"ANALYSIS AND PREDICTION OF SENSORS BASED MULTIVARIATE DATA USING UNSUPERVISED MACHINE LEARNING"

Thesis is submitted for the degree of

Master of Technology

in

Operations Research

By

CHHATRA PAL SINGH
Roll no.: 22MA4105
Reg. No.: 22P10178

Under the Supervision of

DR. ARUN JANA
(Joint Director)

Department of Agriculture and Environment Electronics (AgriEnlcs)

CENTRE FOR DEVELOPMENT OF ADVANCED COMPUTING
(C-DAC) KOLKATA, WB – 700091

DR. GOUTAM PANIGRAHI
(Assistant professor)

Department of Mathematics

NATIONAL INSTITUTE OF TECHNOLOGY
DURGAPUR, WB – 713209

MAY 16, 2024

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in this report entitled "**Analysis and Prediction of Sensors Based Multivariate Data Using unsupervised Machine learning**" in fulfilment of the requirements for the award of the **MASTER OF TECHNOLOGY** in **OPERATIONS RESEARCH** and submitted in the Department of Mathematics, National Institute of Technology, Durgapur is an authentic record of my own work carried by me during Jan 2024 – June 2024, under the guidance of DR. ARUN JANA (Joint Director C-DAC KOLKATA) and DR. GOUTAM PANIGRAHI, Assistant Professor Department of Mathematics, NIT Durgapur.

The matter presented in this report has not been submitted by me for the award of any other degree of this or any other Institute.

CHHATRA PAL SINGH
(22MA4105)
Department of Mathematics
NIT DURGAPUR

This is to certify that the above statement made by the candidate is correct to the best of our knowledge.

DR. GOUTAM PANIGRAHI	**DR. ARUN JANA**	**Head of the Department**
(Supervisor)	**(Supervisor)**	**NIT Durgapur**
NIT Durgapur	**C-DAC Kolkata**	

Date: **Place:** NIT, Durgapur

NIT Durgapur **C-DAC Kolkata**

CERTIFICATE OF RECOMMENDATION

This is to certify that the report entitled "**Analysis and Prediction of Sensors Based Multivariate Data Using unsupervised Machine learning**" submitted by M.Tech Student **CHHATRA PAL SINGH (22MA4105)** as per the requirements of Master of Technology program of National Institute of Technology, Durgapur is a record of bonafied work carried out by him/her under my/our supervision.

Dr. Arun Jana
(Joint Director)
Department of AEE
C-DAC Kolkata

Dr. Goutam Panigrahi
(Assistant professor)
Department of Mathematics
NIT Durgapur - 713209

Date:

Place: NIT, Durgapur

CERTIFICATE OF APPROVAL

This is to certify that we have examined the thesis entitled "**Analysis and Prediction of Sensors Based Multivariate Data Using unsupervised Machine learning**" and hereby accord our approval of it as a study carried out and presented in a manner required for its acceptance in partial fulfilment of the requirements for the award of the degree in **Master of Technology** in **Operations Research** for which it has been submitted. It is to be understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose for which it is submitted.

DR. GOUTAM PANIGRAHI
(Supervisor)

NIT Durgapur

DR. ARUN JANA
(Supervisor)

C-DAC Kolkata

Head of the Department
NIT Durgapur

Date:

Place: NIT, Durgapur

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisor, DR. ARUN JANA (Joint Director C-DAC KOLKATA) and DR. GOUTAM PANIGRAHI, Assistant Professor, Department of Mathematics, NIT Durgapur for providing me with the opportunity to work under his guidance. His continuous support, mentorship, and constructive guidance have been instrumental in shaping my work. I am immensely grateful to him for his unwavering assistance and encouragement throughout the process. I am profoundly grateful to my dear family and friends, especially my parents, for their unconditional love, unwavering support, and endless encouragement. Their belief in my abilities and their constant motivation have been the driving force behind my academic pursuits. I am truly blessed to have them by my side.

CHHATRA PAL SINGH

(22MA4105)

Department of Mathematics
NIT Durgapur

ABSTRACT

This study focuses on assessing the quality of groundnut oil, a crucial parameter in the oil industry, by predicting its fatty acid content through multi-sensor signal analysis. Traditionally, assessing fatty acid content involves costly and complex methods like GC-MS and HPLC, which are unsuitable for routine analysis in continuous production lines due to their expense, need for skilled operators, and time-consuming nature. To address this, the study develops a customized multi-sensor instrument utilizing pattern analysis to test groundnut oil fatty acid content. By employing 8 MOS sensors selected through rigorous procedures, the instrument detects common chemical components in groundnut oil. The sensor array generates signature patterns representing the oil's odor, facilitating fatty acid content prediction. Multivariate Data Analysis, specifically probability neural network, processes the multi-sensor data for accurate prediction.

In addition, the research presents a thorough analysis of blended palm oil and groundnut oil products using clustering and classification techniques. Utilizing an extensive dataset encompassing 11 categories, the study identifies compositional patterns and quality characteristics within the blends. Clustering algorithms reveal natural product groupings based on compositional similarities, while classification models predict blend categories and assess various algorithm performances. These findings contribute valuable insights for stakeholders, aiding in informed decision-making regarding product classification and quality control within the food industry.

TABLE OF CONTENTS

- **Machine Learning Algorithms**

    1. Linear Regression
    2. Logistic Regression
    3. Decision Tree
    4. Random Forest
    5. K Nearest Neighbors
    6. Probabilistic Neural Network

6. Results and Discussions

7. Conclusions

8. Applications and Future Scope

9. References

1. Introduction

• Objective:

The objective for this report is:

1. To develop the OFAM system for quality analysis of groundnut oil based on aroma.
2. Fatty Acid Content is the prime parameter oil quality analysis (oil may be blended or pure).
3. To propose a novel method for assessing the quality of groundnut oil by predicting its fatty acid content using multi-sensor signal analysis.
4. The traditional methods for determining fatty acid content are costly, complex, and unsuitable for routine analysis in continuous production lines.
5. The study aims to develop a customized multi-sensor instrument that can efficiently and accurately predict fatty acid content, thus overcoming the limitations of traditional methods.
6. To conduct a comprehensive analysis of blended palm oil and groundnut oil products using clustering and classification techniques.
7. The research aims to identify compositional patterns and quality characteristics within these blends using an extensive dataset.
8. The findings from this analysis provide valuable insights for stakeholders in the food industry, assisting them in making informed decisions regarding product classification and quality control.

Overall, the objective of the study is to introduce innovative techniques for quality assessment in the oil industry and provide insights that can benefit stakeholders in the food industry.

- # Motivation:

Here are some motivations for the study outlined in this report:

Cost Efficiency: Traditional methods for assessing fatty acid content in groundnut oil, such as GC-MS and HPLC, are expensive and time-consuming. Developing a more cost-effective method can significantly reduce the expenses associated with quality control in the oil industry.

Streamlined Analysis: Continuous production lines require rapid and efficient quality control measures. The proposed multi-sensor instrument offers a streamlined approach to assessing fatty acid content, making it suitable for integration into continuous production processes.

Reduced Dependency on Skilled Operators: Complex methods like GC-MS and HPLC often require highly skilled operators. By developing a more automated and user-friendly instrument, the study aims to reduce the dependency on specialized personnel for quality control tasks.

Enhanced Accuracy: Utilizing multi-sensor signal analysis and pattern recognition techniques can potentially improve the accuracy of fatty acid content predictions compared to traditional methods. This enhanced accuracy is crucial for maintaining product consistency and meeting regulatory standards.

Insights into Blended Products: With the increasing prevalence of blended oil products, there is a need to understand the compositional patterns and quality characteristics of these blends. The study aims to fill this gap by conducting a thorough analysis using clustering and classification techniques, providing valuable insights for stakeholders in the food industry.

Informing Decision-Making: By identifying natural groupings and predicting blend categories, the research outcomes can assist stakeholders in making informed decisions regarding product classification and quality control. This information is essential for ensuring consumer satisfaction and maintaining competitiveness in the market.

Key Points:

Groundnut oil, Palm oil, multi-sensor, phospholipid content, fatty acid content, qualitative data analysis, multi-sensor signal analysis approach, Multivariate Data Analysis, Group Clustering, data processing and prediction.

Groundnut oil (one of the major edible oil) stands as significant edible oil, given its rich nutritional composition and widespread utility. With groundnut kernels containing over 40% oil and 20-30% protein, it emerges as a valuable agricultural commodity. The fatty acid composition of groundnut oil, particularly dominated by oleic and linoleic acids, contributes significantly to its sensory attributes and nutritional value. Notably, India, China, and the USA collectively account for 65% of global groundnut production, with a substantial portion utilized for oil extraction, especially in India where 75-80% of groundnuts are crushed for oil production.

According to Kirk and Sawyer [3], free fatty acid and other fatty materials in ground nut oil plays a major role in the offensive odour and taste also.

However, maintaining the quality of groundnut oil poses challenges, especially concerning its fatty acid content. Free fatty acids and other fatty materials significantly influence the oil's taste and odour, with a tolerance level of free fatty acid in groundnut oil set at less than 5%. Analyzing the fatty acid composition, which includes caprylic, lauric, oleic, linoleic, and other acids, is crucial for assessing the oil's quality.

Table:-1 Summarize the list of fatty acid which are present in the groundnut oil.

Serial number	**Fatty acid**	**Serial number**	**Fatty acid**
1.	Caprylic acid	**7.**	Stearic acid
2.	Capric acid	**8.**	Oleic acid
3.	Lauric acid	**9.**	Linoleic acid
4.	Myristic acid	**10.**	Linolenic acid
5.	Palmitic acid	**11.**	Arachidic acid
6.	Palmitoleic acid	**12.**	Behenic acid

