No	Selected Wavenumber range (cm^{-1})	Indicator
1	1492-937	R1
2	1781 to 1635	R2
3	{1717 to1581, 1501 to 1447, 1372 to 1334, 1263 to 937}	R3

5.3.1.1. Regression results of adulteration in sunflower oil

To calibrate the percentage of palm oil adulterated in sunflower oil, the partial least square regression (PLSR), ANN-based regression (ANNR), and successive projection algorithm-MLR models have been developed. Table 5.2 shows the PLSR regression results for the selected regions, i.e., R1, R2, and R3, and Figures 5.5 ,5.6 and 5.7 show

the regression plots between actual and predicted values. Regression results in R3 are better than R1 and R2 regions. The coefficient of regression was 0.99 and RMSE was 4.01.

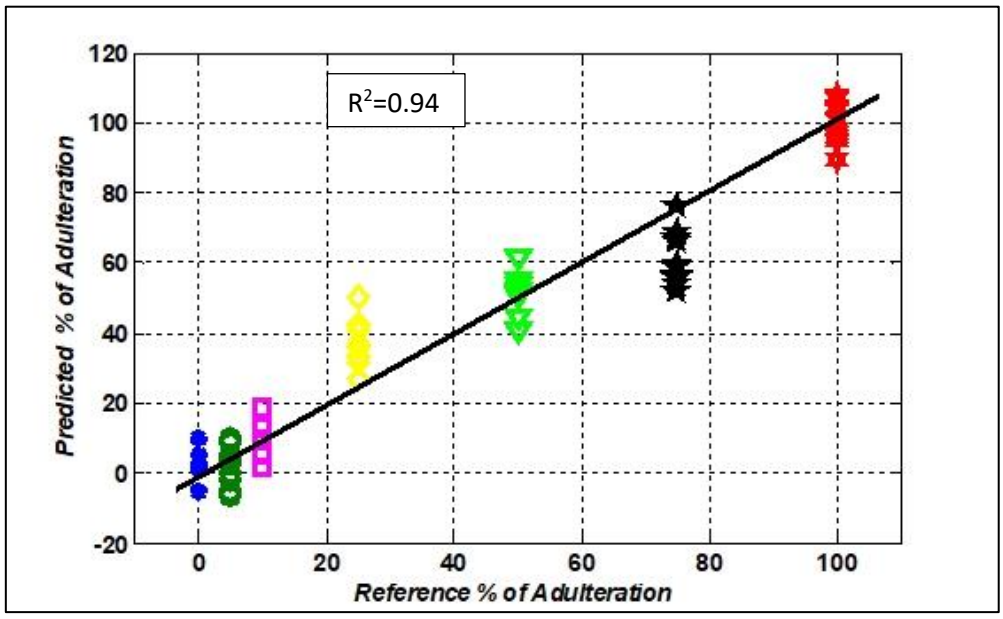


Figure 5.5 Regression plot (reference vs predicted) for sunflower oil adulteration with palm oil in R1 (1781 to 1635cm^{-1})

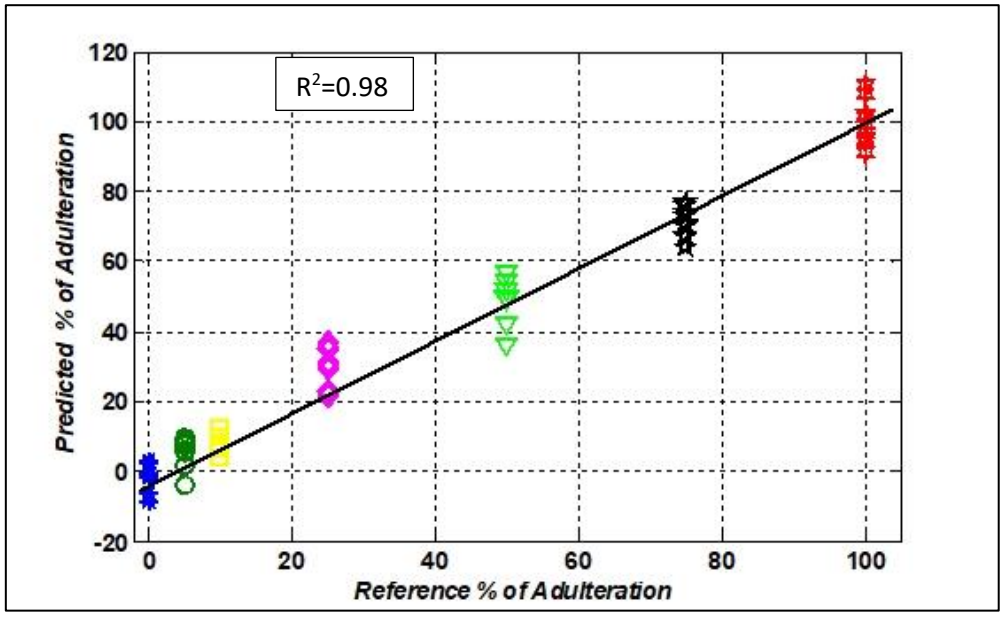


Figure 5.6 Regression plot (reference vs. predicted) for sunflower oil adulteration with palm oil in R2 (1492-937 cm^{-1})

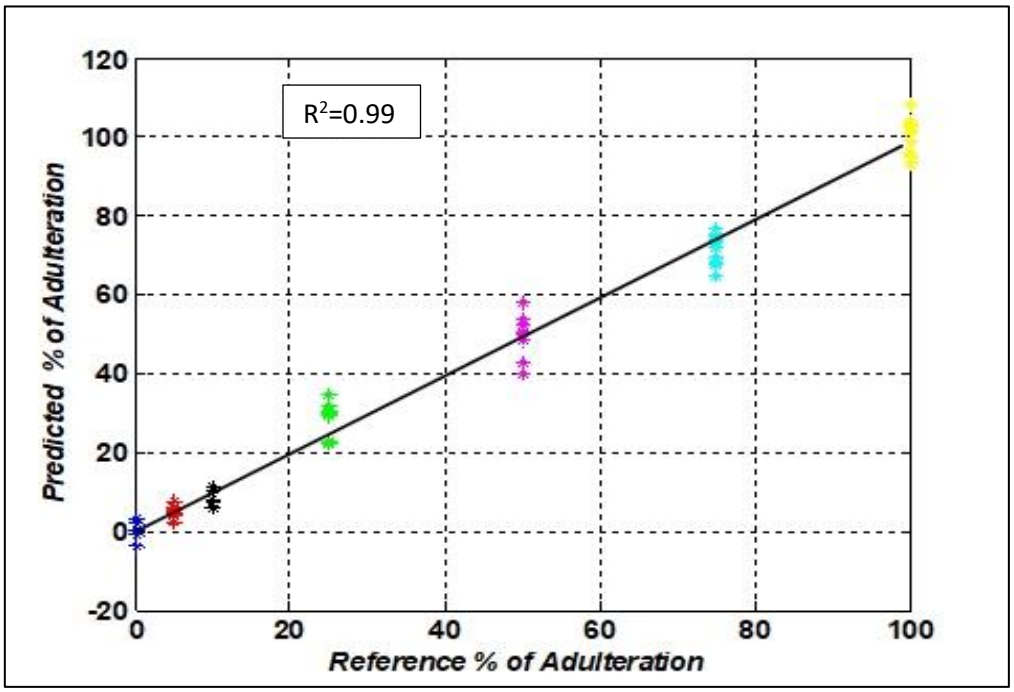


Figure 5.7 Regression plot (reference vs. predicted) for sunflower oil adulteration with palm oil in R3

Table 5.2 calibration results for adulteration of sunflower oil with palm oil

wavenumber (cm^{-1})	**Calibration**		**Validation**	
	R^2	**RMSE**	R^2	**RMSE**
R1(1492-937)	0.98	4.80	0.98	4.66
R2(1781-1635)	0.94	8.70	0.94	8.62
1717-1581, 1501-1447, 1372-1334, 1263-937	0.99	4.01	0.99	4.24

Another method for the quantification of the percentage of palm oil in sunflower oil, a variable selection method based on Successive Projection (SPA) was developed, followed by MLR on the selected variables. Input data is divided into training (48 samples), validation (16 samples), and testing (16 samples) sets in proportions of 70%, 15%, and 15%, respectively. The SPA algorithm was used to select 20 variables from each sample so that the RMSE of the regression model was as minimal as possible. Figure 5.8 depicts the number of selected variables for sunflower adulteration data based on the RMSE. Using these variables from each sample, a multiple linear

regression model is developed. Figure 5.9 depicts the results of regression models in terms of RMSE (5.79), standard deviation (5.77), and coefficient of regression (0.98).

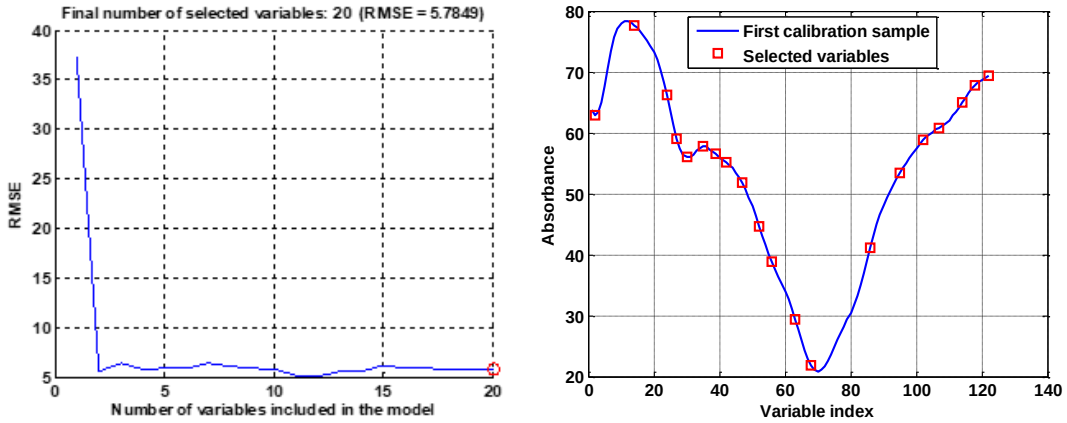


Figure 5.8 Successive projection Algorithm for variable selection

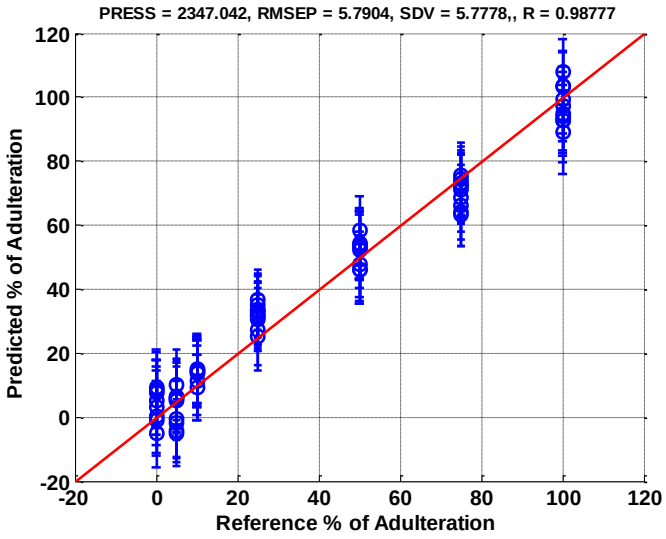


Figure 5.9 Regression results with variable selected using Successive projection Algorithm

5.3.2. Groundnut oil adulterated with cottonseed oil

Cottonseed oil is used as adulteration in groundnut oil and adulterated samples were prepared in the lab in proportions of 5%(v/v), 10%, 25%,50%, and 70%. Pure groundnut oil is considered as 0% adulterated sample and pure cottonseed oil is as 100% adulterated sample while developing a calibration model. ATR sampling method with these edible oil samples resulted in 105x128 data with 15 samples for each class.

The adulteration levels of cottonseed oil in groundnut were calibrated by the regression models like the partial least square regression (PLSR), Principal component regression (PCR), ANN-based regression (ANNR), and Successive projection-MLR. The regression results of PLSR and PCR are given in the Table 5.2 and 5.3. PLSR results were comparatively good with RMSE of 4.4 and R^2 value of 0.98. As compared to PLSR results PCR results were a bit inferior with RMSE of 6.4 and R^2 of 0.96. So PLSR regression inference model was implemented on the embedded platform for detection of adulteration level. Figures 5.10, 5.11 and 5.12 show the PLSR regression results.

Table 5.3 Calibration results of PLSR for adulteration in groundnut oil

Wavenumber (cm^{-1})	**Calibration**		**Validation**	
	R^2	**RMSE**	**R^2**	**RMSE**
1786-1680	0.96	7.5	0.95	7.8
1490-924	0.99	3.8	0.99	4.2
(1270-937)	0.98	4.4	0.98	4.8

Table 5.4 PCR calibration results for adulteration in groundnut oil with cottonseed oil

wavenumber (cm^{-1})	**PCR Calibration**		**PCR Validation**	
	R^2	**RMSE**	***R^2***	**RMSE**
1781-1635	0.97	5.992	0.97	6.1
1492-937	0.96	6.4	0.96	6.52
1270-937	0.974	5.70	0.97	5.78

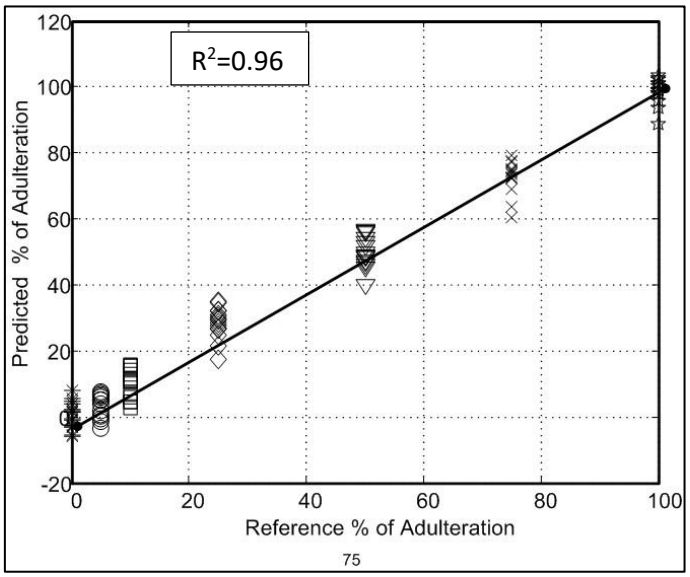


Figure 5.10 Regression plot (reference vs predicted) for groundnut oil adulteration with cotton seed oil in the spectral range R1 (1781 to 1635cm^{-1})

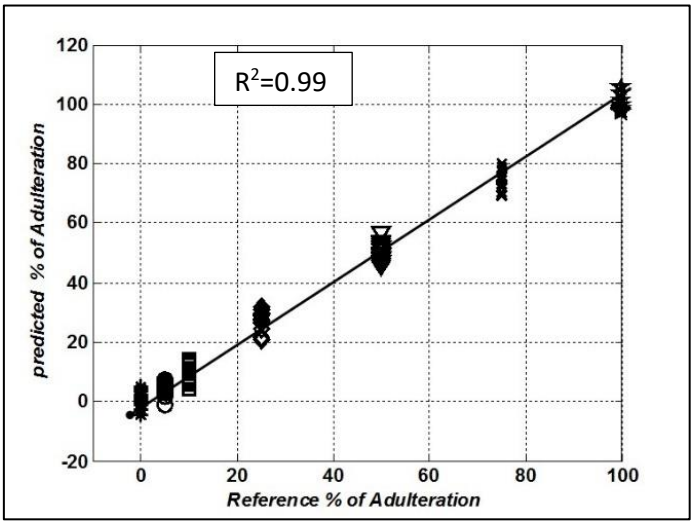


Figure 5.11 Regression plot (reference vs. predicted) for groundnut oil adulteration with cottonseed oil in the spectral range R2(1492-937cm^{-1})

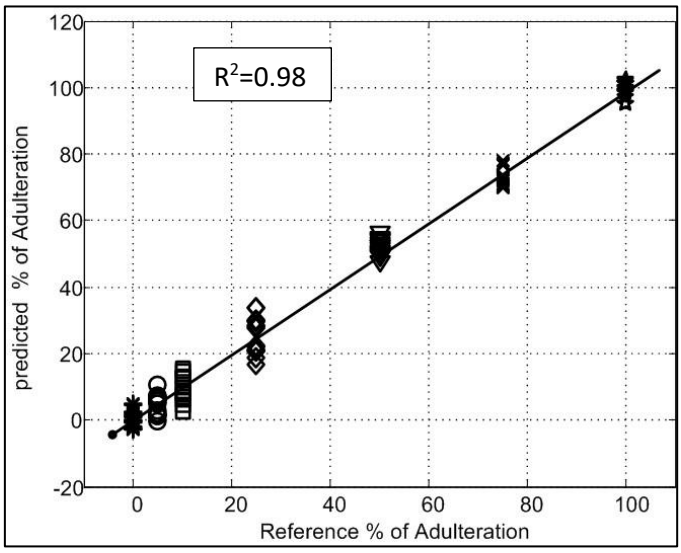


Figure 5.12 Regression plot (reference vs. predicted) for groundnut oil adulteration with cottonseed oil in the spectral range R3

A variable selection method using Successive Projection Algorithm (SPA) was developed for the calibration of Groundnut oil adulteration, followed by Multiple Linear Regression (MLR) on the selected variables. Input data is divided into training (48 samples), validation (16 samples), and testing (16 samples) sets in proportions of 70%, 15%, and 15%, respectively. Using successive projection method 15 variables were selected for which RMSE is minimum. Figure 5.13 depicts the number of selected variables for groundnut adulteration data based on the RMSE (3.7095). Using these variables from each sample, a multiple linear regression model is developed. Figure 5.14 depicts the results of regression models in terms of RMSE (3.716), standard deviation (3.169), and coefficient of regression (0.99).

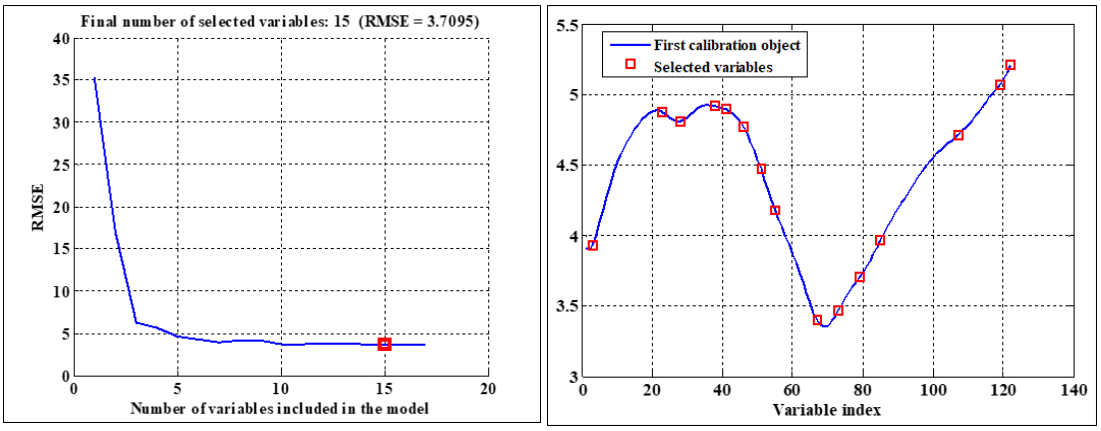


Figure 5.13 Successive projection algorithm for variable selection

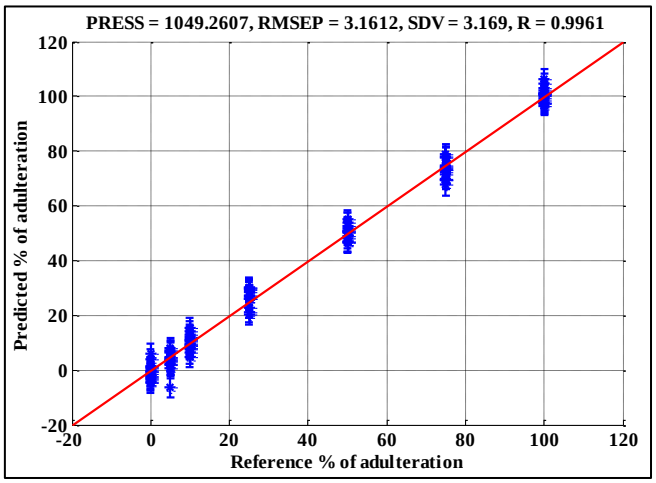


Figure 5.14 Regression result from variables selected using Successive projection algorithm

The percentage of cottonseed oil in groundnut oil was also calibrated using ANN-based regression analysis. A single hidden layer with 10 input nodes was used. The total input data matrix is divided into 70% training data, 15% testing data, and 15% validation data. The ANN regression model trained with different training epochs for different training algorithms such as, gradient descent with adaptive learning rate GDA), resilient backpropagation (RP), conjugate gradient (CG), and Lavenberg-Marquardt (LM) algorithms. The performances of the training models are presented in terms of mean squared error on this training data with LM algorithm yielded R^2 0.99 and an MSE of 3.088 as shown in Table 5.4. Figure 5.15 shows the calibration results of ANN regression for adulteration detection in edible oils in case of ANN LM training algorithm.

Table 5.5 Comparison of performance of ANN regression training algorithms

S. No	**Training Algorithm**	**Epochs (Converged)**	**Training**		**Validation**	
			(R)	**MSE**	**(R)**	**MSE**
1	Levenberg -Marquardt (LM)	20	0.998	$2.48e^{-10}$	0.998	3.088
2.	GD with Adaptive Learning (GDA)	10000	0.999	0.0903	0.998	3.13
3.	Bayesian Regularization (BR)	175	0.999	$1.414e^{-12}$	0.995	15.4
4	Scaled Conjugate Gradient (SCG)	70	0.999	0.787	0.997	6.03
5	Resilient backpropagation (RP)	1085	0.999	4.3	0.998	7.09

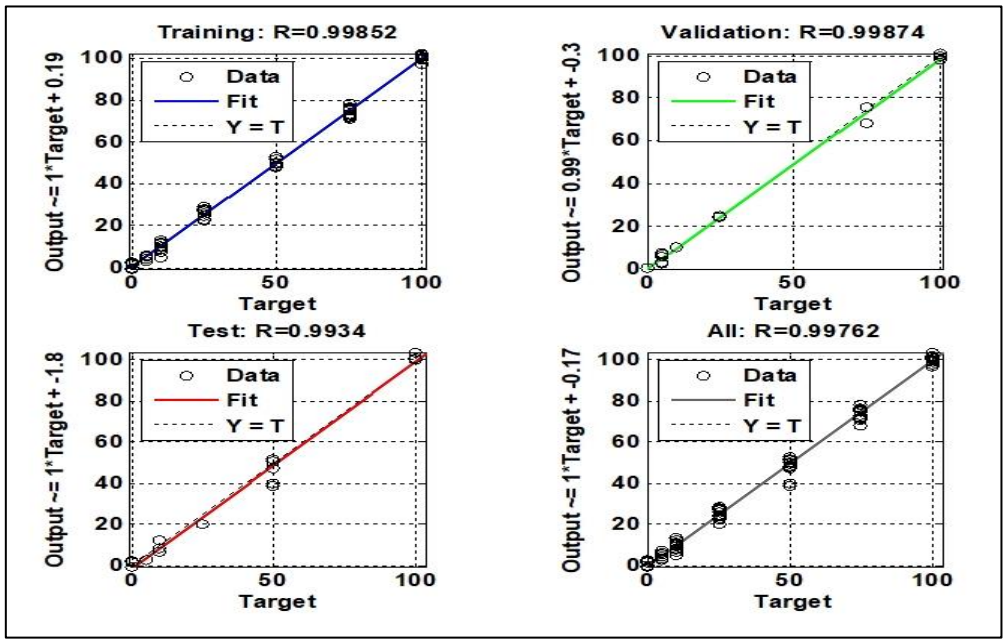


Figure 5.15 ANN regression results for calibrating percentage of adulteration in groundnut oil

5.3.3. Sesame oil adulterated with cottonseed oil

Cottonseed oil is used as adulteration in sesame oil. The adulterated samples were prepared in the lab in proportions of 5%(v/v), 10%, 25%,50%, and 70%. Pure sesame oil is considered as 0% adulterated sample and pure cottonseed oil is as 100% adulterated sample while developing a calibration model. ATR spectroscopy with these edible oil samples resulted in 105x128 data with 15 samples for each class.

The adulteration levels of cottonseed oil in sesame were calibrated by the regression models like the partial least square regression (PLSR), Principal component regression (PCR), ANN-based regression (ANNR), and Successive projection-MLR. The regression results of PLSR and PCR are given in the Tables 5.4 and 5.5. PLSR results are comparatively good with RMSE of 5.09 and R^2 value of 0.98. As compared to PLSR results PCR results (RMSE 7.3 and R^2 0.96) are a bit inferior to PLSR results. So PLSR regression inference model was implemented on the embedded platform for detection of adulteration level.

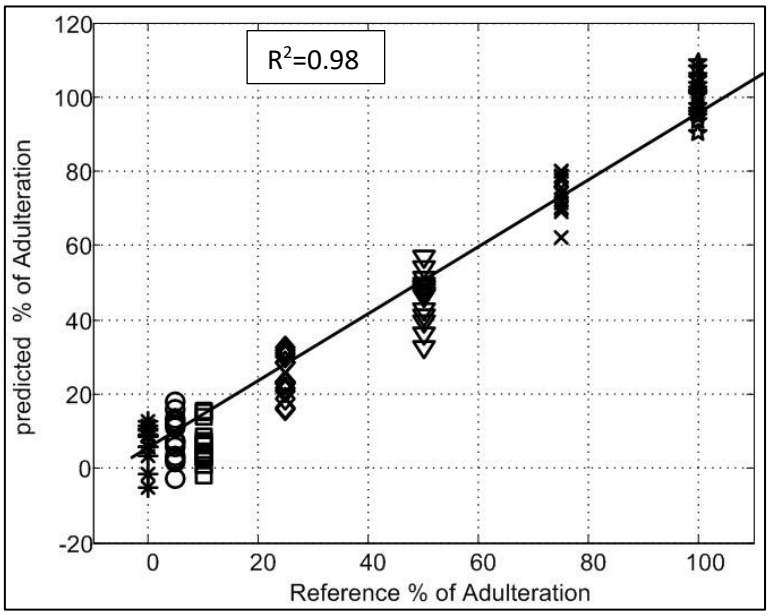


Figure 5.16 regression plot (reference vs. predicted) for sesame oil adulteration with cottonseed oil in the spectral range R1 (1781 to 1635cm^{-1})

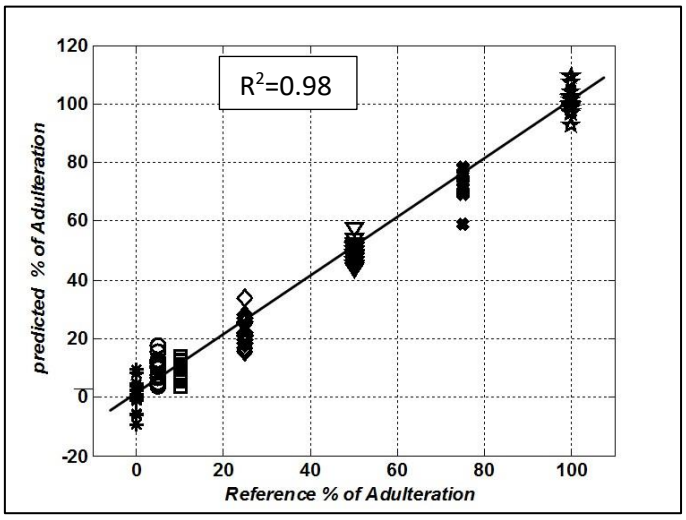


Figure 5.17 Regression plot (reference vs. predicted) groundnut oil adulteration with cottonseed oil in the spectral range R2 (1492-937cm^{-1})

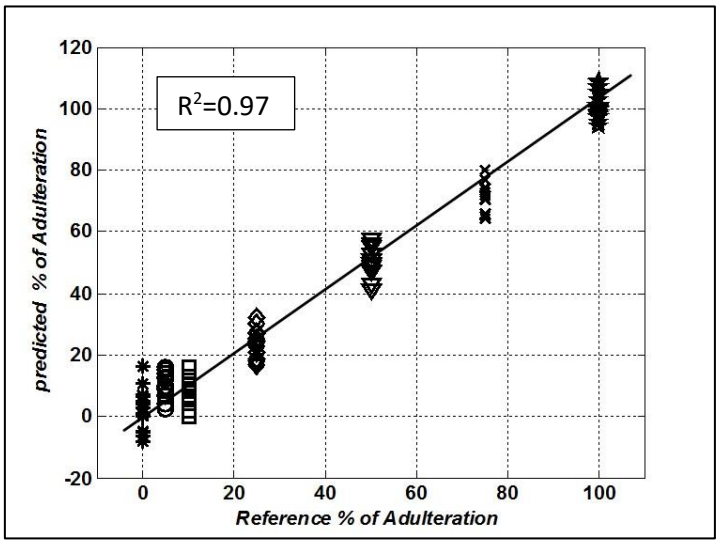


Figure 5.18 Regression plot (reference vs. predicted) for groundnut oil adulteration with cottonseed oil in the spectral range selected from correlation loading plot

Table 5.6 PLSR calibration results

Wavenumber (cm^{-1})	***PLSR Calibration***		***PLSR Validation***	
	R^2	***RMSE***	***R^2***	***RMSE***
1786-1680	0.98	5.4	0.98	5.8
1490-924	0.98	5.09	0.98	5.7
(1717-1581, 1501-1447, 1372-1334, 1263-937)	0.97	6.5	0.97	7.3

Table 5.7 PCR calibration results

wavenumber (cm^{-1})	**PCR Calibration**		**PCR Validation**	
	R^2	***RMSE***	***R^2***	***RMSE***
1781-1635	0.95	7.36	0.95	7.6
1492-937	0.966	6.53	0.96	6.68
1717-1581, 1501-1447, 1372-1334, 1263-937	0.969	6.20	0.96	6.28

A variable selection method using successive projection (SPA) was developed for the calibration of Sesame oil adulteration, followed by MLR on the selected variables. Input data is divided into training (48 samples), validation (16 samples), and testing (16 samples) sets in proportions of 70%, 15%, and 15%, respectively. Using successive projection methods 17 variables selected based on cross-validation results with

minimum RMSE. Figure 5.19 depicts the number of selected variables for sesame oil adulteration data based on the RMSE (3.972). Using these variables from each sample, a multiple linear regression model is developed. Figure 5.20 shows the results of regression models in terms of RMSE (3.97), standard deviation (3.98), and coefficient of regression (0.99).