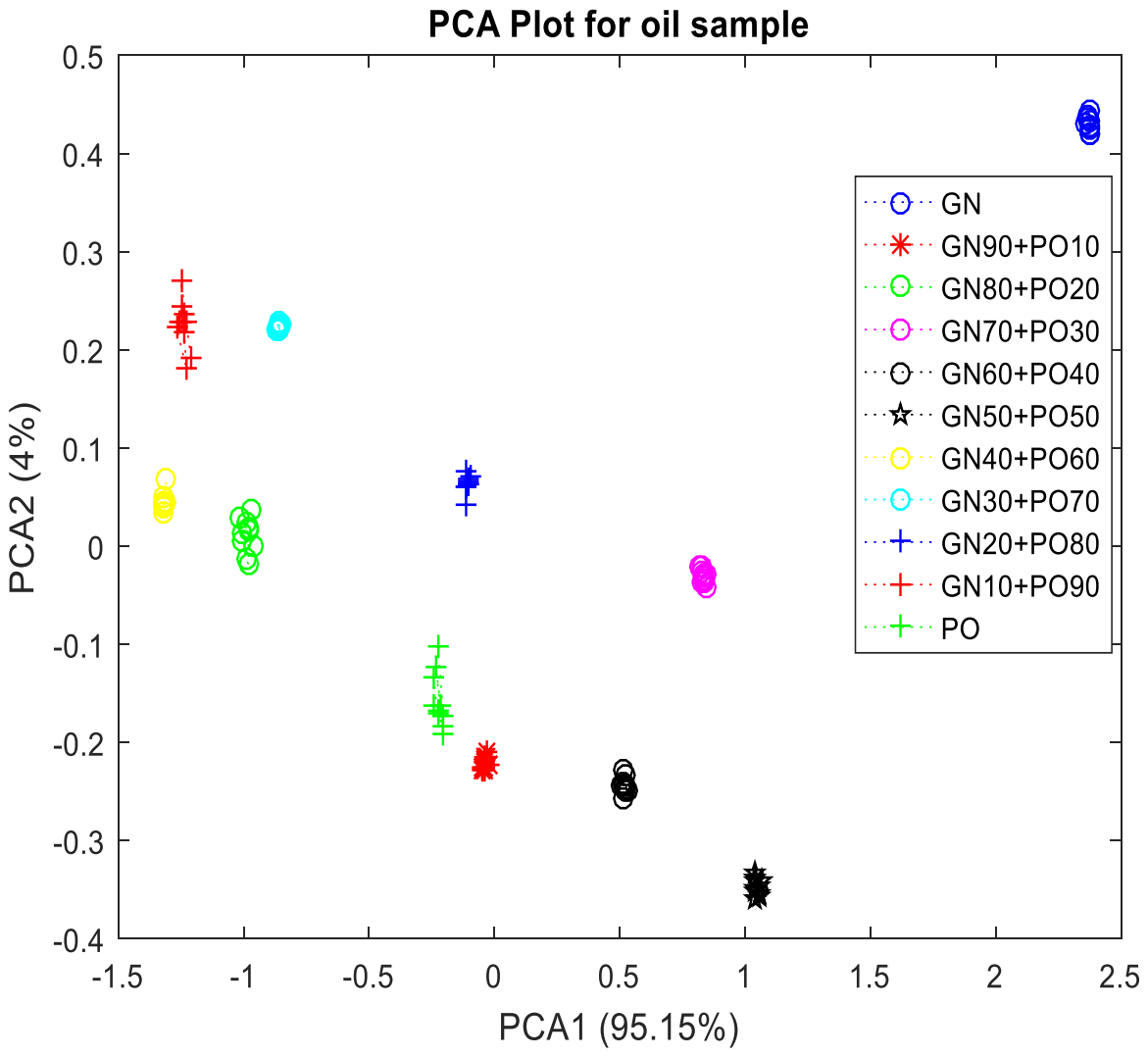
Despite its importance, groundnut oil is vulnerable to adulteration due to its relatively high cost. Unauthorized dealers often blend it with cheaper oils, compromising its quality and authenticity. Thus, there's a pressing need for a simple and rapid method to detect adulteration in groundnut oil.

Conventional analytical methods such as gas chromatography (GC) and high-performance liquid chromatography (HPLC) have been extensively used for this purpose.
However, these methods pose challenges in terms of user-friendliness, skilled manpower requirements, and sample preparation complexity.

To address these challenges, this study proposes a novel approach utilizing a customized multi-sensor instrument for predicting fatty acid content in groundnut oil. This system, known as the Odour and Fatty Acid Measurement (OFAM) system, integrates sensor arrays, odour handling mechanisms, pattern recognition engines, and purging units. By harnessing pattern recognition algorithms like probabilistic neural networks, the OFAM system aims to provide a rapid and user-friendly solution for assessing groundnut oil quality without the need for extensive sample preparation.

This introduction establishes the significance of groundnut oil, highlights challenges in maintaining its quality, and introduces the novel approach proposed in this study for predicting fatty acid content and detecting adulteration.

Description	Number of sample
Pure Ground nut (100% GND)	10
90% GND + 10% PO	10
80% GND + 20% PO	10
70% GND + 30% PO	10
60% GND + 40% PO	10
50% GND + 50% PO	10
40% GND + 60% PO	10
30% GND + 70% PO	10
20% GND + 80% PO	10
10% GND + 90% PO	10
Pure Palm Oil (100% PO)	10
Total	110


PCA Plot for oil sample
PCA2 (4%)
PCA1 (95.15%)
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5
GN
GN90+PO10
GN80+PO20
GN70+PO30
GN60+PO40
GN50+PO50
GN40+PO60
GN30+PO70
GN20+PO80
GN10+PO90
PO

2. Materials

Raw Material: Samples were collected from the Indian Council of Agriculture ResearchIndian Institute of Oilseeds Research (ICAR-IIOR), Hyderabad, India.

Details of samples are given below in Table II.

Serial number	List of major quality Oil Samples
1.	Pure Groundnut Oil
2.	Sample 1 (700+300) blended with palm oil
3.	Sample 2 (800 + 200) blended with palm oil
4.	Sample 3 (900 + 100) blended with palm oil
5.	Pure Palm Oil

Experiments with GC were carried out in the Indian Council of Agriculture Research-Indian Institute of Oilseeds Research (ICAR-IIOR), Hyderabad, India. Results are shown in the Table III.

GC report of oil sample

Sample ID	Myristic	Palmetic	Stearic	Oleic	Linoleic	Linolenic
Pure Groundnut Oil (GND)	0.02	12.42	4.58	42.64	38.05	0.23
Pure Palm Oil (PO)	0.56	27.40	4.88	47.25	18.65	0.75
Sample I[(GND (700 ml + PO 300 ml)]	0.05	13.15	4.76	42.98	36.75	0.13
Sample II[(GND (800 ml + PO 200 ml)]	0.09	14.42	4.65	43.30	34.97	0.21
Sample III[(GND (900 ml + PO 100 ml)]	0.15	16.05	4.83	43.49	33.35	0.35

Fabrication of OFAM System:-

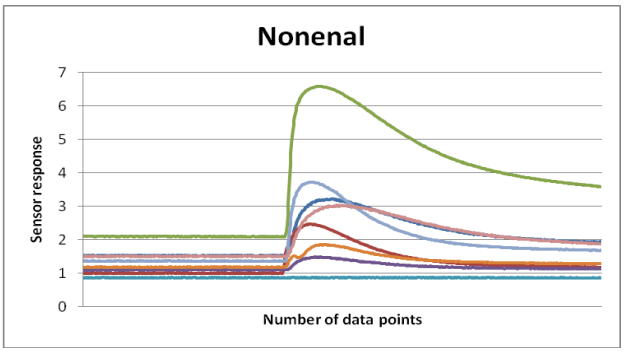
Before the development of the Odour and Fatty Acid Measurement (OFAM) system, an essential step involved the careful selection of sensors based on their sensitivity to pure chemicals. Through an extensive literature survey, specific chemicals crucial for determining the quality of groundnut oil were identified. Table IV presents a list of these key chemicals procured from the market. Subsequently, 30 sensors were evaluated for their response to these target chemicals, leading to the final selection of eight non-specific Metal Oxide Semiconductor sensors based on correlation matrix analysis.

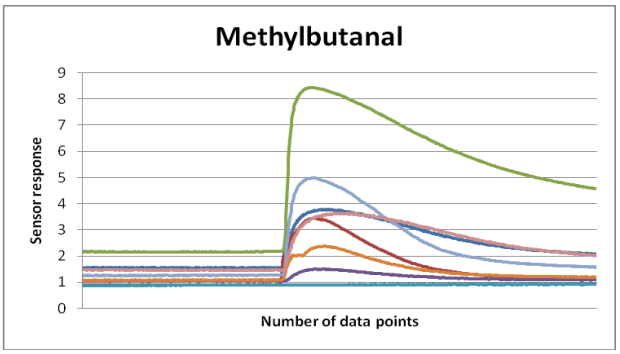
Table IV: List of chemicals

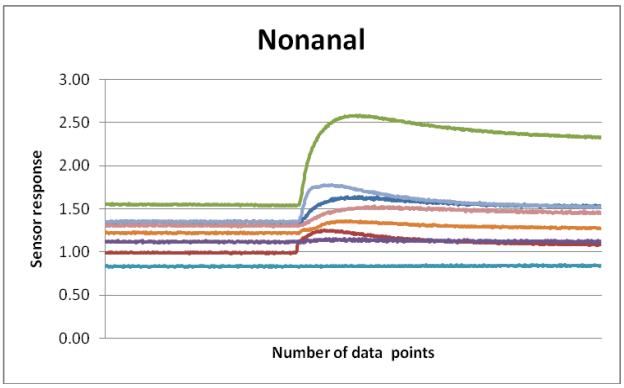
Serial number	List of major quality deterministic chemicals
1	Nonanal
2	Isovaleraldehyde
3	Methylpyrrole
4	Nonenal
5	Methylbutanal
6	Hexanal
7	Menthol
8	Decadienal

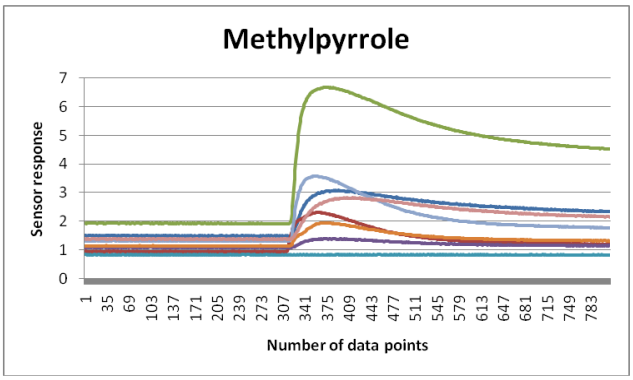
The presence of volatile organic compounds (VOCs) in groundnut oil, particularly during heating processes, necessitated the design and development of a separate sample preparation unit. This unit includes a volatile component delivery system equipped with heating and raking arrangements. To ensure efficient release of volatile compounds, samples are heated to 60°C with motorized agitation, facilitating comprehensive analysis and accurate measurement within the OFAM system.