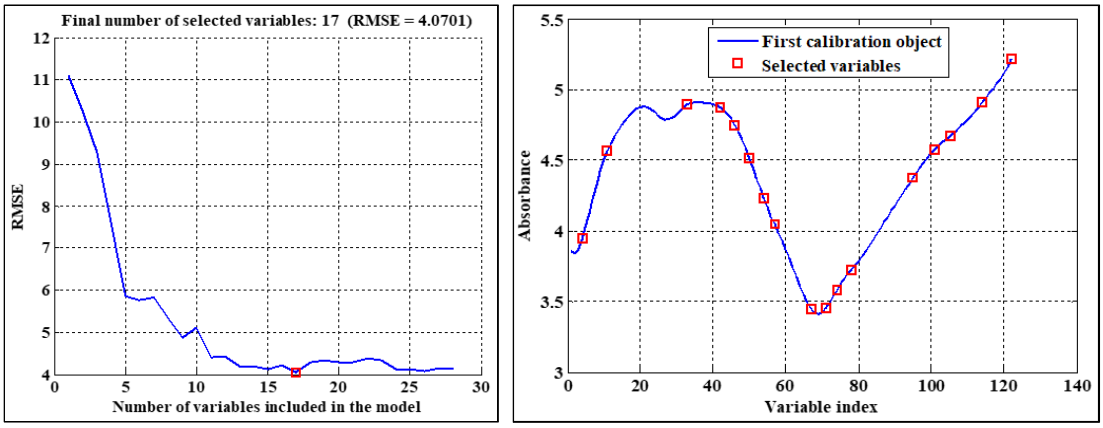


Figure 5.19 Variable selection with successive projection algorithm

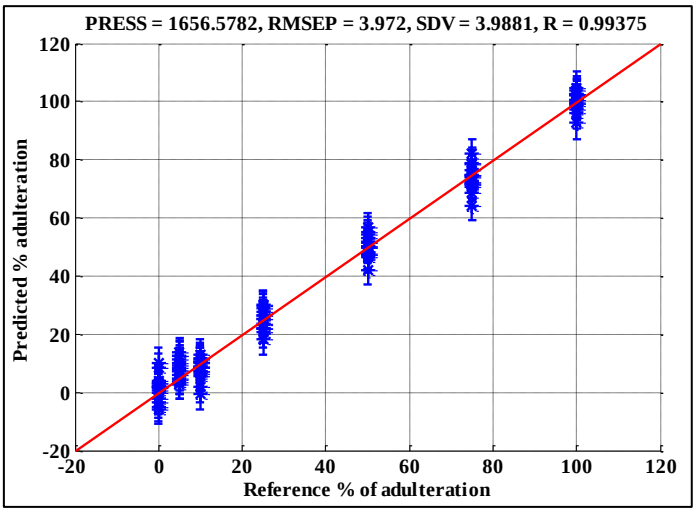


Figure 5.20 Regression result from variables selected using Successive projection algorithm

ANN regression model with different training algorithms have been trained on the data of Cottonseed oil adulterated in Sesame oil resulted. The results of each trained model have been presented in Table 5.7. It is observed that in all ANN fit model correlation coefficient is observed as 0.99. In case of LM and BR method calculated

MSE is 2.61 and 1.87 respectively . Figure 5.21 shows the calibration results of ANN regression for adulteration detection in edible oils with LM algorithm.

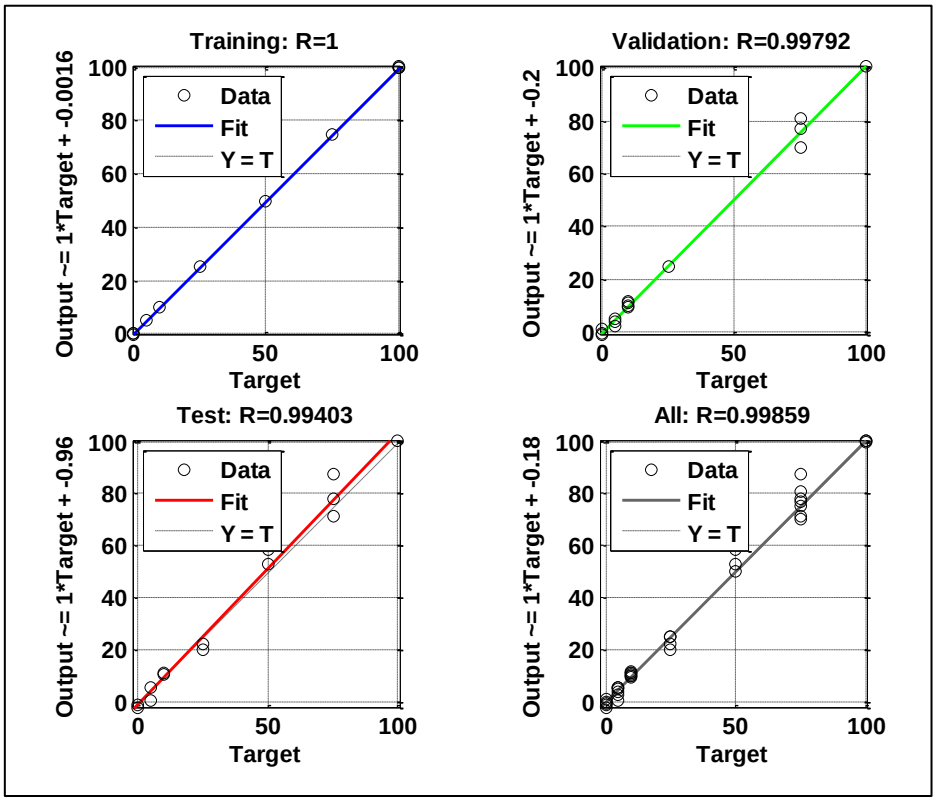


Figure 5.21 ANN regression results for calibrating percentage of adulteration in sesame oil.

Table 5.8 Comparison of performance of ANN regression training algorithms

S. No	Training Algorithm	Epochs	Training		Validation	
			(R)	MSE	(R)	MSE
1	Levenberg -Marquardt (LM)	80	0.999	$7.44e^{-3}$	0.998	2.614
2.	GD with Adaptive Learning (GDA)	10000	0.999	1.891	0.998	4,63
3.	Bayesian Regularization (BR)	1749	0.999	0.540	0.995	1.878
4	Scaled Conjugate Gradient (SCG)	1463	0.999	0.5409	0.998	3.4362
5	Resilient backpropagation (RP)	1085	0.999	3.875	0.998	5.73

5.4. Summary

The fundamentals of statistics, as well as an overview of machine learning and artificial intelligence for regression analysis, was covered at the outset of this chapter. The methodology of data analysis algorithms for adulteration detection was presented.

For calculating the percentage of adulteration in edible oils, regression models based on PLSR, SPA-MLR, and ANN were developed. ANN regression model trained with different training algorithms and their correlation coefficients and MSE values have been compared. The PLSR method and ANN produced the highest R^2 and lowest RMSE with the developed regression model. The parameters of the PLSR and ANN models were saved in a readable CSV file format. These files were used in the development of generic C language inference algorithms as well as their embedded implementation. These implementations will assist in the development of intelligent portable instrumentation for the analysis of edible oils. The implementation phase is covered in the following chapter 6. MIR spectroscopy with ATR sampling has been identified as the simplest and perhaps most accurate method for edible oil analysis in this research. The developed algorithm results show the algorithm's capability in classification and adulteration calibration in edible oils.

Bibliography

[1] T. Frost, "Quantitative analysis," in Encyclopedia of Spectroscopy and Spectrometry, Elsevier, 2016, pp. 811–815.

[2] G. Hanrahan, F. Udeh, and D. G. Patil, "Chemometrics and Statistics - Multivariate Calibration Techniques," in Encyclopedia of Analytical Science: Second Edition, Elsevier Inc., 2004, pp. 27–32.

[3] I. E. Frank and J. H. Friedman, "American Society for Quality A Statistical View of Some Chemometrics Regression Tools A Statistical View of Some Chemometrics Regression Tools," 1993.

[4] S. Wold, M. Sjostrom, L. Eriksson, and S. Sweden˚˚, "PLS-regression: a basic tool of chemometrics." Accessed: Mar. 12, 2021. [Online]. Available: www.elsevier.comrlocaterchemometrics.

[5] M. Mishra and M. Srivastava, "A view of Artificial Neural Network," 2014, doi: 10.1109/ICAETR.2014.7012785.

[6] N. Tehlah, P. Kaewpradit, and I. M. Mujtaba, "Artificial neural network based modeling and optimization of refined palm oil process," Neurocomputing, vol. 216, pp. 489–501, Dec. 2016, doi: 10.1016/j.neucom.2016.07.050.

[7] M. Soltani Firouz, M. Omid, M. Babaei, and M. Rashvand, "Dielectric spectroscopy coupled with artificial neural network for classification and quantification of sesame oil adulteration," Inf. Process. Agric., May 2021, doi: 10.1016/j.inpa.2021.05.001.

[8] S. F. C. Soares, A. A. Gomes, M. C. U. Araujo, A. R. G. Filho, and R. K. H. Galvão, "The successive projections algorithm," TrAC - Trends in Analytical Chemistry, vol. 42. Elsevier B.V., pp. 84–98, Jan. 01, 2013, doi: 10.1016/j.trac.2012.09.006.

[9] M. J. C. Pontes et al., "The successive projections algorithm for spectral variable selection in classification problems," Chemom. Intell. Lab. Syst., vol. 78, no. 1, pp. 11–18, Jul. 2005, doi: 10.1016/j.chemolab.2004.12.001.

[10] P. Schober and L. A. Schwarte, "Correlation coefficients: Appropriate use and interpretation," Anesth. Analg., vol. 126, no. 5, pp. 1763–1768, May 2018, doi: 10.1213/ANE.0000000000002864.

[11] N. Vlachos, Y. Skopelitis, M. Psaroudaki, V. Konstantinidou, A. Chatzilazarou, and E. Tegou, "Applications of Fourier transform-infrared spectroscopy to edible oils," Anal. Chim. Acta, vol. 573–574, pp. 459–465, Jul. 2006, doi: 10.1016/j.aca.2006.05.034.

[12] M. D. Guillén and N. Cabo, "Infrared spectroscopy in the study of edible oils and fats," Journal of the Science of Food and Agriculture. 1997, doi: 10.1002/(SICI)1097-0010(199709)75:1<1::AID-JSFA842>3.0.CO;2-R.

[13] N. Upadhyay, P. Jaiswal, and S. N. Jha, "Detection of goat body fat adulteration in pure ghee using ATR-FTIR spectroscopy coupled with chemometric strategy," J. Food Sci. Technol., vol. 53, no. 10, pp. 3752–3760, Oct. 2016, doi: 10.1007/s13197-016-2353-2.

[14] C. Socaciu et al., "Attenuated Total Reflectance-Fourier Transform Infrared Spectroscopy (ATR-FTIR) Coupled with Chemometrics, to Control the Botanical Authenticity and Quality of Cold-Pressed Functional Oils Commercialized in Romania," Appl. Sci., vol. 10, no. 23, p. 8695, Dec. 2020, doi: 10.3390/app10238695.

[15] I. Sota-Uba, M. Bamidele, J. Moulton, K. Booksh, and B. K. Lavine, "Authentication of edible oils using Fourier transform infrared spectroscopy and pattern recognition methods," Chemom. Intell. Lab. Syst., vol. 210, p. 104251, Mar. 2021, doi: 10.1016/j.chemolab.2021.104251.1.

Chapter 6

Edge Computing-Inference Algorithms on Embedded Platform

6.1. Preamble

An embedded system is a component of computer or microprocessor/microcontroller-based electronic hardware that has been programmed with application software to perform a specific function, either as a stand-alone system or as part of a larger complex system. Some of the fundamental characteristics of embedded systems include the ability to perform complex tasks, being assisted by a wide variety of processors, and having a real-time limitation, and being inexpensive. Embedded computers are used in a wide range of applications, such as digital electronics, telecommunications, computer networks, smart cards, satellite systems, military protection system equipment, robots, machine learning, and data processing systems, and many others. In recent years, machine learning algorithms, software systems, and embedded hardware have all seen significant advancements [1]. Due to the advancement in embedded technology, AI and machine learning inference algorithms can now be implemented on small-scale low-power embedded devices such as microcontrollers, allowing for the development of portable and inexpensive intelligent instrumentation.

This chapter presents Edge Implementation (EI), a machine learning technique that manages the training stage of algorithms on a computer with/without Graphical Processing Unit (GPU) and executes inferences on microcontrollers. A piece of hardware (microcontroller) with application software that can process inferences from a trained model is referred to as an Edge Device. The current work focuses on the development of an edge implementation for edible oil classification and adulteration detection inference models on an ARM embedded processor. The ARM processor was chosen for the inference implementation due to features such as sufficient memory, low development cost, efficiency, and ease of availability of programming and debugging tools. In this implementation, a pre-trained model (Linear SVM, CNN, ANN) parameters were used to develop application software (firmware) on the processor to find the class of an unknown sample and to detect the adulteration in

edible oils. The results from edge devices are compared with results of using standard software like MATLAB, Python. Such implementation will help in the development of portable and efficient intelligent systems having applications in classification and adulteration detection in edible oils. For this development, embedded C with ARM-cortex family controller was used.

This chapter begins with a description of the embedded system and the types of processors used in the embedded system. Following that, the historical and current state-of-the-art implementation of machine learning algorithms on embedded processors for different applications is discussed. Further to that, the inference implementation of developed algorithms for edible oil analysis, as well as the comparison of obtained results from this implementation, are presented.

6.2. Embedded systems

An embedded system is an integrated system of hardware and application software (also known as firmware) designed to perform a specific application. The hardware of an embedded processor consists of microprocessor unit or microcontroller unit as a central processing unit (CPU). In embedded CPU, there is a significant difference between a microprocessor and a microcontroller. A microprocessor is a general-purpose CPU that is connected to external peripherals such as a real-time clock, Flash memory, RAM, USB, Ethernet, and HDMI. An embedded Microcontroller unit, on the other hand, interfaces a few or all the peripherals to the CPU on a single system on chip (system on chip). Figure 6.1. depicts the basic building blocks of an embedded system.

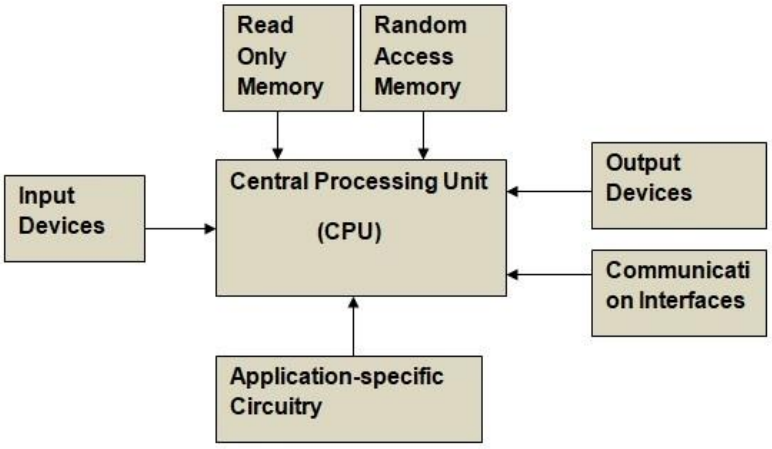


Figure 6.1 Block diagram of an Embedded System

6.3. Embedded processors

Microcontrollers, RISC processors, Digital Signal Processors (DSP), Multimedia processors, and Application Specific Instruction Set Processors (ASIP) are different types of embedded processors. A microcontroller (MCU, which stands for microcontroller unit) is a small computer that is built on a single metal-oxide-semiconductor (MOS) integrated circuit (IC) chip. A microcontroller is made up of one or more CPUs (processor cores), memory, and programmable input/output (analogue, digital) and communication (CAN, I^2C, and SPI) peripherals. A DSP processor is a specialized microprocessor with an architecture designed to satisfy the operational needs of digital signal processing. Primary goal of DSP processor is to measure, filter, and compress digital and analog signals by converting them from an acceptable analogue form to an acceptable digital form. DSPs can perform ultra-fast instruction sequences like shift and add and multiply and add. DSPs are used in products that need this form of signal processing, such as sound cards, modems, mobile phones, high-capacity hard discs, and digital televisions [2].

Harvard architecture and von-Neumann architecture are the two basic design architectures used by all processors. They represent two distinct data exchange methods between the CPU and memory. RISC (Reduced instruction set computers) processors are Harvard architecture processors. CISC (Complex instruction set computers) processors are processors that are built on von Neumann's architecture. The CPU in Harvard architecture is connected to both the data memory (RAM) and the program memory (ROM) separately. There is no separate data memory and program memory in Von-Neumann architecture. Instead, the CPU is given a single memory interface.

Intel developed the CISC instruction set. It has a diverse set of instructions ranging from simple to extremely complex. These instructions are specified at the assembly language level and take longer time to execute. CISC reduces the number of instructions on each program while ignoring the number of cycles. A single instruction in the CISC method can perform a variety of low-level operations such as memory load, arithmetic operation, and memory store. AMD, VAX, System/360, and Intel x86 are some of the best examples of CISC processors.

On the other hand, RISC processor, and its architecture includes a highly customized set of instructions. The main purpose of this is to shorten the time it takes to execute instructions by limiting and optimizing the number of instructions. As a result, each command cycle uses a single clock cycle, with each clock cycle containing three parameters: fetch, decode, and execute. PowerPC, SPARC, RISC-V, and Microchip PIC processors and ARM processors are some of the best examples of RISC processors.

Among the RISC processors ARM has several advantages like adequate size of memory, power consumption and execution time over other processors. The architecture of processor, advantages and disadvantages of ARM processor are discussed in the following section.

6.4. ARM processor

The ARM stands for Advance RISC Machine processors. It is one of the most comprehensive and licensed processor cores in the world. Cambridge University developed the first ARM processor in 1978, and the Acorn Group of Computers released the first ARM RISC processor in 1985. Because of advantages like low power consumption, good performance, and so on, these processors are specifically used in portable devices such as digital cameras, cell phones, networking devises, wireless connectivity technologies, and other embedded systems.

A. Advantages of ARM processors:

Advantages of RISC Architecture of ARM processor are reduced instruction formats, fewer instruction numbers, and fewer addressing modes.

i. Energy efficiency

They are ideal for battery-powered devices due to their high performance per watt. They also required a less sophisticated heat management system because they produce less heat than x86 processors.

ii. Performance capability

Another significant advantage of ARM processor is their performance. Despite their low power consumption, ARM processors have impressive processing capabilities. Consider the latest A series processors and Apple's new M processor as examples.

iii. Affordable

The ARM Processor can be produced at a low cost because it does not require expensive equipment when compared to other processors. As a result, they are well-suited to the production of low-cost mobile phones and other electronic devices.

B. Disadvantages of ARM processor

i. Software Incompatibility

Compatibility issues continue to be a barrier to the full adoption of ARM-based processors for use in personal computers. Programs or apps written for the x86 architecture will not run natively on ARM-powered devices.

ii. Programmer capability

The performance of ARM processors is determined by the quality of software or apps written by developers. When code is fed from poorly written programs, RISC-based CPUs tend to perform poorly.

Based on the development sequence and availability of memory and clock resources, ARM processors are classified into several series, including the ARM 1, ARM 7, ARM 10, and Cortex series. Among them ARM cortex series processors are primarily intended for use in the microcontroller domain, where the need for fast, highly deterministic interrupt management is combined with the desire for extremely low gate-count and low power consumption. In our work for the inference implementation ARM cortex M3 series processor was taken and the details of processor and its peripherals is presented in the next section.

6.5. History of embedded system and edge inference

The trend of shifting computation to edge devices is becoming significant, in the field of data analysis [3]. This computing change from the high-end computing

devices and cloud platforms to the edge have advantages in terms of response latency, bandwidth occupancy, energy consumption, security, and expected privacy [4], [5]. The development toward edge computing also extends to machine learning (ML) techniques, especially for the inference task, which is far less computationally intensive than the previous training process. In the training process, ML systems "learn" to execute tasks by considering instances input/output data, without being configured with task-specific guidelines. Edge devices can process the acquired data using the pre-trained model parameters, resulting in faster results.

Google recently launched the TensorFlow Lite framework, which includes a series of tools that allow users to transform TensorFlow Neural Network (NN) models into a simplified and reduced version, which can then be run on edge devices [6], [7]. Edge ML is a Microsoft suite of machine learning algorithms programmed to operate offline in highly resource-constrained situations [8]. ARM has released an open-source library for Cortex-M processors called Cortex Microcontroller Software Interface Standard Neural Network (CMSIS-NN), which improves NN performance [9]. The implementation of ML on embedded systems, especially with a focus on the technique of bringing computation to the edge, has only recently begun. In the literature the implementation of inference algorithms on embedded processor has been discussed. For the implementation of inference algorithms, a high-end microcontroller, a Raspberry Pi computer, and popular libraries were employed [10],[11]. Most of the ported algorithms are used in target detection, face recognition and IoT applications.

The current study focuses on the implementation of inference algorithms on low cost, general purpose embedded ARM processors that have applications in edible oil analysis without the use of any machine learning libraries. Initially, the inference algorithms were developed in the generic C/C++ programming language. For the implementation of inference algorithms, an Atmel SAM3X8E ARM Cortex-M3 microcontroller has been used. The developed inference algorithms can be used with any microcontroller with minor modifications. The Atmel SMART SAM3X/A series is a flash microcontroller family based on the high performance 32-bit ARM Cortex-M3 RISC processor. It has a maximum speed of 84 MHz and can hold up to 512 Kbytes of flash and 100 Kbytes of SRAM. The other peripherals of processor includes a high speed USB Host and Device port with embedded transceiver, an Ethernet MAC, two Control Area Networks (CAN), and a High Speed MCI for SDIO/SD/MMC, an External Bus Interface with NAND Flash Controller (NFC), 5

Universal Asynchronous Receiver Transmitters (UART), 2 Two Wire Interfaces (TWI), 4 Serial Peripheral Interfaces(SPI), as well as a Pulse width Modulator (PWM), timers, three 3-channel general-purpose 32-bit timers, a low-power Real Time Clock (RTC), a 12-bit Analog to Digital Converter (ADC) and a 12-bit Digital to Analog Converter (DAC). The SAM3X/A architecture is uniquely developed to support high-speed data transfers. It has a multi-layer bus matrix as well as multiple SRAM banks, PDC and DMA channels that allow it to run tasks in parallel and optimize data throughput, and is well suited for networking applications such as industrial and home/building automation, gateways etc. The following section describe the implementation of inference algorithms.

6.6. Qualitative analysis -classification inference models

6.6.1. Linear Support vector machine inference model implementation

As described in Chapter 4, Linear SVM is a supervised learning algorithm identifies the boundaries by creating a hyperplane for separating one sample class from another class. The class of unknown test samples is defined based on their position from these hyperplanes. The hyperplane equation is of the form $\mathbf{a} \cdot \mathbf{x} + \mathbf{b} = 0$, where $\boldsymbol{a}$ is a vector of coefficients normal to the hyperplane and $\mathbf{b}$ is a column vector of its intercept.

$$Class_{i=1}^{q} = a_{i1}f_1 + a_{i2}f_2 \ldots . + a_{ik}f_k + b_i \tag{6.1}$$

where a_{ik} are the coefficients of the hyperplane, f_k are the extracted feature and b_i are the intercepts.

For the implementation of inference algorithms, the loading matrix of training data, the coefficients, and intercepts of developed linear SVM model have been used. These coefficients were stored in a text or csv file. For an unknown input data (either original variable or extracted feature) the above equation 6.1 is computed. If for a particular class q, the results are greater than zero (result> 0), then that sample belongs to that q class, and for all other classes, the results will be less than zero. This principle was used to infer the results of the linear classifier on the embedded platform. The firmware development flow in an embedded processor is shown below in pseudo-code format.

Algorithm 6.1: The pseudo logic of firmware development

Step 1: Initialize the ARM processor.

$Initialize()$;

Number of classes= N;

Number_Coefficients=P;

Number_Variables=M

Step 2: Read the coefficients of developed SVM(coefficients and intercept) and PCA(weights) models

$$[coefficients_{NxP}] = Read\_csv\_weightfile(coefficients.csv)$$

$$[\,intercept_{Nx1}] = Read\_csv\_weightfile(intercept.csv)$$

$$[weights_{MxP}] = Read\_csv\_weightfile(weights.csv)$$

Step 3: Read an input e-tongue data

$$input\_data = Read\_sample\ data(X_{1xM})$$

Step4: Calculate the features from input data

$$Features_{1xP} = features\_extract(X, weights)$$

Step5: **for** i=0 to number classes(m)

$Class[m] = [coefficients_{NxP}] * [Transpose(Features_{1xP})] + [intercept_{Nx1}]$

end

Step6: **for** i=0 to m

if (Class[m]>0**)**

Assigned class=m;

end if

end

Step 7: Repeat from step 3 for new sample.

Case 1: Electronic tongue data with inference models on the processor

As discussed in the previous chapters regarding the electronic tongue, the training data for edible oil analysis with three working electrodes was of the size 40x24000 with eight classes of edible oils. These eight classes of edible oils are Canola (CAN) , Mustard(MUS),Olive(OL),Safflower(SAFF),Palm(PALM),Soya(SOYA),Sesame (SSM) and Groundnut (GN) Oils The developed SVM model with this data as input will produce eight hyperplane equations of 24000 dimensions. But solving for eight hyperplane equations with 24000 variables is a computationally complex task on an embedded platform. So, feature extraction method PCA was used to find the best possible principal components from this data. Using PCA, the first four principal components ($f_1, f_2, f_3 f_4$) retained, and the training data for SVM became 40x4. A linear SVM model with four principal components was developed. The model

parameters like coefficients (8x4) and intercept (8x1) were saved in a readable format. The following equations describe the hyperplane equations for eight classes.

$$
\begin{aligned}
Class1 &= \ 0.23970844 * f_1 - 0.09178667 * f_2 - 0.10610768 * f_3 + 0.40482797 * f_4 - 1.72912363 \\
Class2 &= -0.10821971 * f_1 + 0.04776101 * f_2 + 0.17060967 * f_3 - 0.10838028 * f_4 - 0.97070584 \\
Class3 &= \ 0.31691518 * f_1 - 0.30479953 * f_2 - 0.21303589 * f_3 + 0.50252771 * f_4 - 2.10594616 \\
Class4 &= -0.03102418 * f_1 - 0.09050121 * f_2 - 0.17146761 * f_3 - 0.74549052 * f_4 - 1.94948233 \\
Class5 &= \ 0.10136706 * f_1 + 0.03385491 * f_2 - 0.36656168 * f_3 - 0.09558325 * f_4 - 1.46670073 \\
Class6 &= \ 0.05640343 * f_1 - 0.4341491 * f_2 + 0.26851092 * f_3 - 0.56292715 * f_4 - 2.22412899 \\
Class7 &= \ 0.10948559 * f_1 + 0.04193459 * f_2 + 0.06694059 * f_3 - 0.29455079 * f_4 - 0.91300369 \\
Class8 &= \ 0.0476876 * f_1 + 0.20885354 * f_2 - 0.02118063 * f_3 + 0.47125695 * f_4 - 1.5192361
\end{aligned}
$$

Electronic tongue data is fed to the processor from a datafile sequentially, and the step-by-step computation results are compared with the results from standard software. During the processor initialization stage, SVM model parameters (coefficients and intercepts) were read from the model file and saved in a variable. The results of hyperplane equations with the calculated features and parameters read from the saved model file are given in Table 6.1. Table 6.1 and 6.2 show an instance of evaluating the algorithm with a test data. Computed feature values and hyperplane values matched with the standard Python results up to 3 or 4 decimal places.