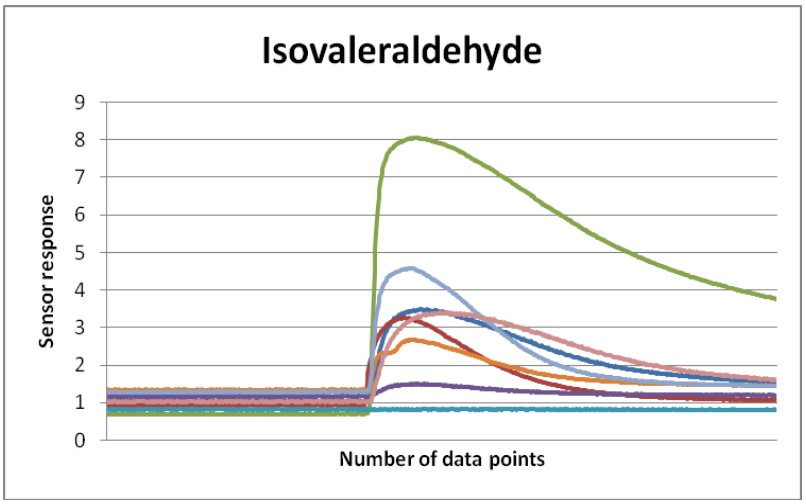
Sensor response graph for different raw chemicals



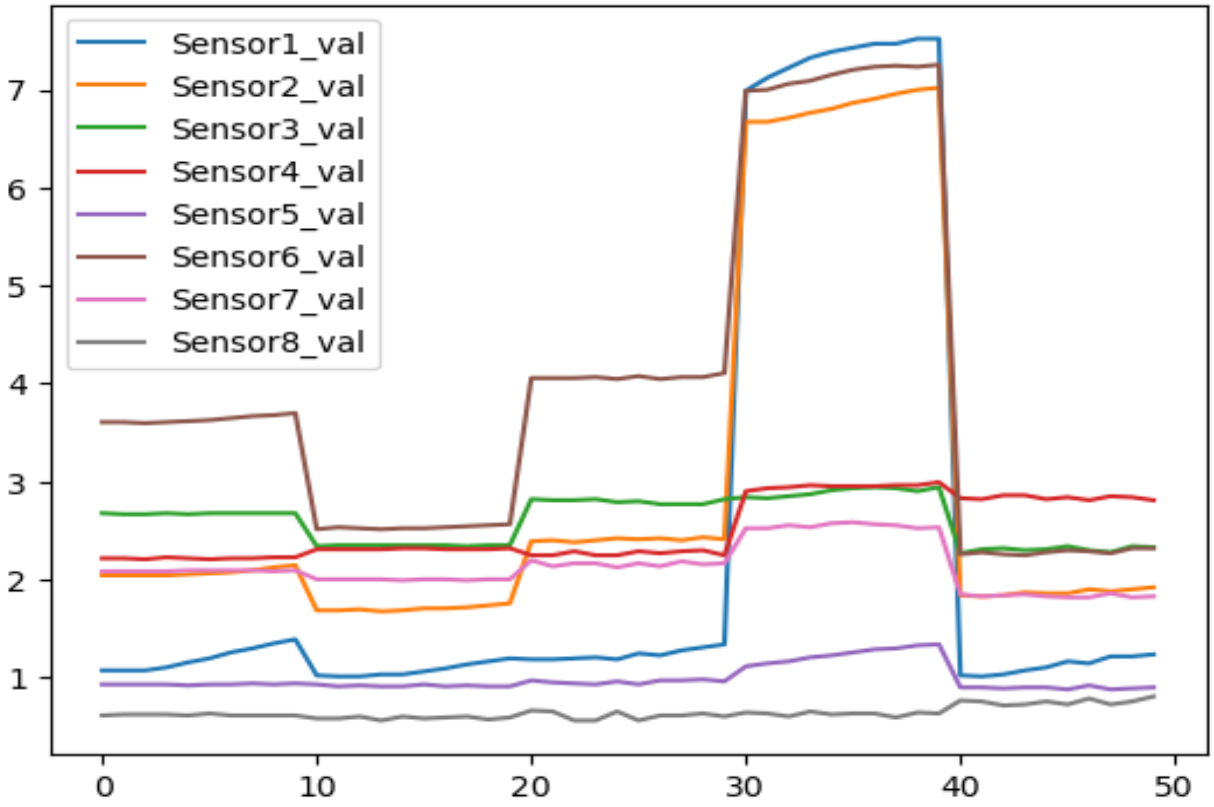


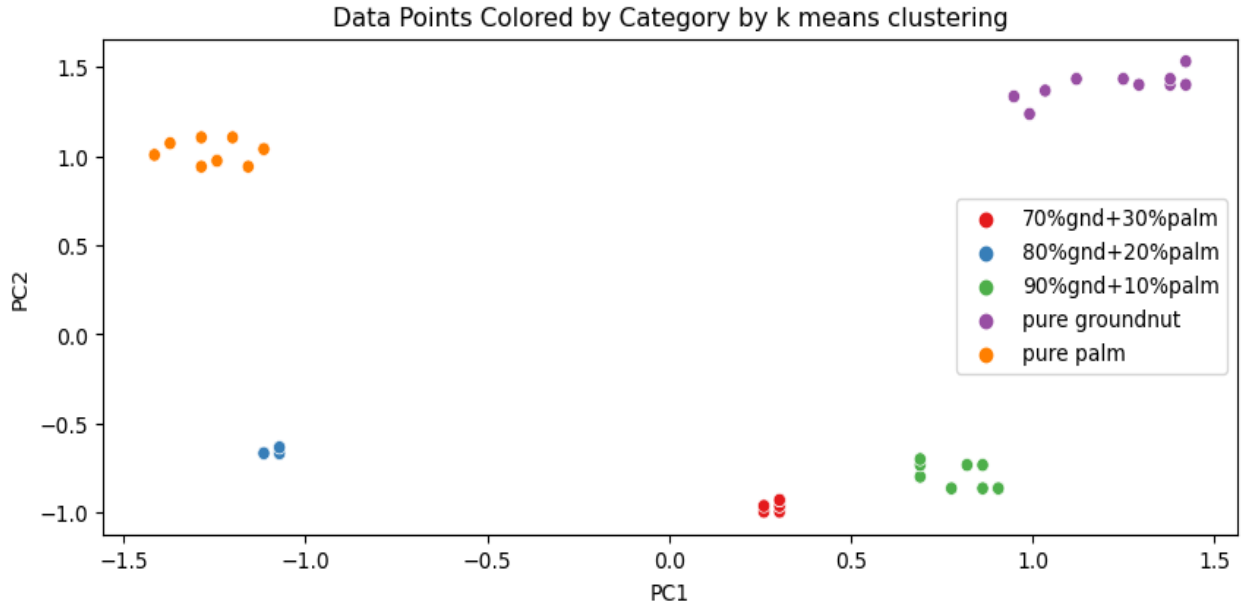






Sensor response graph for Oil Samples

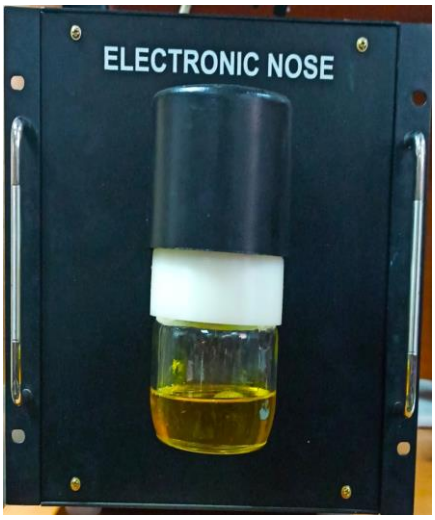


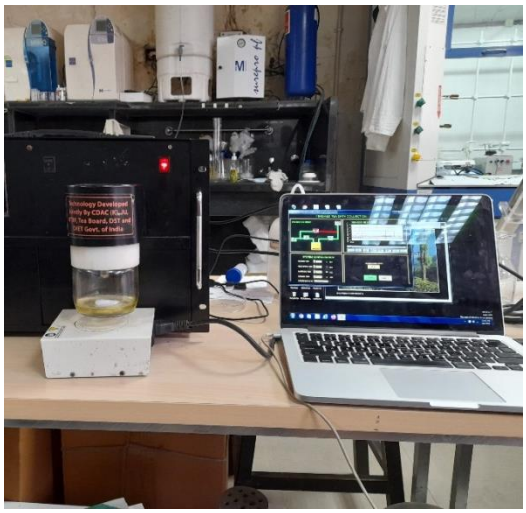


3. Methods

The Odour Handling and Delivery (OHD) unit of the OFAM system was meticulously designed to accommodate various functionalities crucial for the analysis of groundnut oil samples. This unit not only enables the heating of oil samples up to 60°C but also facilitates the generation of pressure within the sample container through a racking arrangement. Constructed with borosilicate materials for durability and chemical resistance, the OHD unit is an integral component of the OFAM system setup.

Fabrication of OFAM System:





To initiate the headspace generation process, a 30 ml oil sample containing magnetic beads is placed within the borosilicate sample container. Subsequently, the sample undergoes heating while being subjected to raking at a fixed RPM. This heating and agitation process aids in the efficient release of volatile compounds from the oil sample, ensuring a robust headspace generation.
During the headspace operation, meticulous attention is paid to ensuring that the concentration of volatiles in the blended oil increases significantly. This heightened concentration ensures that an ample amount of volatile components are available for detection by the sensor array during the sampling operation. Moreover, a constant flow of volatiles is maintained through pipelines within the sensor chamber, ensuring continuous exposure of the sensor array to the volatile components of the oil sample.

Following the sampling operation, the purging operation is initiated to clear the sensor surfaces. This purging process involves the application of a blow of fresh air to the sensor array, effectively resetting the sensors to their initial values. By initializing the sensor array in this manner, the OFAM system ensures consistency and accuracy in subsequent measurements and analyses.

4. Basics of Machine Learning

Machine Learning: Machine learning is a branch of artificial intelligence (AI) that focuses on developing algorithms and statistical models that enable computers to learn and improve from experience without being explicitly programmed. In essence, it empowers machines to recognize patterns in data and make predictions or decisions based on that data, with the ability to adapt and improve over time as they are exposed to more data.

Machine learning can be categorized into several types based on different criteria. Here are some common types:

Supervised Learning: In supervised learning, the algorithm is trained on a labeled dataset, where each input is associated with the correct output. The algorithm learns to map inputs to outputs, making predictions or decisions based on new data.

Unsupervised Learning: Unsupervised learning involves training algorithms on unlabeled data, where the algorithm tries to find patterns or structure in the data on its own. It aims to discover hidden patterns or groupings in the data without explicit guidance.

Semi-supervised Learning: Semi-supervised learning combines elements of both supervised and unsupervised learning. It leverages a small amount of labeled data along with a larger amount of unlabeled data to improve learning accuracy.

Reinforcement Learning: Reinforcement learning is a type of machine learning where an agent learns to interact with an environment by performing actions and receiving rewards or penalties in return. The agent learns to maximize cumulative rewards over time through trial and error.

Deep Learning: Deep learning is a subset of machine learning that uses neural networks with multiple layers (deep neural networks) to learn complex patterns in large amounts of data. Deep learning has shown remarkable success in tasks such as image and speech recognition, natural language processing, and more.

Transfer Learning: Transfer learning involves transferring knowledge from one domain or task to another. In this approach, a model trained on one task is adapted or fine-tuned to perform a related task, often with less data and training time required.

Online Learning: Online learning, also known as incremental learning or lifelong learning, involves updating the model continuously as new data becomes available. This type of learning is particularly useful in scenarios where data streams in real-time or evolves over time.

These are some of the common types of machine learning, each with its own techniques, algorithms, and applications.

Here we are using only Supervised and Unsupervised Machine Learning Algorithms

5. Description and use of machine learning

A total of 50 samples were meticulously chosen for experimental purposes, as detailed in Table II. These samples were categorized into five distinct groups, with each group comprising 10 data points. To facilitate analysis, a custom sensor array consisting of eight sensors was fabricated. Each sample was subjected to analysis using this array, resulting in a final analytical dimension of 50x8.
The generated matrix, with dimensions 50x8, served as the basis for subsequent analysis tasks. Principal Component Analysis (PCA) was employed for clustering purposes.