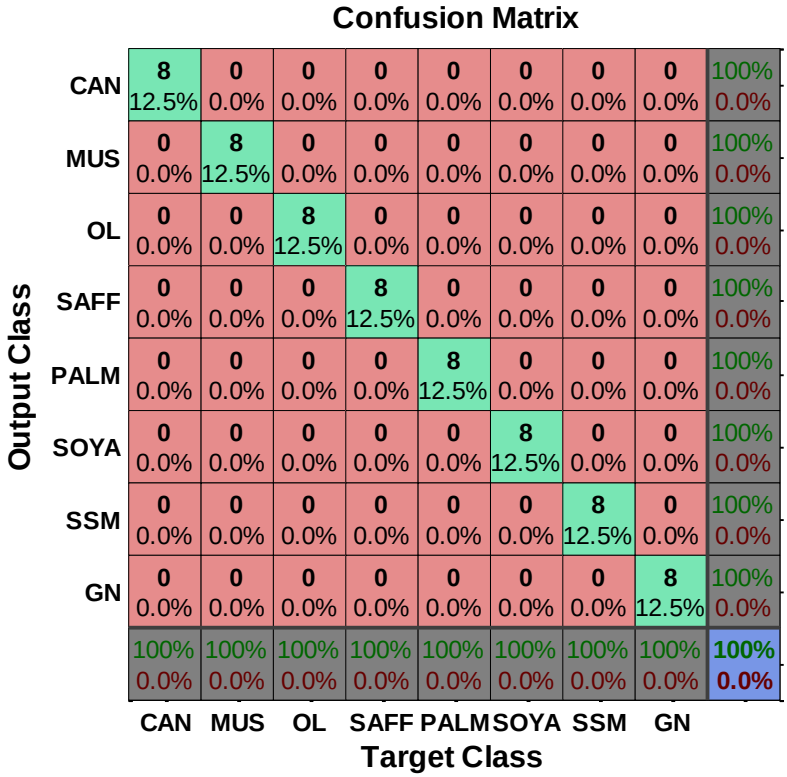
Table 6.1 Extracted feature values from software and embedded platform

Features	Calculated in Python	Calculated on Embedded
$\boldsymbol{f_1}$	4.25933	4.25929
$\boldsymbol{f_2}$	-4.70943	-4.70939
$\boldsymbol{f_3}$	2.71762	2.71751
$\boldsymbol{f_4}$	1.79898	1.79898

Table 6.2 An instance of calculated hyperplane values for a test sample from software and embedded platform

Hyperplane For class	Calculated in Python	Calculated in Embedded
Class1	0.16415	0.16414
Class2	-1.3879	-1.3879
Class3	-1.6952	-1.6952
Class4	-3.4625	-3.46256
Class5	-2.36285	-2.3628
Class6	-0.22232	-0.22238
Class7	-0.99217	-0.99217
Class8	-1.50944	-1.50943

Table 6.3 Confusion matrix of classification of edible oils using inference algorithm on an embedded platform



In the confusion matric the 12.5% corresponds to the detection accuracy (in percentage) of each class when considering all samples (Samples/ Total number of samples). The findings show that there is no variance between results calculated using an inference algorithm on embedded hardware and results calculated using standard software. The inference algorithm was applied to all test sample data sets, and the classification results observed to be 100 percent correct and comparable to the results from standard software results. Table 6.3 depicts the classification findings in the form of a confusion matrix.

Case 2: Mid infrared-ATR sampling data with linear SVM inference model for classification of edible oils

Linear SVM inference model developed for discrimination(classification) of nine types of edible oils using mid infrared ATR Sampling data. These Nine types of edible oils include Canola (CN), Groundnut (GN), Mustard (MS), Olive (OL), Palm (PALM), Safflower (SAFF), Sesame (SSM), Soya (SOY), and Sunflower (SUN) oils. The training data was of the size X_105x128 (105 samples of 9 classes and 128

variables). Principal component data reduction techniques used to extract the first four components f_1, f_2, f_3, f_4)The SVM trained model parameters were saved in a readable format, and the inference algorithm firmware was developed as explained in the previous section. The inference results were compared with actual results from standard software (Python). The following equations describe the four-dimensional hyperplane for the classification of edible oils.

$$Class1 = 0.086360 * f_1 - 0.17936 * f_2 - 0.31581 * f_3 + 0.1218 * f_4 - 2.08576$$
$$Class2 = -0.00024 * f_1 + 0.11455 * f_2 - 0.26192 * f_3 + 0.13477 * f_4 - 1.55583$$
$$Class3 = 0.033350 * f_1 + 0.11054 * f_2 - 0.05267 * f_3 - 0.04213 * f_4 - 1.05122$$
$$Class4 = 0.09757 * f_1 - 0.07358 * f_2 + 0.23006 * f_3 - 0.48512 * f_4 - 1.86340$$
$$Class5 = 0.04852 * f_1 - 0.05348 * f_2 + 0.02111 * f_3 - 0.47279 * f_4 - 1.51861$$
$$Class6 = -0.02245 * f_1 + 0.03706 * f_2 - 0.2333 * f_3 - 018366.* f_4 - 1.53427$$
$$Class7 = 0.10399 * f_1 - 0.11759 * f_2 + 0.17644 * f_3 - 0.33267 * f_4 - 1.88539$$
$$Class8 = -0.021610 * f_1 - 0.07088 * f_2 + 0.33572 * f_3 - 0.56192 * f_4 - 2.32127$$
$$Class9 = -0.093980 * f_1 - 0.19003 * f_2 + 0.12654 * f_3 - 0.23526 * f_4 - 1.75039$$

The features are calculated using the principal component method. Table 5.4 shows the calculated features from python and ARM processor using firmware.

Table 6.4 Extracted feature values from software and embedded platform

Features	Calculated in Python	Calculated on embedded processor
f_1	10.0917	10.0911
f_2	-3.71019	-3.7098
f_3	-4.63673	-4.6362
f_4	0.46640	0.4659

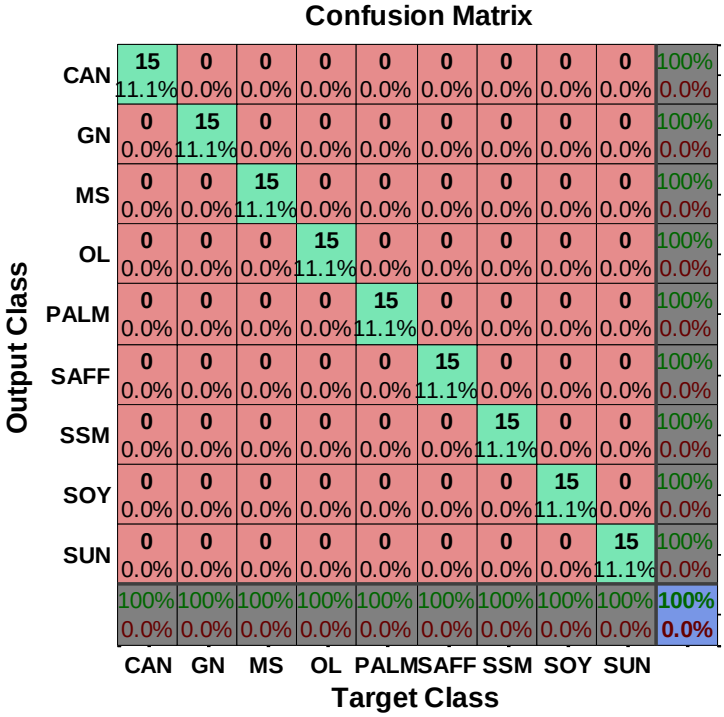
The class 1(CAN) sample data was tested on the processor for its classification. The sample was correctly classified. The Hyperplane equation for class 1(CAN) using features and saved pre-trained model coefficients are shown in Table 6.5.

The hyperplane for class 1 value is greater than zero (0), indicating the test sample belongs to class1. All test data (15 samples each class) are fed to the inference firmware on the processor and the classification results are shown in a confusion matrix Table 6.6. All samples are correctly classified with 100% classification accuracy.

Table 6.5 An instance of calculated hyperplane equations for a test sample with Python and embedded processor

Hyperplane For class	Calculated in Python	Calculated in Embedded
Class1	0.96788	0.96753
Class2	-3.1350	-3.13490
Class3	-0.90023	-0.90021
Class4	-1.44626	-1.4464
Class5	-1.14626	-1.14871
Class6	-0.90189	-0.90190
Class7	-3.4718	-3.47152
Class8	-0.98360	-0.98402
Class9	-2.47079	2.47085

Table 6.6 Classification results using inference algorithm on embedded processor



Case 3: Mid infrared spectroscopy- ATR sampling data with Linear SVM inference model for detection of adulteration

Linear SVM inference model developed for detection adulteration in Groundnut oil with cottonseed oil using mid infrared spectroscopy- ATR sampling data. The training data was of the size X_{70x128} (70 samples of 7 classes (2 Pure and 5 adulterated) and 128 variables). Principal component data reduction techniques used to extract the first four components (f_1, f_2, f_3, f_4) The inference results were compared with actual results from Python. The following equations describe the four-dimensional hyperplane for

the classification of edible oils. The features are calculated using the principal component method. Table 6.7 shows the calculated features from python and processor using firmware.

Table 6.7 Extracted feature values from software and embedded platform

Features	Calculated in Python	Calculated on embedded processor
f_1	10.0917	10.0911
f_2	-3.71019	-3.7098
f_3	-4.63673	-4.6362
f_4	0.46640	0.4659

The class 1(Groundnut Oil) sample data was tested on the processor for its classification. The sample was correctly classified. The Hyperplane equation for class 1(Groundnut) using features and saved pre-trained model coefficients are shown in Table 6.8. The hyperplane for class 1 value is greater than zero (0), indicating the test sample belongs to class1. All test data (10 samples each class) fed to the inference firmware on the processor and the classifications are shown in a confusion matrix Table 6.9. All samples are correctly classified with 100% classification accuracy.

Table 6.8 An instance of calculated hyperplane equations for a test sample with Python and embedded processor

Hyperplane For class	Calculated in Python	Calculated in Embedded
Class1	0.96788	0.96753
Class2	-3.1350	-3.13490
Class3	-0.90023	-0.90021
Class4	-1.44626	-1.4464
Class5	-1.14626	-1.14871
Class6	-0.90189	-0.90190
Class7	-3.4718	-3.47152

Table 6.9 Classification results using inference algorithm on embedded processor

% Adulteration Predicted Class	**Target Samples class**							**Accuracy (%)**
	GNUT	**5%**	**10%**	**25%**	**50%**	**75%**	COT	
GNUT	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
5%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
10%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
25%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	100% 0.0%
50%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	100% 0.0%
75%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	100% 0.0%
COT	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
	100 0%	100 0%	100 0%	100 0%	100 0%	100% 0%	100% 0%	100% 0%

6.6.2. 1-D Convolution neural network inference model on embedded processor

1-D CNN model is developed for the analysis of edible oils using Mid infrared spectroscopy with ATR sampling method. The Developed (trained) CNN algorithm for classification of nine varieties of edible oils produces a set of optimized weights and biases for each layer. The input for the CNN model is the spectrum data(1x128) and the output is the class of the edible oil. The first layer weights are of the size 1x3x10, and the bias is of size 1x10. Second layer weights are of the size(1x2x10) and bias of 1x10. Dense layer weights are of the shape 1x1080 and bias of 1x10. The last dense layer weights are of shape 10x1, and bias is of size 9x1. These layers weights and bias were stored in a readable format in the processor during the initialization. The pseudo-code for the CNN algorithm is given in algorithm 2.

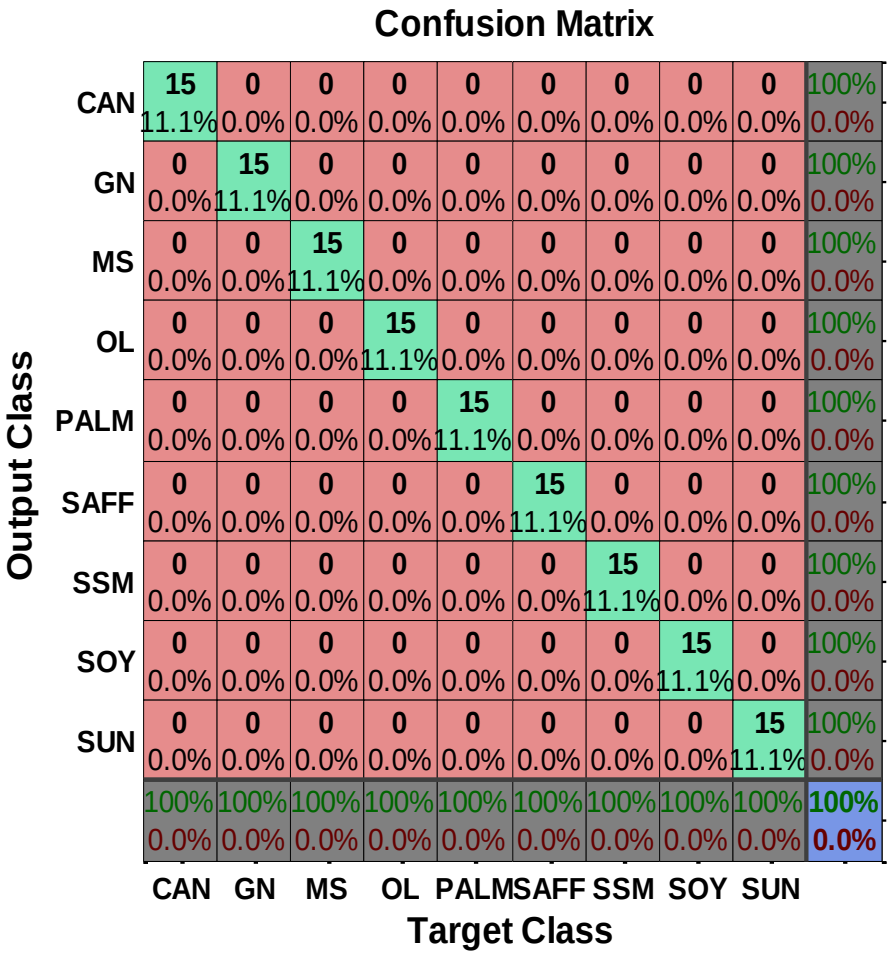
Algorithm 6.2: The pseudo logic of 1D-CNN inference algorithm development

Step 1: Initialize the ARM processor.
$Initialize()$;
Input_Vaiable size=N;
Stride=1;
Padding=0;
Number_outputclass=q
Step 2: Read the weights and bias of the developed CNN model
$[Weights\_conv1, bias1] = Read\_csv\_(weights\_bias1.csv)$
$[Weights\_conv2, bias2] = Read\_csv\_(weights\_bias2.csv)$
$[Weights\_dense1, bias3] = Read\_csv\_(weights\_bias3.csv)$
$[Weights\_dense2, bias4] = Read\_csv\_(weights\_bias4.csv)$
Step 3: Read an input ATR data
$input\_data = Read\_sample\ data(X_{1xN})$
Step4: Minmax Normalize;
Step5: **for** i=0 to 10
$Conv1\_output_{119xi} = conv\_layer(Features_{Nx1}, weights\_conv_{1x3xi})$
end
Step6: for i=0 to 10
$Max\_output1_{118xi} = Maxpool\_conv(Conv1\_output_{119xi}, Size_{1x2})$
end
Step7: **for** i=0 to 10
$Conv2\_output_{109xi} = conv\_layer(Max\_output1_{118xi}, weights\_conv_{1x2xi})$
end
Step8: for i=0 to 10
$Max\_output2_{108xi} = Maxpool\_conv(Conv2\_output_{109xi}, Size_{1x2xi})$
end
Step 9: $Flatten_{1x1080}$= Flatten_layer($Max\_output2_{108x10}$)
Step 10: $Dense1\_output_{10}$=Dense1_layer(Flatten $Weights\_dense1$,bias3)
Step 11: $Dense2\ \_output_{q}$ =Dense2_layer($Dense1_{output_{10x1}}$, $Weights\_dense2$,bias4)
Step 12: Provide Softmax (Dense2_output) as result.

Case 1: Classification of edible oils

A test data (PALM oil data) for edible oil classification was fed to the 1-D CNN inference algorithm. The output from each layer calculated from the embedded processor was verified with standard python outputs. A total of 105 samples (nine classes, 15 samples in each class) were randomly fed to the inference algorithm, and the classification results were tested using the ARM processor. Table 6.10 shows the classification accuracy in a confusion matrix format.

Table 6.10 Classification results using inference model on an embedded processor



Case 2: Classification of adulterated edible oils

A test data of mid infrared spectroscopy with ATR sampling for adulteration detection (sunflower oil adulterated with palm oil) was fed the 1-D CNN inference algorithm. The output from each layer calculated from the embedded processor was verified with standard python outputs. A total of 70 samples (7 classes, 10 samples in each class) were randomly fed to the inference algorithm, and the classification results were tested using the ARM processor. Table 6.11 shows the classification accuracy in a confusion matrix format.

Table 6.11 1D CNN results using inference algorithm on an embedded processor

% Adulteration Predicted Class	**Target Samples class**							**Accuracy (%)**
	SUN	**5%**	**10%**	**25%**	**50%**	**75%**	PALM	
SUN	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
5%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
10%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
25%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	0 0%	100% 0.0%
50%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	0 0%	100% 0.0%
75%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	0 0%	100% 0.0%
PALM	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	10 14.3%	100% 0.0%
	100 0%	100 0%	100 0%	100 0%	100 0%	100% 0%	100% 0%	100% 0%

6.6.3. 2-D Convolution neural network inference model on embedded processor

A novel 2D CNN model for edible oil analysis has been developed. The input for the 2D CNN model is the Correlation matrix(128x128) calculated from the spectrum with ATR sampling method(1x128) the output of the 2D CNN is the class of the input edible oil sample. The Developed (trained) 2D-CNN algorithm produces a set of optimized weights and biases for each layer. The first layer weights are of the size 3x3x10, and the bias is of size 1x10. Second layer weights are of the size(2x2x10) and bias of 1x10. Dense layer weights are of the shape 1x151290 and bias of 1x10. The last dense layer weights are of shape 10x1, and bias is of size 9x1. These layers weights and bias were stored in a readable format in the processor during the initialization. The pseudo-code for the CNN algorithm is given in algorithm 3.

Case 1: Classification of edible oils

Mid infrared spectroscopy with ATR sampling test data for edible oil classification was fed the 2D-CNN inference algorithm. The output of each layer calculated from the embedded processor was verified with standard python outputs. A total of 45 samples (nine classes, 5 samples in each class) were randomly fed to the inference algorithm, and the classification results were tested using the ARM processor. Table 6.12 shows the classification accuracy in a confusion matrix format.

Algorithm 6.3: The pseudo logic of 2D-CNN Inference model development

Step 1: Initialize the ARM processor.
$Initialize()$;
Input_Vaiable size=N;

Step 2: Read the weights and bias of the developed CNN model
$[Weights\_conv1, bias1] = Read\_csv\_(weights\_bias1.csv)$
$[Weights\_conv2, bias2] = Read\_csv\_(weights\_bias2.csv)$
$[Weights\_dense1, bias3] = Read\_csv\_(weights\_bias3.csv)$
$[Weights\_dense2, bias4] = Read\_csv\_(weights\_bias4.csv)$
Step 3: Read an input ATR data
$input\_data = Read\_sample\ data(X_{1x128})$

Step4: Calculate the correlation matrix
$Features_{128x128} = X^{T}_{128x1} * X_{1x128}$
Step5: **for** i=0 to 10
$Conv1\_output_{126x126xi} = conv\_layer(Features_{128x128x1}, weights\_conv_{3x3xi})$
end
Step6: for i=0 to 10
$Max\_output1_{125x125xi} = Maxpool\_conv(Conv1\_output_{126x126xi}, Size_{2x2})$
end
Step7: **for** i=0 to 10
$Conv2\_output_{124x124xi} = conv\_layer(Max\_output1_{125x125xi}, weights\_conv_{2x2xi})$
end
Step8: for i=0 to 10
$Max\_output2_{123x123xi} = Maxpool\_conv(Conv2\_output_{124x124xi}, Size_{2x2xi})$
end
Step 9: $Flatten_{1x151290}$= Flatten_layer($Max\_output2_{123x123x10}$)
Step 10: $Dense1\_output_{1x10}$=Dense1_layer(Flatten)
Step 11: $Dense2\ \_output_{1x9}$ =Dense2_layer($Dense1\_output_{1x10}$)
Step 12: Provide Softmax (Dense2_output) as result.

Case2: Classification of adulterated edible oils

Mid infrared spectroscopy with ATR sampling data for sunflower oil adulterated with palm oil in proportions of 5%(v/v), 10%(v/v), 15%(v/v), 25%(v/v), 75%(v/v) was used to develop a CNN classification algorithm. The developed model trained with 70 samples (7 classes, 10 samples in each class). The correlation spectra of each sample are calculated and used for training the model. Trained model layers weights and bias were stored in a readable CSV format. Testing the inference model with 35 samples of adulterated edible oil samples (7 classes, 5 samples each) randomly fed to the inference algorithm gave 100% classification accuracy. All adulterated samples were clearly classified as adulterated and pure samples were classified as pure.

Table 6.12 Confusion matrix of Classification results using 2D CNN inference model on an embedded processor

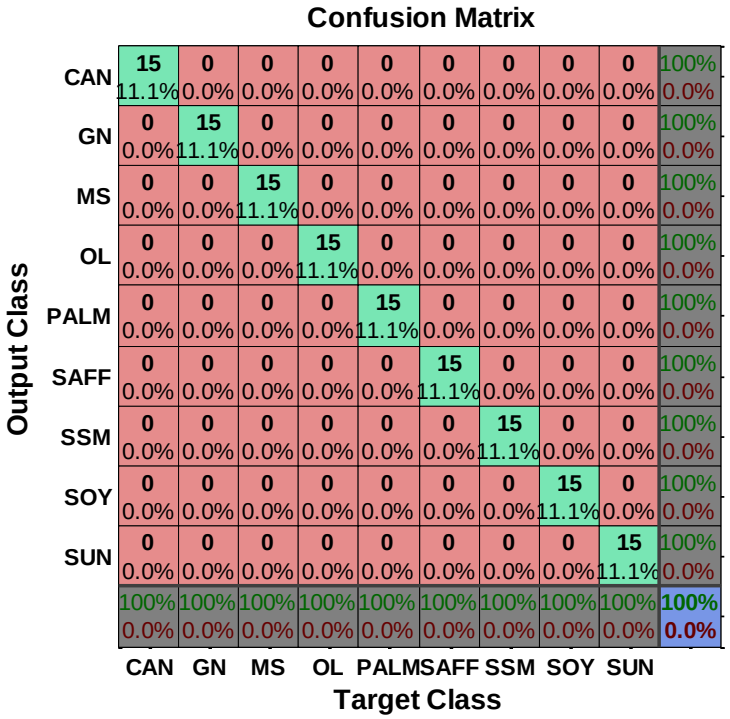


Table 6.13 Classification results of 2D CNN inference model on an embedded processor

% Adulteration Predicted Class	Target Samples class							Accuracy (%)
	SUN	5%	10%	25%	50%	75%	PALM	
SUN	5 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
5%	0 0%	5 14.3%	0 0%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
10%	0 0%	0 0%	5 14.3%	0 0%	0 0%	0 0%	0 0%	100% 0.0%
25%	0 0%	0 0%	0 0%	5 14.3%	0 0%	0 0%	0 0%	100% 0.0%
50%	0 0%	0 0%	0 0%	0 0%	5 14.3%	0 0%	0 0%	100% 0.0%
75%	0 0%	0 0%	0 0%	0 0%	0 0%	5 14.3%	0 0%	100% 0.0%
PALM	0 0%	0 0%	0 0%	0 0%	0 0%	0 0%	5 14.3%	100% 0.0%
	100 0%	100 0%	100 0%	100 0%	100 0%	100% 0%	100% 0%	100% 0%

6.7. Regression inference model

6.7.1. PLS regression inference model on embedded processor

As discussed in Chapter 5, Partial Least Square Regression (PLSR) model was developed for the quantification of percentage of adulteration in edible oils using mid infrared spectroscopy with ATR sampling data. From the trained model, selected variables indexes and weight matrix for regression model have been saved in a readable csv file format. These variables and weights were used in prediction model development on ARM-based microcontroller. The pseudocode for calculation of approximate percentage of adulteration in edible oil is given below.

Algorithm 6.4: The pseudo logic of firmware development for the regression model

Step 1: Initialize the ARM processor.

$Initialize();$

Step 2: Read the weights and bias of developed PLSR

$$[Weights, Variables] = Read\_csv\_(weights\_variable.csv)$$

Step 3: Read an input ATR spectroscopy data

$$input\_data = Read\_sample\ data(X_{1x128})$$

Step4: Smooth data with Savgol filter

$$Smooth\_data_{1x128} = Savgol(X_{1x128})$$

Step5: Select the variables from the index saved in model

$$Select\_data_{1xn} = Smooth(X[variable\ indexes])$$

Step6: Calculate Y as

$$Y\ calc = [1\ Select\_data] * weights;$$

Step 7: end.

The prediction of percentage of adulteration in groundnut oil with cotton seed oil has been used for the testing of inference model results. The regression results are compared with the results from MATLAB. The following Table 6.14 shows the comparative results of predicted values from MATLAB and regression model on an embedded platform.

Table 6.14 Regression results calculated from MATLAB and inference algorithm on the processor

The actual percentage of adulteration	**MATLAB regression result (%)**	**Embedded result (%)**
5%	5.2230	5.22
10%	10.50	10.52
25%	28.57	28.54
50%	52.85	52.84
70%	71.85	71.83

6.7.2. ANN based regression model on embedded Processor

Artificial Neural Network (ANN) based regression model was developed for the quantification of edible oil adulteration (groundnut oil with cottonseed oil). The input for the inference model is the spectrum data and output are the percentage of adulteration in it. The weights and bias matrix from the developed ANNR model were used in the inference algorithm development on processor. The following pseudo logic explains the implementation procedure on the processor logic.

Algorithm 6.5: The pseudo logic ANN regression on the Embedded Processor

Step 1: Initialize the ARM processor. $Initialize()$;

Step 2: Read the Noamalization and denormalization parameters from csv

$$[\mathrm{xoffset}_{128x1}, \mathrm{xgain}_{128x1}, \mathrm{xmin}_{1x1}\ ] = Read\_csv\_(Normalization.csv)$$

$$[\mathrm{yoffset}_{1x1}, \mathrm{ygain}_{1x1}, \mathrm{ymin}_{1x1}] = Read\_csv\_(DenormNormalization.csv)$$

Step 3: Read the weights and bias of developed ANNR regressin model

$$[\mathrm{Weights1}_{10x128}, \mathrm{Bias1}_{10x1}, \mathrm{Weights2}_{10x1}, \mathrm{Bias2}_{1x1}] = Read\_csv\_(weights\_variable.csv)$$

Step 4: Read an input ATR spectroscopy data

$$\mathrm{input\_data} = \mathrm{Read\_sample\ data}(\mathrm{X}_{128x1})$$

Step 5: Normalize the input data

$$\mathrm{Normdata}_{128x1} = \left[\left(\mathrm{X}_{128x1} - \mathrm{xoffset}_{128x1}\right) * \mathrm{xgain}_{128x1}\right] + \mathrm{ones}_{128x1} * (\mathrm{xmin}_{1x1})$$

Step 6: Calculate the regression equation output

$$\mathrm{Layer1}_{10x1} = \mathrm{Sigmoid}[(\mathrm{Weights}_{10x128} * \mathrm{Normdata}_{128x1}) + \mathrm{Bias1}_{10x1}]$$

$$\mathrm{Layer2}_{1x1} = (\mathrm{Weights}_{1x10} * \mathrm{Layer1}_{10x1}) + \mathrm{Bias2}_{1x1}$$

Step 7: Calculate Y as Denormalized Layer2

$$\mathrm{Y\ calc} = \left(\frac{\mathrm{Layer2}_{1x1} - \mathrm{ymin}_{1x1}}{\mathrm{ygain}_{1x1}}\right) + \mathrm{yoffset}_{1x1};$$

Step 8: end.

The regression results are compared with the results from standard software. The following Table 6.15 shows the results of predicted adulteration levels from MATLAB and inference regression model on an embedded processor.

Table 6.15 Regression results calculated from MATLAB and ANNR inference algorithm on the processor

The actual percentage of adulteration	**MATLAB regression result (%)**	**Embedded result (%)**
5%	5.0057	5.01
10%	9.9726	9.97
25%	25.0343	25.03
50%	51.0829	51.07
70%	75.0044	75.00

6.8. Summary

Implementation of AI inference algorithms on embedded platforms will lead to the development of intelligent instruments for a specific application. In this chapter, porting inference algorithms on embedded systems for classification and adulteration detection were presented. Linear SVM model and 1D-CNN and 2D CNN inference model for classification of edible oils using electronic tongue data and ATR spectroscopy data are presented. The PLSR and ANN based regression inference models were implemented on the embedded processor for the quantification of adulteration in edible oils. Results for classification and regression from the embedded platform were compared to results from standard software packages. Results from embedded software were close to the results obtained from the standard software (MATLAB, Python). This implementation may be used for the development of portable edible oil classification and adulteration detection systems in the future.