I have used here K means clustering for cluster all the data points of each groups and utilizing the matrix to identify underlying patterns and groupings within the data.

I have also used here Linear Discriminant Analysis that is one of the commonly used dimensionality reduction techniques in machine learning to solve more than two-class classification problems. It is also known as Normal Discriminant Analysis (NDA) or Discriminant Function Analysis (DFA).

Additionally, for classification tasks, I have used many distinct machine learning algorithms employed:

1. Linear Regression
2. Logistic Regression
3. K Nearest Neighbors classifier
4. Decision Tree
5. Random Forest
6. Probabilistic Neural Network

These algorithms leveraged the data matrix to accurately classify samples into their respective categories, providing valuable insights into the composition and characteristics of the analysed Samples.

- **Clustering and Dimensionality Reduction**

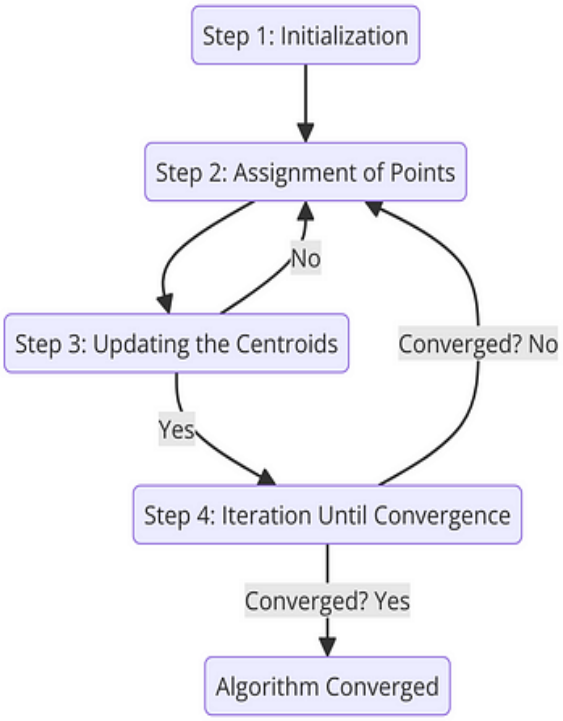
Clustering: A process of organizing objects into groups such that data points in the same groups are similar to the data points in the same group. A cluster is a collection of objects where these objects are similar and dissimilar to the other cluster.

1. K means clustering:

K-Means Clustering is an Unsupervised Learning Algorithm, which groups the unlabeled dataset into different clusters. There is no labeled data for this clustering, unlike in supervised learning. K-Means performs the division of objects into clusters that share similarities and are dissimilar to the objects belonging to another cluster. The term 'K' is a number. You need to tell the system how many clusters you need to create. For example, K = 2 refers to two clusters. There is a way of finding out what is the best or optimum value of K for a given data. The k-means clustering algorithm mainly performs two tasks:

- Determines the best value for K center points.
- Assigns each data point to its closest K-center. Groups assign based on k center points by measuring the distance between k points and data points.

Implications of K-means Clustering:



Mathematics behind K means Clustering:

Let's put on our math hats for a moment and peek into the engine room of the K-Means algorithm to see what makes it tick. K-Means is all about finding the sweet spot:
"The optimal placement of centroids that minimizes the distance between points in a cluster and their central point".

K-Means Clustering Algorithm involves the following steps:

Step 1: Calculate the number of K (Clusters).

Step 2: Randomly select K data points as cluster center.

Step 3: Using the Euclidean distance formula measure the distance between each data point and each cluster center.

$$\boldsymbol{Distance} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Step 4: Assign each data point to that cluster whose center is nearest to that data point.
Assign point x to cluster C_i if

$$\boldsymbol{d(x, \mu_i) \leq d(x, \mu_j)\ \forall j, 1 \leq j \leq k}$$

Step 5: Re-compute the center of newly formed clusters. The center of a cluster is computed by taking the mean of all the data points contained in that cluster.

New centroid position μ_i' is calculated as

$$\boldsymbol{\mu_i' = \frac{1}{|C\_i|} \sum_{x \epsilon c_i} x}$$

Step 6: Keep repeating the procedure from Step 3 to Step 5 until any of the following stopping criteria is met-

- If data points fall in the same cluster.
- Reached maximum of iteration.
- The newly formed cluster does not change in center points.

According to the coding background the data points are clustered like:

2. Principal Component Analysis (PCA):

Principal component analysis (PCA) is a widely covered machine learning method on the web. And while there are some great articles about it, many go into too much detail. Below we cover how principal component analysis works in a simple step-by-step way, so everyone can understand it and make use of it — even those without a strong mathematical background. Principal component analysis (PCA) is a dimensionality reduction and machine learning method used to simplify a large data set into a smaller set while still maintaining significant patterns and trends.

Principal component analysis, or PCA, is a dimensionality reduction method that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information in the large set.

Reducing the number of variables of a data set naturally comes at the expense of accuracy, but the trick in dimensionality reduction is to trade a little accuracy for simplicity. Because smaller data sets are easier to explore and visualize, and thus make analyzing data points much easier and faster for machine learning algorithms without extraneous variables to process.

So, to sum up, the idea of PCA is simple: reduce the number of variables of a data set, while preserving as much information as possible.

What are Principal Components: Principal components are new variables formed by combining the original variables in a way that maximizes information retention. They are constructed as linear combinations of initial variables, ensuring that they are uncorrelated and capture most of the information from the original dataset.

The goal of PCA is to reduce dimensionality without significant loss of information. It does so by prioritizing the extraction of maximum information in the first principal component, followed by the subsequent components in decreasing order of importance. This organization facilitates dimensionality reduction by discarding components with low information while preserving meaningful patterns.

Despite their lack of direct interpretability, principal components represent directions in the data that explain the highest variance. Essentially, they capture the most informative aspects of the dataset. Visually, principal components can be viewed as axes that offer the optimal perspective to understand the differences between observations, highlighting variations in the data

How PCA Constructs the Principal Components:

PCA constructs principal components in a step-by-step manner, aiming to maximize the variance captured by each component while ensuring orthogonalit (uncorrelatedness) between them.

First Principal Component: The initial principal component is determined to capture the maximum variance present in the dataset. This component is aligned in the direction where the projected

Subsequent Principal Components: Each successive principal component is calculated while ensuring it remains orthogonal to the previous components. This means that the second principal component is perpendicular to the first one. The second component captures the next highest variance in the dataset, orthogonal to the first, and so forth.
Number of Principal Components: The process continues until a total of p principal components have been derived, where p equals the original number of variables in the dataset.
By constructing principal components in this manner, PCA effectively reduces the dimensionality of the dataset while preserving the most significant patterns and trends present in the data.

Principal component analysis can be broken down into five steps:

Step 1: Standardization:

Standardize the range of continuous initial variables to ensure each contributes equally to the analysis.

$$z = \frac{\boldsymbol{value - mean}}{\boldsymbol{standerd\ deviation}}$$

Step 2: Covariance Matrix Computation:

Compute the covariance matrix to understand how variables vary relative to each other. Identifies correlations and redundant information among variables. The covariance matrix is a $p \times p$ symmetric matrix (where p is the number of dimensions) that has as entries the covariances associated with all possible pairs of the initial variables. For example, for a 3-dimensional data set with 3 variables x, y, and z, the covariance matrix is a 3×3 data matrix of this from: Covariance matrix is a symmetric matrix representing covariances between all pairs of variables.

$$\begin{bmatrix} Cov(x,x) & Cov(x,y) & Cov(x,z) \\ Cov(y,x) & Cov(y,y) & Cov(y,z) \\ Cov(z,x) & Cov(z,y) & Cov(z,z) \end{bmatrix}$$

Step 3: Compute Eigenvalues and Eigenvectors:

Eigenvalues and eigenvectors are computed from the covariance matrix.
Eigenvectors represent directions of axes with the most variance (principal components), and eigenvalues represent the amount of variance carried by each component.

Step 4: Create a Feature Vector:

Choose whether to keep all principal components or discard less significant ones.
Form a feature vector with eigenvectors of selected components.
Dimensionality reduction is achieved by choosing fewer eigenvectors, reducing the final dataset's dimensions.

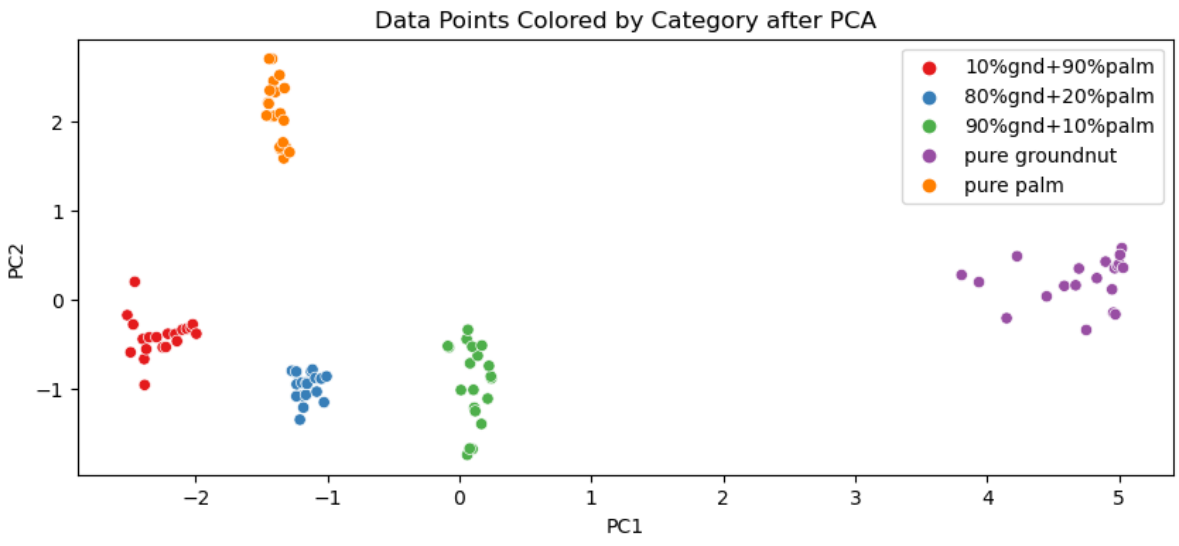
Step 5: Recast the Data Along Principal Components Axes:

Reorient the data from original axes to those represented by principal components.
Multiply the transpose of the original dataset by the transpose of the feature vector.