Bibliography

[1] F. Sakr, F. Bellotti, R. Berta, and A. De Gloria, "Machine Learning on Mainstream Microcontrollers," *Sensors*, vol. 20, no. 9, p. 2638, May 2020, doi: 10.3390/s20092638.

[2] R. Oshana, "Overview of Embedded Systems Development Life Cycle Using DSP," in *DSP Software Development Techniques for Embedded and Real-Time Systems*, Elsevier, 2006, pp. 35–58.

[3] Q. Yang, X. Luo, P. Li, T. Miyazaki, and X. Wang, "Computation Offloading for Fast CNN Inference in Edge Computing," in *Proceedings of the 2019 Research in Adaptive and Convergent Systems, RACS 2019*, Sep. 2019, pp. 101–106, doi: 10.1145/3338840.3355669.

[4] M. Satyanarayanan and (Satya, "Edge Computing: Vision and Challenges," 2016. http://www.cse.wustl.edu/~jain/.

[5] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016, doi: 10.1109/JIOT.2016.2579198.

[6] "TensorFlow Lite | ML for Mobile and Edge Devices." https://www.tensorflow.org/lite .

[7] M. S. Louis *et al.*, "Towards Deep Learning using TensorFlow Lite on RISC-V," 2019, doi: 10.1145/1122445.1122456.

[8] "Dennis: EdgeML: Machine Learning for resource-constrained... - Google Scholar."https://scholar.google.com/scholar?cluster=9550598496014511606&hl=en&oi=scholar.

[9] N. Suda, D. L.-A. L. . Cambridge, undefined UK, and undefined 2019, "Machine Learning on ARM Cortex-M Microcontrollers," *developer.arm.com*, Accessed: Apr. 14, 2021. [Online]. Available: https://developer.arm.com/-/media/Files/pdf/Arm ML on Cortex-M Microcontrollers.pdf.

[10] M. Yazici, S. Basurra, and M. Gaber, "Edge Machine Learning: Enabling Smart Internet of Things Applications," *Big Data Cogn. Comput.*, vol. 2, no. 3, p. 26, Sep. 2018, doi: 10.3390/bdcc2030026.

[11] G. Cerutti, R. Prasad, and E. Farella, "Convolutional Neural Network on Embedded Platform for People Presence Detection in Low Resolution Thermal Images," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, May 2019, vol. 2019-May, pp. 7610–7614, doi: 10.1109/ICASSP.2019.8682998.

Chapter 7

Conclusions and Future Recommendations

7.1. Preamble

The overall conclusions drawn from the thesis work covering the analysis of edible oils for the classification and detection of adulteration using electrochemical methods and mid-infrared spectroscopy methods coupled with artificial intelligent pattern recognition algorithms for rapid inferences are described in this chapter. This chapter also discusses the conclusions drawn regarding the implementation of developed algorithms on embedded platforms in the direction of developing handheld systems for the analysis of edible oils and adulterations in them. Furthermore, new research directions that have emerged as a result of the thesis work are presented in future recommendation section.

7.2. Conclusions

The following are the issues and challenges that motivated the work presented in this thesis.

- Need for a dedicated analytical system for detection of adulterations in edible oils as these edible oils are the vital components in human diet for providing nutritional values.
- FSSAI survey and study in several states of INDIA resulted in higher percentage of adulterated edible oils samples emphasizing the need of a dedicated analytical system for detection of adulterations in edible oils.
- Need for simple, transparent, reliable, and accurate systems.
- Need for on-line monitoring and rapid analysis, and accurate results.
- Need for embedded systems in developing intelligent algorithms for the analysis of edible oils to have miniaturized, portable, and standalone systems.

The research work was primarily concerned with various issues and problems encountered as well as appropriate solutions to address these issues. This thesis represents a step toward the development of intelligent instrumentation systems for the rapid analysis of edible oils i.e., detection of adulteration in edible oils and discrimination of different types of edible oils based on their signatures. The work

started with traditional standard chemical methods for measuring physicochemical parameters of collected edible oils such as peroxide value, refractive index, iodine value, fatty acid profile values, saponification values, and un-saponifiable matter. As a result, the authentication of collected edible oils ensuring no adulteration was done by validating with FSSAI specification values. The comparison results showed that the measured parameters were in the specified ranges approved by FSSAI.

The thesis presents instrumental methods for determining simple yet accurate systems for the analysis of edible oils. The algorithms presented in this thesis are concerned with dimensionality reduction, appropriate feature selection/extraction classification, and detection of adulterations in edible oils using data obtained from analytical instruments. In addition, inference algorithms are being implemented on embedded platforms.

The work focuses on the investigation of analytical instruments for the analysis of edible oils, such as electrochemical and spectroscopic methods for determining a simple but reliable method. For this, first the classification of different types of collected edible oils with analytical instruments such as electronic tongues based on voltammetry, electrochemical impedance sampling methods, and mid infrared spectroscopy with ATR sampling method have been investigated. Electronic tongue experiments require sample preparation for making the edible oils conductive, which utilizes hazardous chemicals like petroleum ether, and it is a time-consuming laborious task.

Whereas near infrared (NIR) spectroscopy doesn’t require any specific sample preparation for the analysis of edible oils, but due to the viscous and sticky nature of oil samples, cuvette cleaning after each sample data acquisition is a difficult task. The residual stains will affect the next sample reading. On the other hand, mid infrared spectroscopy with ATR sampling techniques is suitable for direct analysis of liquid and solid samples and the cleaning of ATR crystal after each sample acquisition is also simple. Because of its ability to perform experiments without sample preparation, mid infrared spectroscopy with ATR sampling method has been found as the best method for the analysis of edible oils.

For the study of detecting adulterations in edible oils lab-made adulterated samples in proportions of 5%(v/v),10%(v/v),25%(v/v),50%(v/v) and 75%(v/v) are used in mid infrared spectroscopy with ATR sampling. Emphasis has been put on the development of AI based pattern recognition algorithms for detecting adulteration in

edible oils. ATR-MIR spectroscopy and regression analysis methods are also used to calibrate the percentage of adulteration. Finally, the thesis focuses on the realization of developed model inference algorithms on embedded platforms and the validation of their results using standard software such as Matlab. The thesis's objective of developing simple, transparent, and yet accurate pattern recognition algorithms for detecting adulterations in edible oils, specifically with ATR sampling, has been met.

The following are the thesis work's conclusions

- The measurement of physicochemical properties of edible oil samples using traditional standard chemical methods has been presented to assess the authenticity and absence of adulteration in the collected edible oil samples. These measured values have been validated and compared to the FSSAI standard specifications for edible oil samples. The obtained values are within the range of specifications, indicating the authenticity of the collected edible oil samples and the absence of adulterations.

- Various analytical instruments methods for food analysis, such as electrochemical methods and spectroscopic methods, have been investigated for applications in edible oil discrimination. Based on this analysis, a simple, quick, and accurate analytical method for edible oil analysis is identified. The term "simple" refers to analytical instruments that do not require sample preparation and has smaller, simpler experimentation procedure. Using this method [mid infrared spectroscopy with ATR sampling technique], the data acquisition with lab made adulterated samples have been completed.

- A rapid mid infrared (MIR) spectroscopic method with ATR sampling coupled with chemometric model for discrimination of different edible oil and detection of adulteration in these edible oils has been presented. The chemometric models were developed, which utilize the power of artificial intelligence and classical classification algorithms. For classification, several statistical classification algorithms and soft computing algorithms have been developed. The effectiveness of classification algorithms has been investigated. Linear Support Vector Machine (LSVM) algorithms and Convolution Neural Network (CNN)-based classification models have been proposed and found to be suitable and accurate for edible oil classification.

- A novel method for quantification of percentage of adulteration in edible oil samples has been proposed. Partial Least Squares Regression (PLSR) models, Successive Projection Analysis (SPA) methods for variable selection and regression analysis, and soft computing regression analysis based on artificial neural networks have all been investigated. The MLR and ANNR regression results based on SPA have been found to be more accurate in predicting the percentage of adulteration.
- The realization of selected classification models for discriminating the different types of edible oils and adulterated samples on embedded platform has been accomplished. Regression model's implementation on embedded processor (ARM7 TDMI) has also been proposed. The classification and regression results from embedded processor have been compared with the results of standard software (MATLAB, Python). The results are on par with the standard results from commercial software packages indicating the feasibility of developing portable intelligent instrumentation for the analysis of edible oils.

7.3. Future Recommendations

The thesis work opened up new research avenues. Some of the intriguing aspects that can be addressed as a result of the work are as follows

- Integration of Mid infrared spectroscopy with ATR sampling method hardware and embedded processor with the inference algorithms can lead to the development of systems for analysis of edible oils, which can be further extended for the analysis other food items like paneer, Butter.
- In the thesis, a few data reduction algorithms were proposed. Improvements can be made to reduce the dimensionality and complexity of the systems involved even further.
- There is a scope to develop a supply chain management system employing IoT systems for food analysis using the inference algorithms on edge devices. This can be used to reduce the computational complexity at the cloud side and manage cloud with less space.
- Decreasing minimum detection limits (LOD). In this investigation, we considered the level of adulteration from 5% to 75% (v/v). The experimental approach can be studied with a lower percentage of adulterations (less than 5% v/v) and to develop better algorithms to reduce the minimum detection level to 1% (v/v) or below.

Appendix-A

A.1 : MATLAB COMMANDS FOR K-MEANS CLUSTERING

```
%%%%%%K means clustering MATLAB R2014a %%%%%
clear all
clc
load ('Data.mat').
Dst=Data;
[idx3,C,sumdist3] =
kmeans(Dst,7,'Distance','sqeuclidean','Display','final');
Figure(1)
plot (Dst(idx3==1,1),Dst(idx3==1,2),'r*','MarkerSize',12)
hold on
plot (Dst(idx3==2,1),Dst(idx3==2,2),'b*','MarkerSize',12)
plot (Dst(idx3==3,1),Dst(idx3==3,2),'g*','MarkerSize',12)
plot (Dst(idx3==4,1),Dst(idx3==4,2),'m*','MarkerSize',12)
plot (Dst(idx3==5,1),Dst(idx3==5,2),'k*','MarkerSize',12)
plot (Dst(idx3==6,1),Dst(idx3==6,2),'c*','MarkerSize',12)
plot (Dst(idx3==7,1),Dst(idx3==7,2),'y*','MarkerSize',12)
plot(C(:,1),C(:,2),'ko','MarkerSize',12,'LineWidth',1)

legend('C1','C2','C3','C4','C5','C6','C7','Centroids','Locatio
n','NW')
grid on
%%%%%%K means clustering MATLAB R2014a %%%%%
```

A.2: MATLAB COMMANDS FOR HIERARCHICAL CLUSTERING

```
clear all
clc
load ('Data.mat').
Dst= Data.
tree = linkage (Dst,'complete');
H=dendrogram(tree,0,'Orientation','left','ColorThreshold',0.02
5).
set(H,'LineWidth',2)
grid on
```

A.3: MATLAB COMMANDS FOR SUBTRACTIVE CLUSTERING

```
 clear all % clearing all variables from workspace
clc       % clearing all commands from command window
load('Nine Oils.mat')%% loading the data
A=load('Nine Oils.mat')
%% C refer to the Cluster centre matrix. S refers to the sigma
values.
```

```
[C,S] = subclust(A,2);%% Subtractive clustering command.
%%Initialize a figure here%%
figure(1);
hold on;
%% Scatter Plot between first and second variables%%
plot(A(1:5,1),A(1:5,2),'b*','MarkerSize',12)
plot(A(6:10,1),A(6:10,2),'r*','MarkerSize',12)
plot(A(11:15,1),A(11:15,2),'g*','MarkerSize',12)
plot(A(16:20,1),A(16:20,2),'k*','MarkerSize',12)
plot(A(21:25,1),A(21:25,2),'c*','MarkerSize',12)
plot(A(26:30,1),A(26:30,2),'m*','MarkerSize',12)
plot(A(31:35,1),A(31:35,2),'y*','MarkerSize',12)
plot(A(36:40,1),A(36:40,2),'rs','MarkerSize',12)
plot(A(41:45,1),A(41:45,2),'bs','MarkerSize',12)
plot(C(:,7),C(:,15),'ko','MarkerSze'12)
%% legend indicates the sample labels%%
legend('CAN','GN','MS','OLIVE','PALM','SAFF','SSM','SOYA','SUN
','Cluster');
grid on
```

A.4: PYTHON CODE COMMANDS FOR SVM

```
%%%%%%%% import specific libraries required%%%%%%%
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from scipy.signal import savgol_filter
from sklearn.decomposition import PCA as sk_pca
from sklearn.preprocessing import StandardScaler
from sklearn import svm
from sklearn.utils.extmath import randomized_svd
from sklearn.svm import LinearSVC
```

```
%%%%%%%% import specific libraries required%%%%%%%

%%%%%%%% Read Data from csv File %%%%%%%
```

```
data = pd.read_csv('Data.csv', header=None) % reading content
of csv

Data_label = data.values[:,-1] % the label data%
print (Data_label.shape)

features=data.values[:,0:-1] % the spectra data%
print (features.shape)
```

```
Data_smooth = savgol_filter(features, Num, polyorder = 0,
deriv=0)
%%%%%%%% Calculate SVD for loading matrix V %%%%%%%
```

```
U, Sigma, VT = randomized_svd(feat,
                              n_components=4,
                              n_iter=100,
                              random_state=None)


%%%%%%%% %%%%%%%%%%%%%%%%%%%%%%%% %%%%%%%
%%%%%%%% Calculate the Principal components  %%%%%%%
PC=feat.dot(VT.T)
Print(PC.shape)

%%%%%%%% Identify unique labels in the Data_label  %%%%%%%


unique = list(set(Data_label))
colors = [plt.cm.jet(float(i)/max(unique)) for i in unique]
label_plot = ["C1","C2","C3", "C4","C5", "C6","C7"]
%%%%%%%%    %%%%%%%%%%%%%% %%%%%%%%%% %%%%%%%%%% %%%%%%%

%%%%%%%% Score Plot for PCS %%%%%%%

with plt.style.context(('ggplot')):
    plt.figsize=(15,15)
    for i, u in enumerate(unique):
        col = np.expand_dims(np.array(colors[i]), axis=0)
        xi = [PC[j,0] for j in range(len(PC[:,0])) if lab[j]
== u]
        yi = [PC[j,1] for j in range(len(PC[:,1])) if lab[j]
== u]
    plt.scatter(xi, yi, c=col, s=60,
edgecolors='k',label=str(u))
    plt.xlabel('PC1')
    plt.ylabel('PC2')
    plt.legend(label_plot,loc='best')
    plt.title('Principal Component Analysis')
    plt.show()

%%%%%%%% Train Linear SVM Model  %%%%%%%

X=PC
Y=Data_label
clf=LinearSVC(C=1.0, class_weight=None, dual=True,
                fit_intercept=True, intercept_scaling=1,
               loss='squared_hinge', max_iter=100000,
               multi_class='ovr', penalty='l2',
               random_state=None, tol=0.0001,
          verbose=0)clf.fit(X,Y)
print("Model training is complete")

%%%%%%%% Model training Complete   %%%%%%%
%%%%%%%% Test the Model with a new data    %%%%%%%
```

```
Test=np.reshape(feature[n],(1,-1)) %% reshape test data
Test1=Test.dot(VT.T)           %% calculate PCS

clf.predict(Test1)              %% test with developed SVM
model.
```