By following these steps, PCA extracts meaningful patterns and trends from high-dimensional datasets, facilitating data analysis and visualization.

$$FinalDataSet = FeatureVector^T * StandardizedOriginalDataSet^T$$

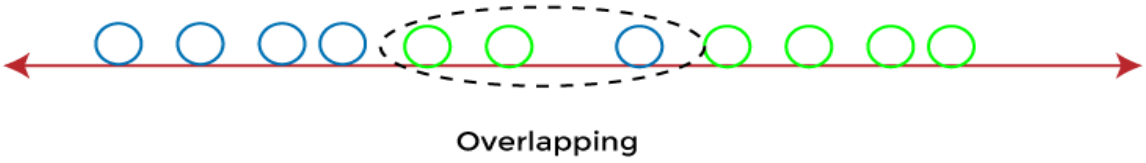
According to the coding background the data points are clustered after PCA like:



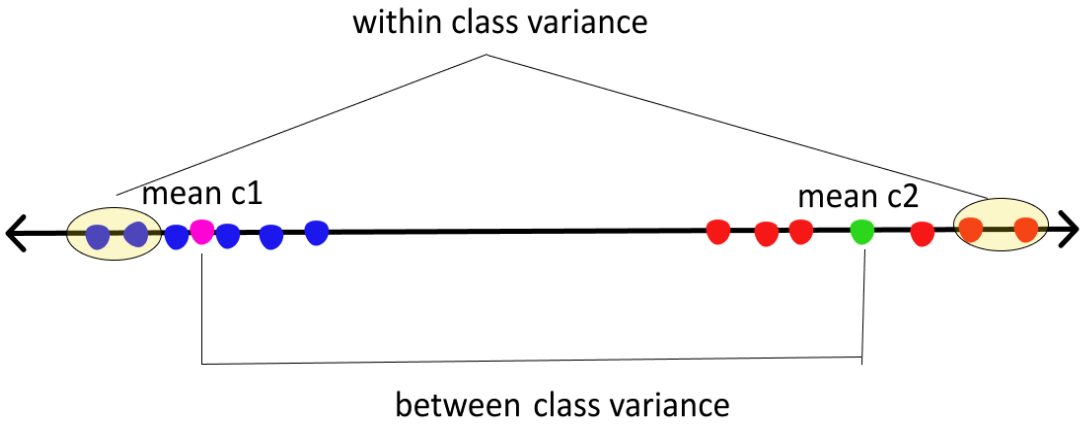
3. Linear Discriminant Analysis (LDA) :

Linear Discriminant analysis is one of the most popular dimensionality reduction techniques used fo r supervised classification problems in machine learning. It is also considered a pre-processing step for modeling differences in ML and applications of pattern classification.
Although the logistic regression algorithm is limited to only two-class, linear Discriminant analysis is applicable for more than two classes of classification problems.

Whenever there is a requirement to separate two or more classes having multiple features efficiently , the Linear Discriminant Analysis model is considered the most common technique to solve such classification problems. For e.g., if we have two classes with multiple features and need to separate them efficiently. When we classify them using a single feature, then it may show overlapping.



Overlapping



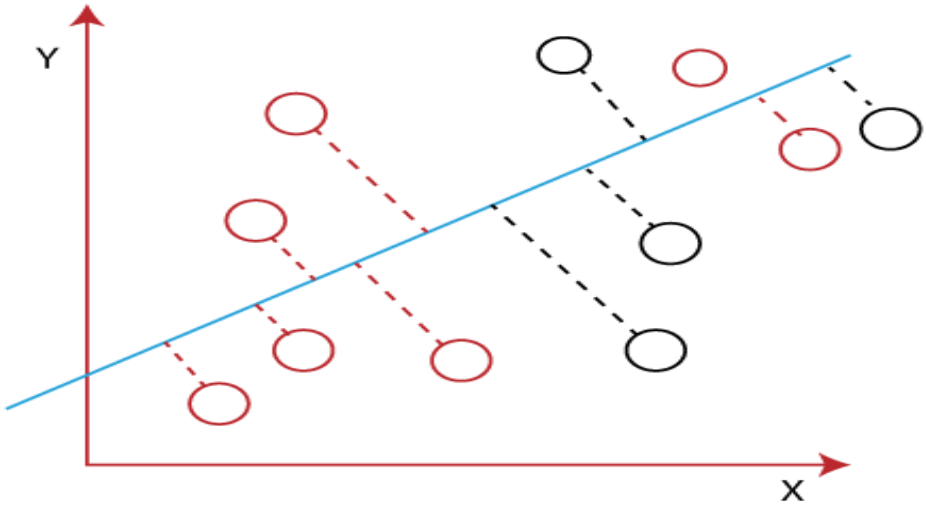
To overcome the overlapping issue in the classification process, we must increase the number of feat ures regularly.

How Linear Discriminant Analysis (LDA) Works:

Linear Discriminant analysis is used as a dimensionality reduction technique in machine learning, using which we can easily transform a 2-D and 3-D graph into a 1-dimensional plane.

Let's consider an example where we have two classes in a 2-D plane having an X-Y axis, and we need to classify them efficiently. As we have already seen in the above example that LDA enables us to draw a straight line that can completely separate the two classes of the data points. Here, LDA uses an X-Y axis to create a new axis by separating them using a straight line and projecting data onto a new axis.

Hence, we can maximize the separation between these classes and reduce the 2-D plane into 1-D.



To create a new axis, Linear Discriminant Analysis uses the following criteria:

1. It maximizes the distance between means of two classes.
2. It minimizes the variance within the individual class.

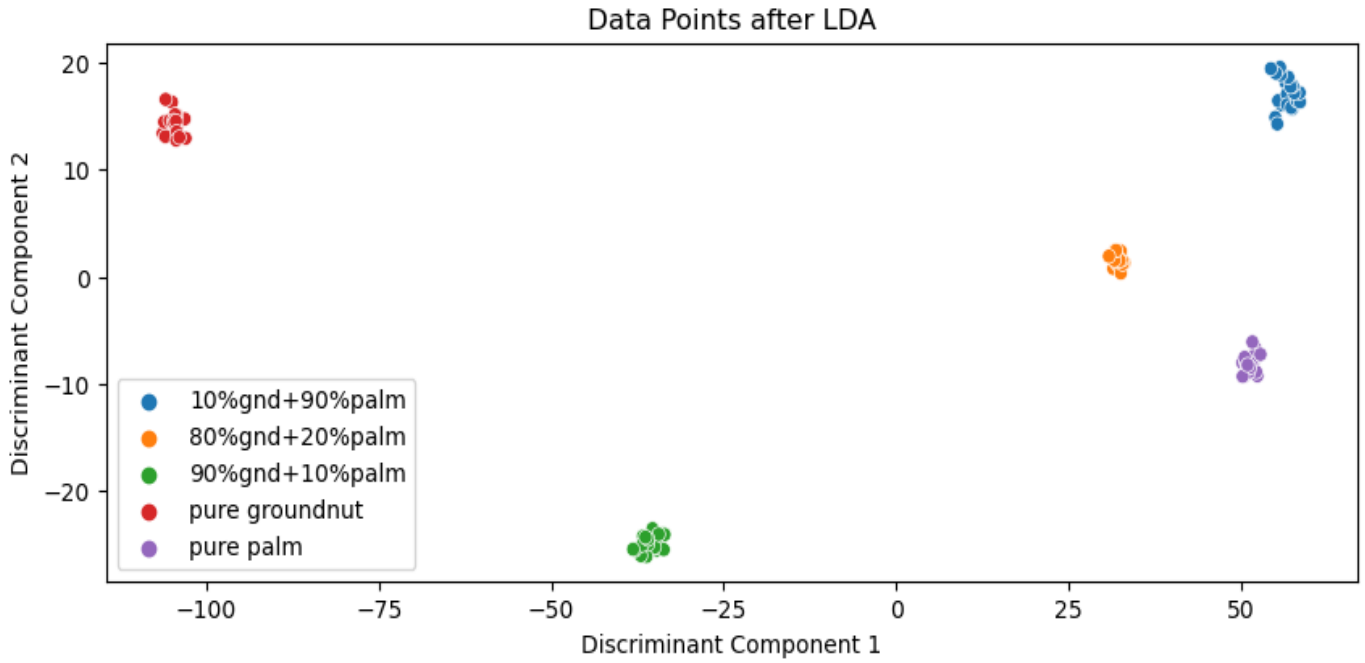
Using the above two conditions, LDA generates a new axis in such a way that it can maximize the distance between the means of the two classes and minimizes the variation within each class.
In other words, we can say that the new axis will increase the separation between the data points of the two classes and plot them onto the new axis.

Why LDA?

Logistic Regression is one of the most popular classification algorithms that perform well for binary classification but falls short in the case of multiple classification problems with well-separated classes. At the same time, LDA handles these quite efficiently.

- LDA can also be used in data pre-processing to reduce the number of features, just as PCA, which reduces the computing cost significantly.
- LDA is also used in face detection algorithms. In Fisherfaces, LDA is used to extract useful data from different faces. Coupled with eigenfaces, it produces effective results.

According to coding background the following results has been found:



Difference between LDA and PCA

Below are some basic differences between LDA and PCA:

- PCA is an unsupervised algorithm that does not care about classes and labels and only aims to find the principal components to maximize the variance in the given dataset. At the same time, LDA is a supervised algorithm that aims to find the linear discriminants to represent the axes that maximize separation between different classes of data.

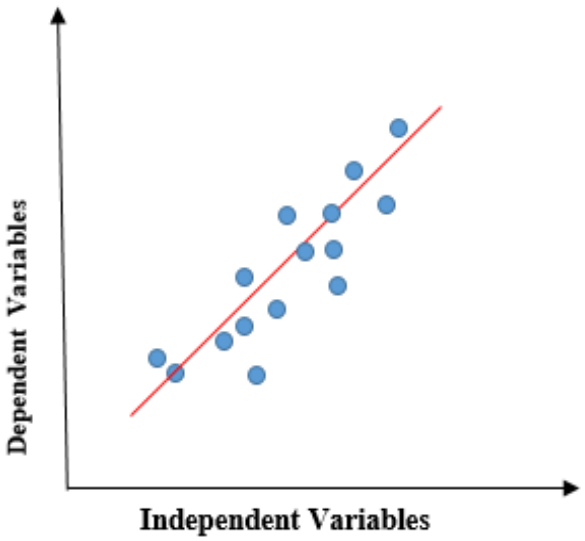
- LDA is much more suitable for multi-class classification tasks compared to PCA. However, PCA is assumed to be an as good performer for a comparatively small sample size.

- Both LDA and PCA are used as dimensionality reduction techniques, where PCA is first followed by LDA.

- **Machine Learning Algorithm Descriptions**

1. Linear Regression:

Linear regression is a quiet and simple statistical regression method used for predictive analysis and shows the relationship between the continuous variables. Linear regression shows the line ar relationship between the independent variable (X-axis) and the dependent variable (Y-axis), c onsequently called linear regression. If there is a single input variable (x), such linear regression is called simple linear regression. And if there is more than one input variable, such linear regre ssion is called multiple linear regression. The linear regression model gives a sloped straight lin e describing the relationship within the variables.



The above graph presents the linear relationship between the dependent variable and independent variables. When the value of x (independent variable) increases, the value of y (dependent variable) is likewise increasing. The red line is referred to as the best fit straight line. Based on the given data points, we try to plot a line that models the points the best.
To calculate best-fit line linear regression uses a traditional slope-intercept form.

$$y = mx + b \Rightarrow y = a_0 + a_1 x$$

Where

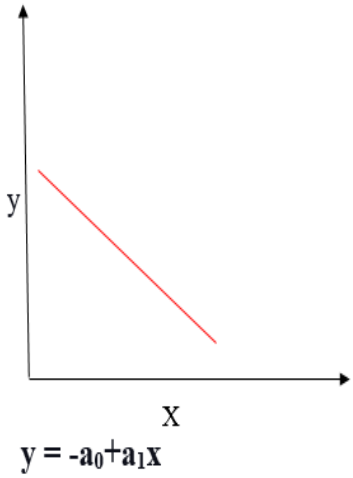
y = Dependent Variable, x = Independent Variable, a_0 = intercept of the line, and a_1 = Linear regression coefficient.

Here a and b can be find by using:

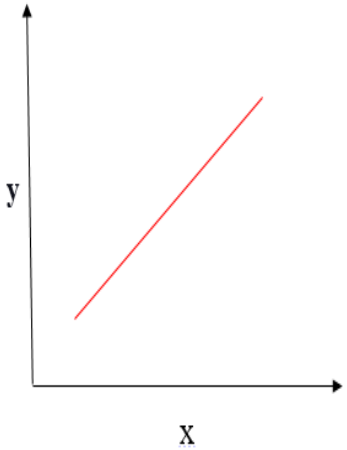
$$a = \frac{[(\sum y)(\sum x^2) - (\sum x)(\sum xy)]}{[n(\sum x^2) - (\sum x)^2]}$$

$$b = \frac{[n(\sum xy) - (\sum x)(\sum y)]}{[n(\sum x^2) - (\sum x^2)]}$$

If the dependent variable expands on the Y-axis and the independent variable progress on X-axis, then such a relationship is termed a Positive linear relationship.



$y = -a_0 + a_1 x$



$y = a_0 + a_1 x$

If the dependent variable decreases on the Y-axis and the independent variable increases on the X-axis, such a relationship is called a negative linear relationship.

The goal of the linear regression algorithm is to get the best values for a0 and a1 to find the best fit line. The best fit line should have the least error means the error between predicted values and actual values should be minimized.

2. Logistic Regression:

Logistic regression is a supervised machine learning algorithm used for classification tasks where the goal is to predict the probability that an instance belongs to a given class or not. Logistic regression is a statistical algorithm which analyze the relationship between two data factors.
The article explores the fundamentals of logistic regression, it's types and implementations.
Logistic regression is used for binary classification where we use sigmoid function, that takes input as independent variables and produces a probability value between 0 and 1.

For example, we have two classes Class 0 and Class 1 if the value of the logistic function for an input is greater than 0.5 (threshold value) then it belongs to Class 1 it belongs to Class 0. It's referred to as regression because it is the extension of linear regression but is mainly used for classification problems.

How Linear Regression Works:

The logistic regression model transforms the linear regression function continuous value output into categorical value output using a sigmoid function, which maps any real-valued set of independent variables input into a value between 0 and 1. This function is known as the logistic function.

Let the independent input features be:

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1m} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mm} \end{bmatrix}$$

and the dependent variable is Y having only binary value i.e. 0 or 1.

$$Y = \begin{cases} 0 & if\ class\ 1 \\ 1 & if\ class\ 2 \end{cases}$$

then, apply the multi-linear function to the input variables X.

$$z = \sum_{i=1}^{n} w_i\, x_i$$

Here x_i is the ith observation of X, $w_i = w_1, w_2, \ldots, w_m$ is the weights or Coefficient, and b is the bias term also known as intercept.

Simply this can be represented as the dot product of weight and bias.

$$z = w.X + b$$

whatever we discussed above is the linear regression.

3. K Nearest Neighbors classifier:

KNN stands for K-nearest neighbour, it's one of the Supervised learning algorithm mostly used for classification of data on the basis how it's neighbour are classified. KNN stores all available cases and classifies new cases based on a similarity measure. K in KNN is a parameter that refers to the number of the nearest neighbours to include in the majority voting process.
How do we choose K?

$\sqrt{n}$, where n is a total number of data points(if in case n is even we have to make the value odd by adding 1 or subtracting 1 that helps in select better)
When to use KNN?

We can use KNN when Dataset is labelled and noise-free and it's must be small because KNN is a "Lazy learner".

How KNN works:

Distance Metric: KNN relies on a distance metric to determine the 'closeness' of data points.

The most commonly used distance metrics are Euclidean distance and Manhattan distance, though other metrics like Murkowski distance can also be used.

- **Euclidean Distance:**
  For two points $(x_1, y_1)$and (x_2, y_2) in a 2-dimensional space, the Euclidean distance is calculated as:

$$d^2 = (x_2 - x_1)^2 + (y_2 - y_1)^2$$

- **Manhattan Distance:**
  For two points $(x_1, y_1)$and (x_2, y_2) in a 2-dimensional space, the Manhattan distance is calculated as the sum of the absolute differences of their coordinates:

$$d = |x_2 - x_1| + |y_2 - y_1|$$

KNN Algorithm:

- **Training Phase:** In the training phase, KNN simply memorizes the entire training dataset. No explicit training is involved as there are no model parameters to learn.

- **Prediction Phase:** Given a new data point x_{test} , KNN identifies the k closest data points in the training set based on the chosen distance metric.

- For classification, it assigns the majority class among the k nearest neighbours to x_{test}
- For regression, it assigns the average (or weighted average) of the target values of the k nearest neighbours x_{test}.

Choice of k:

- The parameter k represents the number of nearest neighbours to consider.
- The choice of k can significantly impact the performance of the algorithm. Smaller values of k lead to more complex decision boundaries, while larger values of k lead to smoother decision boundaries.

Weighted KNN:

In weighted KNN, instead of simply considering the majority vote of k nearest neighbours in classification (or the simple average in regression), the contributions of neighbours can be weighted based on their distance to the query point. This means closer neighbours have a greater influence on the prediction than farther ones.

Cross-validation: Cross-validation is often used to determine the optimal value of k and to evaluate the performance of the KNN algorithm on unseen data.

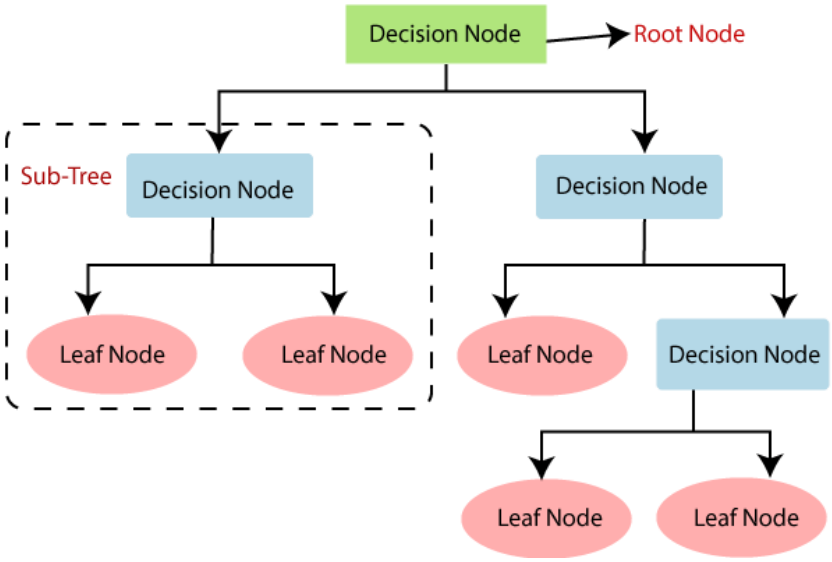
Curse of Dimensionality: KNN can suffer from the curse of dimensionality, where the distance between data points becomes less meaningful as the dimensionality of the feature space increases. This can lead to degraded performance, especially with high-dimensional data.

Scalability: KNN can be computationally expensive, especially when dealing with large datasets, since it requires computing distances between the query point and all training points. Approximate nearest neighbour techniques are often employed to address this scalability issue.

Understanding these mathematical concepts is crucial for effectively implementing and tuning the KNN algorithm for different machine learning tasks.

4. Decision Tree Classifier:

A decision tree is a type of supervised learning algorithm that is commonly used in machine learning to model and predict outcomes based on input data. It is a tree-like structure where each internal node tests on attribute, each branch corresponds to attribute value and each leaf node represents the final decision or prediction. The decision tree algorithm falls under the category of supervised learning. They can be used to solve both regression and classification problems.



How Decision Tree formed:

The process of forming a decision tree involves recursively partitioning the data based on the values of different attributes. The algorithm selects the best attribute to split the data at each internal node, based on certain criteria such as information gain or Gini impurity. This splitting process continues until a stopping criterion is met, such as reaching a maximum depth or having a minimum number of instances in a leaf node.

Decision tree uses the tree representation to solve the problem in which each leaf node corresponds to a class label and attributes are represented on the internal node of the tree. We can represent any Boolean function on discrete attributes using the decision tree.

Below are some assumptions that we made while using decision tree

At the beginning, we consider the whole training set as the root.

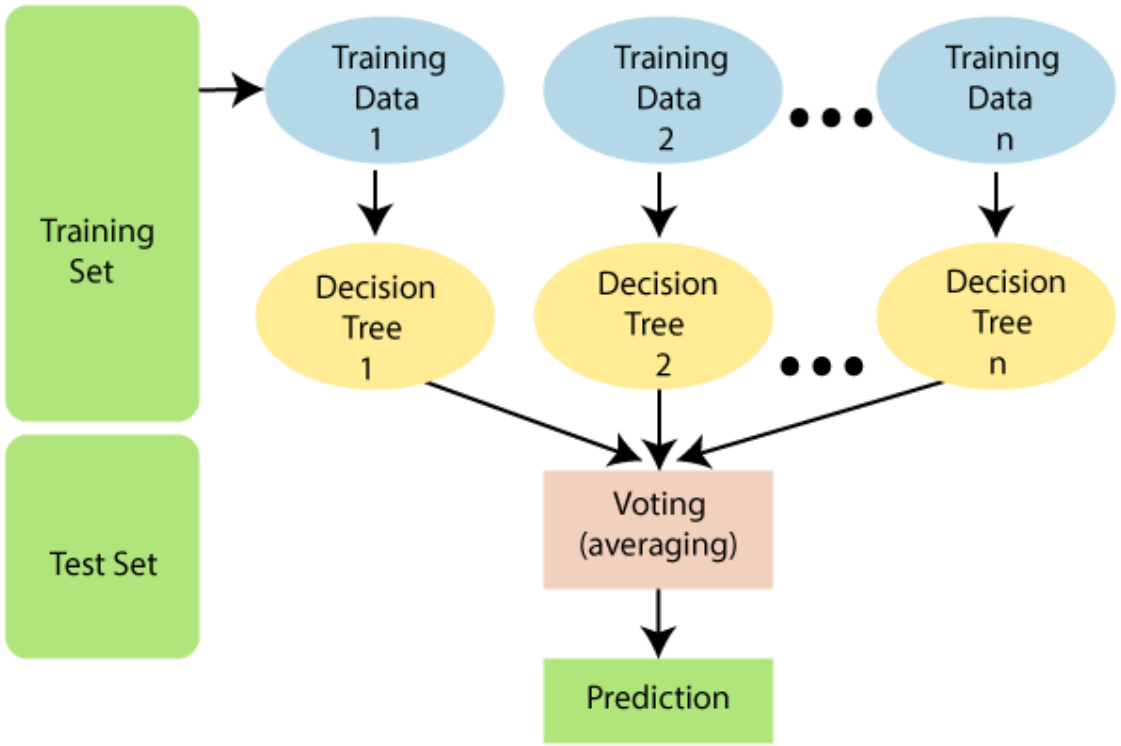
- Feature values are preferred to be categorical. If the values are continuous then they are discretized prior to building the model.
- On the basis of attribute values, records are distributed recursively.
- We use statistical methods for ordering attributes as root or the internal node.