A.5: PYTHON CODE COMMANDS FOR 1D-CNN

```
%%%%%%%% import specific libraries required%%%%%%%
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import keras
from   keras.models import Sequential
from keras.layers import
Conv2D,MaxPooling2D,Flatten,Dense,Conv1D,MaxPooling1D
from sklearn.preprocessing import MinMaxScaler
 %%%%%%%% Import of libraries completed %%%%%%%

%%%%%%%% Read CSV file for data and labels%%%%%%%
S = pd.read_csv('Classif.csv', header=None) %%% read csv file

Label=S.iloc[:,-1]                %%% read labels from last
column

Spectrum=S.iloc[:,:-1].values  %%% read spectra data %%%%

%%%%%%%% Read CSV file for data and labels%%%%%%%
A=np.empty((45,128,1))    % define a empty array with 45x128
size


%%%%%%%% Normalize spectra data with standard scalar%%%%%%%

xscaler= MinMaxScaler()
xscaler.fit(np.transpose(Spectrum))
x=xscaler.transform(np.transpose(Spectrum))

%%%%%%%% reshape the data in to 128x1%%%%%%%

for i in range(105):
    A[i]=x[i].reshape(128,1)
%%%%%%%% Divide data in to training and testing data%%%%%%%

Train_data =[]
Train_data=A[0:10,:,:]
Train_data=np.append(Train_data,A[15:25,:,:],axis=0)
Train_data=np.append(Train_data,A[30:40,:,:],axis=0)
Train_data=np.append(Train_data,A[45:55,:,:],axis=0)
Train_data=np.append(Train_data,A[60:70,:,:],axis=0)
```

```
Train_data=np.append(Train_data,A[75:85,:,:],axis=0)
Train_data=np.append(Train_data,A[90:100,:,:],axis=0)
```

```
Test_data =[]
Test_data=A[10:15,:,:]
Test_data=np.append(Train_data,A[25:30,:,:],axis=0)
Test_data=np.append(Train_data,A[40:45,:,:],axis=0)
Test_data=np.append(Train_data,A[55:60,:,:],axis=0)
Test_data=np.append(Train_data,A[70:75,:,:],axis=0)
Test_data=np.append(Train_data,A[85:90,:,:],axis=0)
Test_data=np.append(Train_data,A[100:105,:,:],axis=0)
```

```
%%%%%%%% Training and testing labelled data%%%%%%%
```

```
Train_Y=keras.utils.to_categorical(Train_Label_matrix)
Train_Y.shape
Test_Y=keras.utils.to_categorical(Test_Label_matrix)
Test_Y.shape
%%%%%%%% 1D CNN model Architecture for edible oil
analysis%%%%%%%
```

```
model = Sequential()
model.add(Conv1D(10, kernel_size=10,
strides=1,activation='relu',input_shape=(128,1)))
model.add(MaxPooling1D(pool_size=2, strides=1))
model.add(Conv1D(10, kernel_size=10,
strides=1,activation='relu'))
model.add(MaxPooling1D(pool_size=2, strides=1))
model.add(Flatten())
model.add(Dense(10, activation='relu'))
model.add(Dense(7, activation='softmax'))
```

```
%%%%%%%% Compile and save 1D CNN model %%%%%%%
```

```
callbacks = [
    keras.callbacks.ModelCheckpoint(
        "best_model.h5", save_best_only=True,
monitor="val_loss"
    ),
    keras.callbacks.ReduceLROnPlateau(
        monitor="val_loss", factor=0.5, patience=20,
min_lr=0.0001
    ),
    keras.callbacks.EarlyStopping(monitor="val_loss",
patience=50, verbose=1),
]
%%%%%%%% Compile and save 1D CNN model %%%%%%%
```

```
model.compile(optimizer="adam",loss=keras.losses.categorical_c
rossentropy,metrics=["accuracy"])
```

```
%%%%%%%% Fit CNN Model %%%%%%%
```

```
history=model.fit(Train_data, Train_Y,
validation_data=(Train_data, Train_Y),
epochs=1000,callbacks=callbacks,
    validation_split=0.2,
    verbose=1,)
model.summary()
```

```
%%%%%%%% Test CNN Model %%%%%%%
```

```
model = keras.models.load_model("best_model.h5")

test_loss, test_acc = model.evaluate(Train_data, Train_Y)

print("Test accuracy", test_acc)
print("Test loss", test_loss)

metric = "accuracy"
plt.figure()
plt.plot(history.history[metric])
plt.plot(history.history["val_" + metric])
plt.title("model " + metric)
plt.ylabel(metric, fontsize="large")
plt.xlabel("epoch", fontsize="large")
plt.legend(["train", "val"], loc="best")
plt.show()
plt.close()
```

A.5: PYTHON CODE COMMANDS FOR 2D-CNN

```
%%%%%%%% import specific libraries required%%%%%%%
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import keras
from   keras.models import Sequential
from keras.layers import Conv2D,MaxPooling2D,Flatten,Dense.
from sklearn.preprocessing import MinMaxScaler
 %%%%%%%% Import of libraries completed %%%%%%%

%%%%%%%% Read CSV file for data and labels%%%%%%%
S = pd.read_csv('Classif.csv', header=None) %%% read csv file
```

```
Label=S.iloc[:,-1]                %%% read labels from last
column

Spectrum=S.iloc[:,:-1].values  %%% read spectra data %%%%
```

```
%%%%%%%% Read CSV file for data and labels%%%%%%%
% define a empty array with 45x128 size
B=np.empty((105,128,128,1))
C=np.empty((128,128))
A=np.empty((105,128,1))
```

```
%%%%%%%% Normalize spectra data with standard scalar%%%%%%%
xscaler= MinMaxScaler()
xscaler.fit((Spectrum))
x=xscaler.transform((Spectrum))
```

```
%%%%%%%% Calculate the correlation matrix from each
spectrum%%%%%%%
```

```
fig = plt.figure(figsize=(15, 15))
for i in range(105):
    A[i]=x[i].reshape(128,1)
    C=A[i]*A[i].transpose()%%% Correlation matrix
    B[i]=C.reshape(128,128,1) %%% reshape Correlation matrix
    ax = plt.subplot(21, 5, i+1)
    plt.imshow(B[i].reshape(128,128)
%%%%%%%% Divide data in to training and testing data%%%%%%%
```

```
Train_data =[]
Train_data=A[0:10,:,:.:]
Train_data=np.append(Train_data,A[15:25,:,:,:],axis=0)
Train_data=np.append(Train_data,A[30:40,:,:,:],axis=0)
Train_data=np.append(Train_data,A[45:55,:,:,;],axis=0)
Train_data=np.append(Train_data,A[60:70,:,:,:],axis=0)
Train_data=np.append(Train_data,A[75:85,:,:,:],axis=0)
Train_data=np.append(Train_data,A[90:100,:,:,:],axis=0)
```

```
Test_data =[]
Test_data=A[10:15,:,:,:]
Test_data=np.append(Train_data,A[25:30,:,:.:],axis=0)
Test_data=np.append(Train_data,A[40:45,:,:,:],axis=0)
Test_data=np.append(Train_data,A[55:60,:,:,:],axis=0)
Test_data=np.append(Train_data,A[70:75,:,:,:],axis=0)
Test_data=np.append(Train_data,A[85:90,:,:,:],axis=0)
Test_data=np.append(Train_data,A[100:105,:,:,:],axis=0)
```

```
%%%%%%%% Training and testing labelled data%%%%%%%
```

```
Train_Y=keras.utils.to_categorical(Train_Label_matrix)
Train_Y.shape
Test_Y=keras.utils.to_categorical(Test_Label_matrix)
Test_Y.shape
%%%%%%%% 1D CNN model Architecture for edible oil
analysis%%%%%%%
```

```
model = Sequential()
model.add(Conv2D(10, kernel_size=(3, 3), strides=(1,
1),activation='relu',input_shape=(128,128,1)))
model.add(MaxPooling2D(pool_size=(2, 2), strides=(1, 1)))
model.add(Conv2D(10, kernel_size=(2, 2), strides=(1,
1),activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2), strides=(1, 1)))
model.add(Flatten())
model.add(Dense(10, activation='relu'))
model.add(Dense(7, activation='softmax'))
```

```
%%%%%%%% Compile and save 1D CNN model %%%%%%%
```

```
callbacks = [
    keras.callbacks.ModelCheckpoint(
        "best_model.h5", save_best_only=True,
monitor="val_loss"
    ),
    keras.callbacks.ReduceLROnPlateau(
        monitor="val_loss", factor=0.5, patience=20,
min_lr=0.0001
    ),
    keras.callbacks.EarlyStopping(monitor="val_loss",
patience=50, verbose=1),
]
```

```
%%%%%%%% Compile and save 1D CNN model %%%%%%%
```

```
model.compile(optimizer="adam",loss=keras.losses.categorical_c
rossentropy,metrics=["accuracy"])
```

```
%%%%%%%% Fit CNN Model %%%%%%%
```

```
history=model.fit(Train_data, Train_Y,
validation_data=(Test_data, Test_Y),
epochs=1000,callbacks=callbacks,
    validation_split=0.2,
    verbose=1,)
model.summary()
```

```
%%%%%%%% Test CNN Model %%%%%%%
```

```
model = keras.models.load_model("best_model.h5")
```

```
test_loss, test_acc = model.evaluate(Train_data, Train_Y)

print("Test accuracy", test_acc)
print("Test loss", test_loss)

metric = "accuracy"
plt.figure()
plt.plot(history.history[metric])
plt.plot(history.history["val_" + metric])
plt.title("model " + metric)
```

List of Publications

Journal Paper:

- **Srinath K** , A.H Kiranmayee .Surekha Bhanot and P.C Panchariya., " Detection and Quantification of palm oil adulteration in Sunflower oil", MAPAN. Under review. **(SCIE)**

- **Srinath K**, Punit Khatri, Surekha Bhanot, and P.C Panchariya."Detection and Quantification of afulteration in groundnut and sesame oils using ATR based Mid infrared spectroscopy coupled with chemometric algorithms ", Food Analytical method, Decision Pending. **(SCIE).**

Conference and Proceedings:

- **Srinath K** , Kiranmayee ,Panchariya Surekha bhanot. "Qualitative and quantitative analysis of cottonseed oil adulteration in groundnut oil using ATR spectroscopy coupled with chemometric strategy" International Conference on Recent Advances in Food Processing Technology (iCRAFPT 2018), 2018, Poster presentation.

- **Srinath K** ,Panchariya Surekha bhanot. " Development of Electronic Tongue for Qualitative Discrimination of Drinking Water - Electrochemical Impedance Spectroscopy Approach" Advances in Microelectronics Instrumentation and Communication (MICOM 2015) BITS Pilani Rajasthan, 2015, .

Brief Biography of Candidate

Mr. Srinath K joined BITS-Pilani in 2013 as a Research Scholar in the Department of Electrical and Electronics Engineering. He has completed his B.Sc. and M.Sc. (Tech.) in 2007 and 2010, respectively. During his Ph.D. tenure, he has published many papers in reputed National and International conferences. Currently, he is working as Manager (Embedded Systems) at Meslova Systems Pvt. Ltd. He has R & D work experience of around 10 years in different organizations, such as CSIR-CEERI, Pilani, Autometers Alliance Ltd., Titagarh Wegans Ltd. His areas of interest are intelligent system design, multivariate data analysis, intelligent pattern recognition algorithms, Embedded systems, and prototype development.

Brief Biography of Supervisor

Prof. Surekha Bhanot obtained her B.E.(Hons) in Mechanical Engineering and M.Phil. (Instrumentation) from BITS Pilani, Rajasthan. She obtained her Ph.D. from IIT Roorkee (then University of Roorkee). She has 17 years of teaching experience in BITS Pilani and 19 years of teaching experience in Thapar University. She is presently Professor at the EEE department BITS Pilani. Her teaching and research areas are sensors, industrial instrumentation & automation, biomedical instrumentation, application of AI techniques in process modeling, control, image processing. She has been guiding projects/thesis at first degree, higher degree, and Ph.D. level and has published papers in international, national journals, and conferences. She has published a book, “Process Control Principles and Applications,” with Oxford University Press. She is the reviewer of many books, research papers, and sponsored projects. She received the best teacher award at Thapar university and was nominated by globally dispersed BITS alumni as a teacher they would especially like to recognize.

Brief Biography of Co-Supervisor

Dr. P.C. Panchariya received his Masters in Physics (with specialization in Electronics and Instrumentation Engineering Sciences) from Devi Ahilya University, Indore. He continued to work on his Ph.D. at the Institute of Instrumentation, Devi Ahilya University, Indore, and the Institute of Automation, Bremen University, Germany. The title of his Ph.D. was Artificial Intelligent System Modelling and Identification**.** He has a specialization in Electronics and Intelligent systems, which includes the development of sensors and signal conditioning, measurement systems, process automation, communication systems having applications in dairy, food, tea & sugar manufacturing, mushroom cultivation, and coal mines industries. He is presently the Director of CSIR-CEERI Pilani Rajasthan. He has 26 years of long experience as a scientist in the field of instrument design and development. He has guided 4 Ph.D. students and currently guiding 6 Ph.D. students, apart from several undergraduate and post-graduate students. He has published several research papers in peer-reviewed and other reputed journals and international conferences. He also has 7 patents and one copyright to his credit. Amongst his many contributions in R&D, the most significant ones are the range of devices developed by him related to milk quality measurement that include Ksheer Scanner & Ksheer Tester (Adulteration Detection System), Milk Fat Tester and Ksheer Analyzer.