5. Random Forest classifier:

Random Forest is a widely-used supervised learning algorithm in machine learning, suitable for both classification and regression tasks. It operates on the principle of ensemble learning, combining multiple decision trees to enhance predictive accuracy.

By leveraging subsets of the dataset, Random Forest creates a multitude of decision trees. Each tree contributes its prediction, and the final output is determined by the majority vote of these predictions. This approach mitigates overfitting and improves accuracy, with the performance increasing as more trees are added to the forest.

The below diagram explains the working of the Random Forest algorithm:



Since the random forest combines multiple trees to predict the class of the dataset, it is possible that some decision trees may predict the correct output, while others may not. But together, all the trees predict the correct output. Therefore, below are two assumptions for a better Random forest classifier:

- There should be some actual values in the feature variable of the dataset so that the classifier can predict accurate results rather than a guessed result.

- The predictions from each tree must have very low correlations.

How Random Forest Works:

Random Forest algorithm works in two main phases:

1. **Random Forest Creation Phase**:

- Randomly select K data points from the training set.
- Build decision trees based on these selected data points (subsets).
- Repeat this process to create N decision trees (where N is predetermined).

2. **Prediction Phase:**

- For new data points, obtain predictions from each decision tree.
- Assign the new data points to the category that receives the majority of votes from all decision trees.

This process helps in creating an ensemble of decision trees and leveraging their collective wisdom to make accurate predictions.

Why we use Random Forest:

Below are some points that explain why we should use the Random Forest algorithm:

- It takes less training time as compared to other algorithms.
- It predicts output with high accuracy, even for the large dataset it runs efficiently.
- It can also maintain accuracy when a large proportion of data is missing.

6. Probabilistic Neural Network

The Probabilistic Neural Network (PNN) is a type of neural network primarily used for classification tasks. It employs a probabilistic approach to modelling data and making predictions. Below, I'll provide a mathematical explanation of how PNN works:

Probability Density Function (PDF): The core concept of PNN is based on probability density functions. Given a dataset with features x and corresponding labels y, the objective is to estimate the conditional probability P(y|x),
i.e., the probability of a certain label given the input features.

Radial Basis Function (RBF) Kernel: In PNN, the estimation of $P\binom{y}{x}$ is done using a radial basis function (RBF) kernel. The RBF kernel calculates the similarity between input data points.
The RBF kernel function $K(x, x')$ is defined as:

$$\boldsymbol{K(x, x') = e^{-\frac{||x-x'||^2}{\sigma^2}}}$$

Where ||x−x′||^2 is the squared Euclidean distance between the input vectors x and x′, and σ is a parameter controlling the width of the kernel.

Network Architecture:

- PNN consists of four layers: Input, Pattern, Summation, and Output.
- The Input layer represents the input features.
- The Pattern layer consists of neurons corresponding to each training example.
- The Summation layer computes the sum of the outputs of the Pattern layer neurons.
- The Output layer performs the final classification.

Training:

- During training, PNN stores the input feature vectors along with their corresponding labels in the Pattern layer.
- For each training example, the activation of the Pattern layer neuron is calculated using the RBF kernel with respect to the input data.
- The Summation layer aggregates the outputs of the Pattern layer neurons.

Prediction:

- To predict the label of a new input x, PNN calculates the output of each neuron in the Pattern layer using the RBF kernel.
- The Summation layer aggregates the outputs of the Pattern layer neurons.
- The Output layer applies a softmax function to compute the probability distribution over the classes.

Softmax Function:

The softmax function computes the probabilities of each class given the inputs. For a class j, the softmax function is defined as

$$P\left(y = \frac{j}{x}\right) = \frac{e^{a_j}}{\sum_{k=1}^{K} e^{a_k}}$$

Where a_j is the activation of the Output layer neuron corresponding to class j, and K is the total number of classes.

By utilizing these mathematical concepts, the Probabilistic Neural Network provides a probabilistic framework for classification tasks, allowing it to model complex decision boundaries and handle uncertainties in the data.

6. Results and Discussion

Based on the experimental setup described, a dataset of 100 samples has been collected, with each sample comprising 8 sensor readings. These samples are categorized into 5 distinct categories. These 5 categories are

1. Pure groundnut oil
2. Pure palm oil
3. 70% groundnut and 30% palm oil
4. 80% groundnut and 20% palm oil
5. 90% groundnut and 10% palm oil

And for each category, 10 data points have been collected for analysis, resulting in a final analytical dimension of 100x8.

For clustering analysis, K-means, PCA (Principal Component Analysis), and LDA (Linear Discriminant Analysis) techniques have been utilized. K-means clustering aims to partition the dataset into K clusters based on similarity. PCA is employed to reduce the dimensionality of the data while retaining as much variance as possible. LDA is used for dimensionality reduction and to find the linear combinations of features that best separate the categories.

For classification tasks, Linear Regression, Logistic Regression, Decision Tree, Random Forest, and Probabilistic Neural Network (PNN) algorithms have been applied. Linear Regression models the relationship between the independent variables (sensor readings) and the dependent variable (category) using a linear approach. Logistic Regression is suitable for binary classification problems. Decision Tree recursively splits the dataset into subsets based on the most significant features. Random Forest combines multiple decision trees to improve accuracy and robustness. PNN is a neural network model particularly useful for pattern classification tasks.

After analysing these entire algorithms now we have to get the results of all the algorithm to classify the given data and also put the accuracy of all these models.

Now I have to put the results of each model and analyse the accuracy and training and testing samples of each category so that we have to understand the each step of machine learning and fit the model properly.

a). Clustering Algorithms:

Clustering algorithms are used to classify data for several reasons:

Unsupervised Learning: Clustering is a form of unsupervised learning, meaning the algorithm doesn't require labeled data to learn patterns. This is particularly useful when labeled data is scarce or expensive to obtain.

Exploratory Data Analysis: Clustering helps in exploring the structure of the data by grouping similar data points together. It can reveal patterns, associations, or hidden structures in the data that might not be immediately apparent.

Data Compression and Summarization: By grouping similar data points together, clustering can reduce the complexity of the data. This can be useful for summarizing large datasets into a more manageable form, making it easier to analyze.

Customer Segmentation: In marketing and customer analytics, clustering is often used to segment customers into groups based on their similarities. This can help businesses in targeted marketing, product recommendations, and understanding customer behaviour.

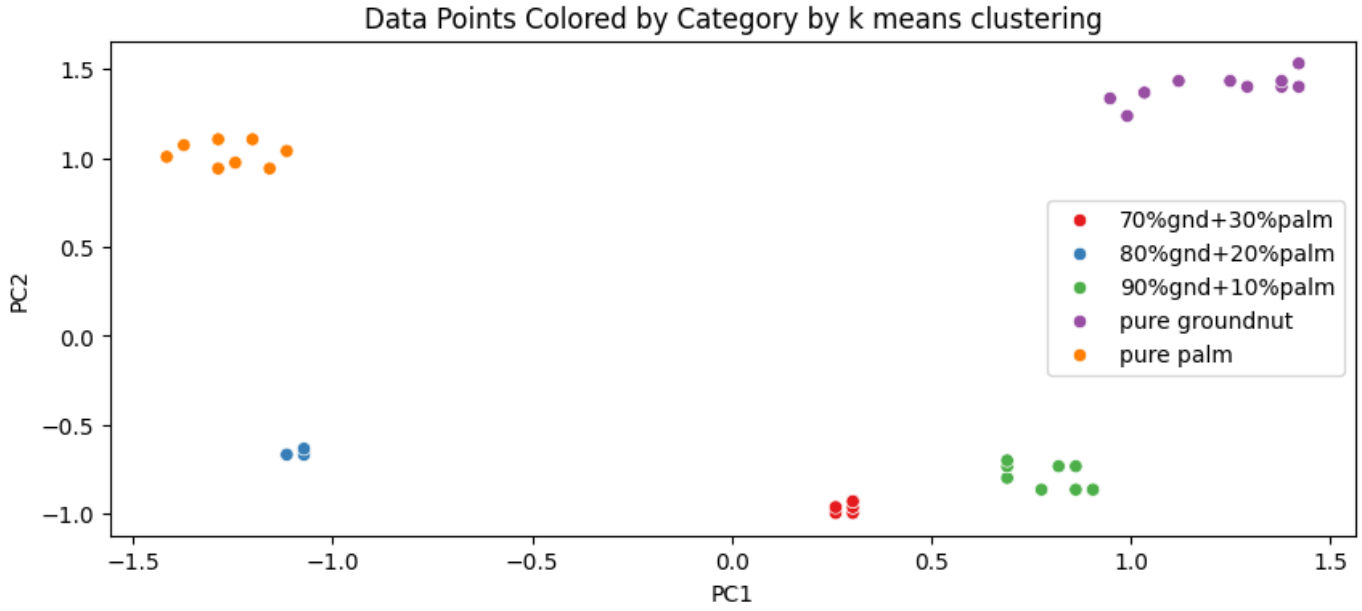
Image Segmentation: In image processing, clustering algorithms are used to segment images into regions with similar characteristics. This is useful in tasks such as object detection, image compression, and medical image analysis.

Overall, clustering algorithms provide valuable insights into the underlying structure of data and can be used in a wide range of applications across various domains.

In Our classification I have use k means clustering, DBSCAN Clustering, and Hierarchical Clustering. Meanwhile all the algorithms gave the same clusters. So I have put the summery of only k means clustering. Here I am putting the graphs of all the clustering.

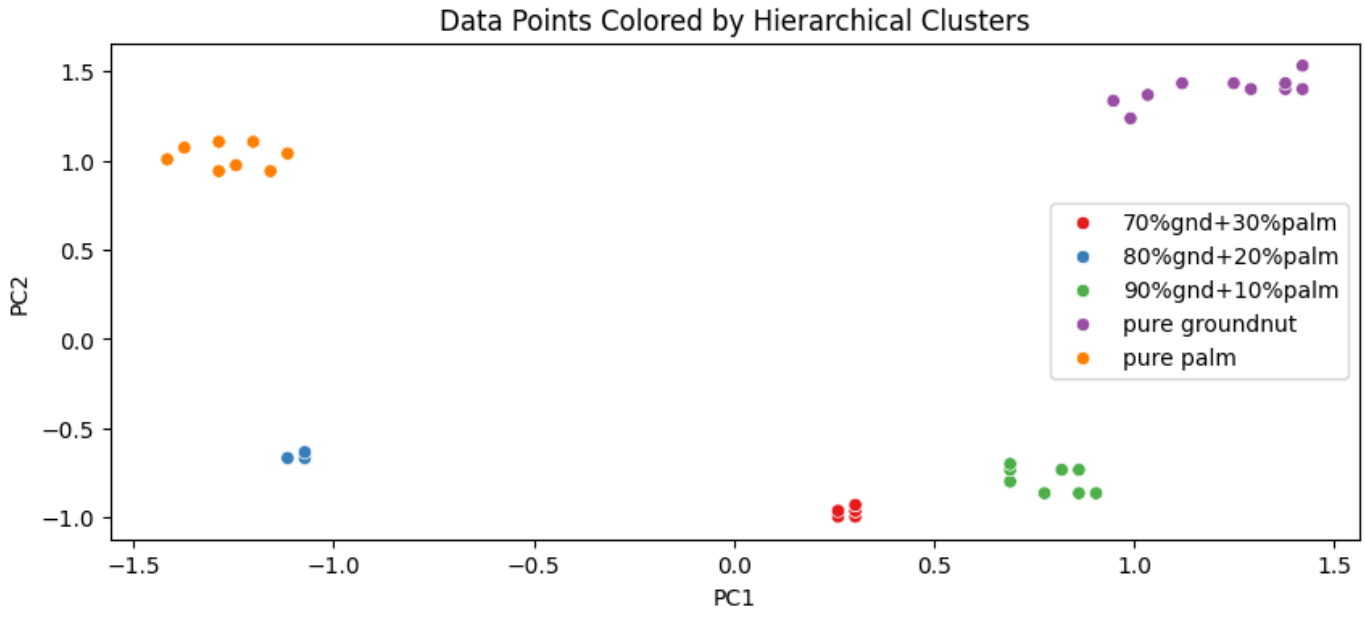
1. **K- Means Clustering:**

By using the given dataset of 5 categories the graph represented by K- Means Clustering is given below.



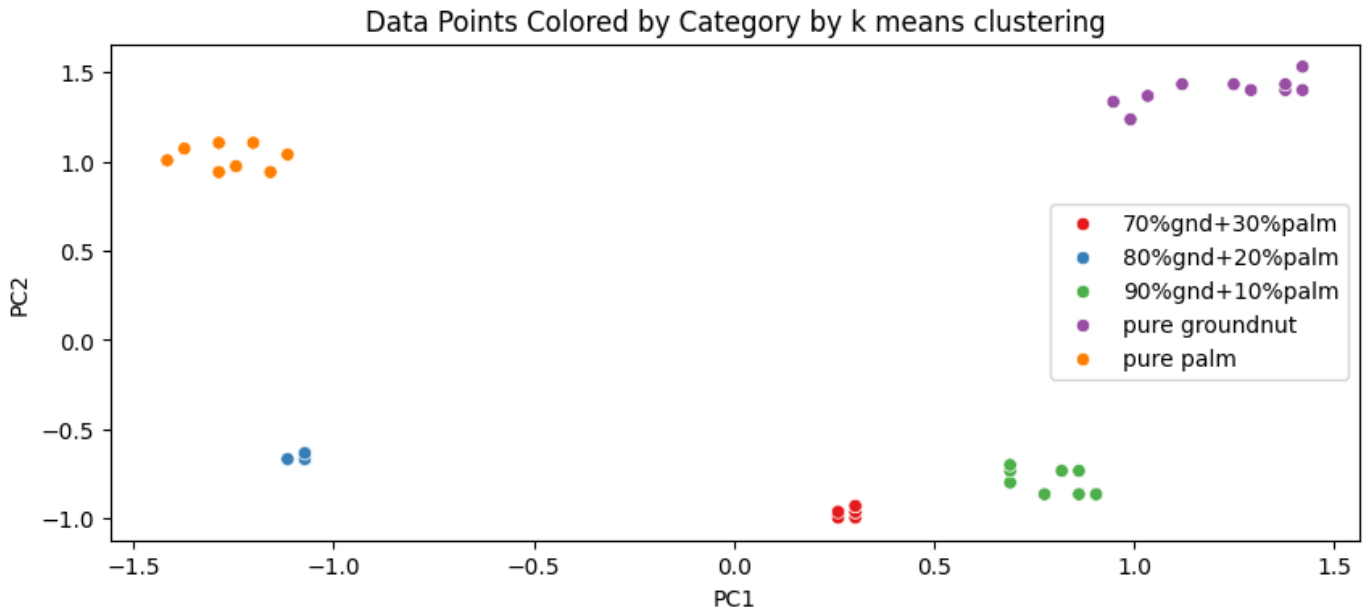
2. **DBSCAN Clustering:**

By using the given dataset of 5 categories the graph represented by DBSCAN Clustering is given below.



3. Hierarchical Clustering:

By using the given dataset of 5 categories the graph represented by Hierarchical Clustering is given below.



4. Principal Component Analysis

After using the clustering algorithms I have used Principal Component Analysis and the results by using PCA is having the crucial role in this work.

Since, Principal Component Analysis (PCA) is a dimensionality reduction technique used to simplify complex datasets by transforming variables into a new set of uncorrelated variables called principal components. These components retain the maximum variance from the original data, aiding in visualization, compression, and pattern recognition while minimizing information loss.

I am putting the explained variance ratio and Principal Components of PCA,

```
Explained Variance Ratio: [9.70966300e-01 2.66800792e-02 2.03426959e-03 1.46423269e-04
 8.71246228e-05 5.38505897e-05 1.79848008e-05 1.39675996e-05]
Principal Components: [[ 0.68838821  0.54315819  0.04729908  0.05528024  0.03551347  0.46980754
   0.06182278 -0.00231505]
 [ 0.45376299  0.08287473 -0.27591542  0.34552778 -0.0050474  -0.75571594
  -0.13055972  0.05944963]
 [ 0.39935449 -0.31829207 -0.16250299 -0.76363495  0.02381807 -0.14034345
   0.19977213 -0.26360595]
 [ 0.38364877 -0.68895427  0.23129635  0.18788759 -0.08780209  0.2400185
  -0.32449725  0.34458575]
 [-0.01228731 -0.02737567 -0.33119445  0.08195732 -0.92029825  0.1300396
   0.0941494  -0.09984761]
 [ 0.09831197 -0.24037218  0.29267446  0.42222595  0.05008352  0.01389348
   0.2522825  -0.77606381]
 [ 0.03622804 -0.16747275 -0.09235526  0.19289709  0.09943842  0.00815812
   0.86251953  0.41353007]
 [-0.04921117 -0.18859428 -0.79865553  0.19212892  0.36206571  0.33719319
  -0.13219008 -0.15805456]]
```

These Principal Components are known as the eigenvectors and Explained variance ratio having all the 8 eigenvalues.

The explained variance ratio in PCA indicates the proportion of variance in the original data that is accounted for by each principal component. It helps to assess the significance of each component in retaining information from the original dataset.

Principal components in PCA are new variables that are linear combinations of the original variables. They are ordered by the amount of variance they capture, with the first principal component capturing the most variance. These components serve to represent the data in a lower-dimensional space while retaining as much information as possible.

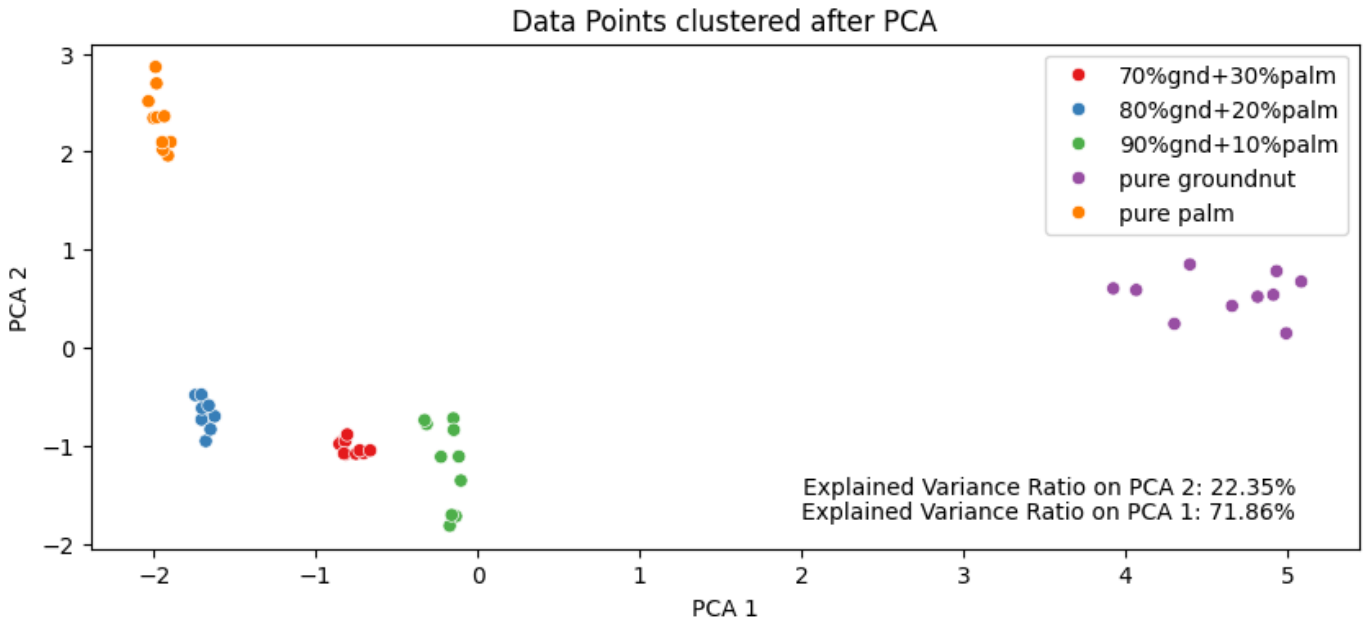
Now I am putting the PCA correlation matrix that is explained:

The PCA correlation matrix provides insight into the relationships between the original variables and the principal components. It shows the correlations between each original variable and each principal component. This information helps in understanding which original variables contribute the most to each principal component, aiding interpretation and feature selection in dimensionality reduction and data exploration tasks.

PCA Correlation Matrix:

	Sensor1_val	Sensor2_val	Sensor3_val	Sensor4_val	Sensor5_val	Sensor6_val	Sensor7_val	Sensor8_val
0	0.688388	0.543158	0.047299	0.055280	0.035513	0.469808	0.061823	-0.002315
1	0.453763	0.082875	-0.275915	0.345528	-0.005047	-0.755716	-0.130560	0.059450
2	0.399354	-0.318292	-0.162503	-0.763635	0.023818	-0.140343	0.199772	-0.263606
3	0.383649	-0.688954	0.231296	0.187888	-0.087802	0.240018	-0.324497	0.344586
4	-0.012287	-0.027376	-0.331194	0.081957	-0.920298	0.130040	0.094149	-0.099848
5	0.098312	-0.240372	0.292674	0.422226	0.050084	0.013893	0.252282	-0.776064
6	0.036228	-0.167473	-0.092355	0.192897	0.099438	0.008158	0.862520	0.413530
7	-0.049211	-0.188594	-0.798656	0.192129	0.362066	0.337193	-0.132190	-0.158055

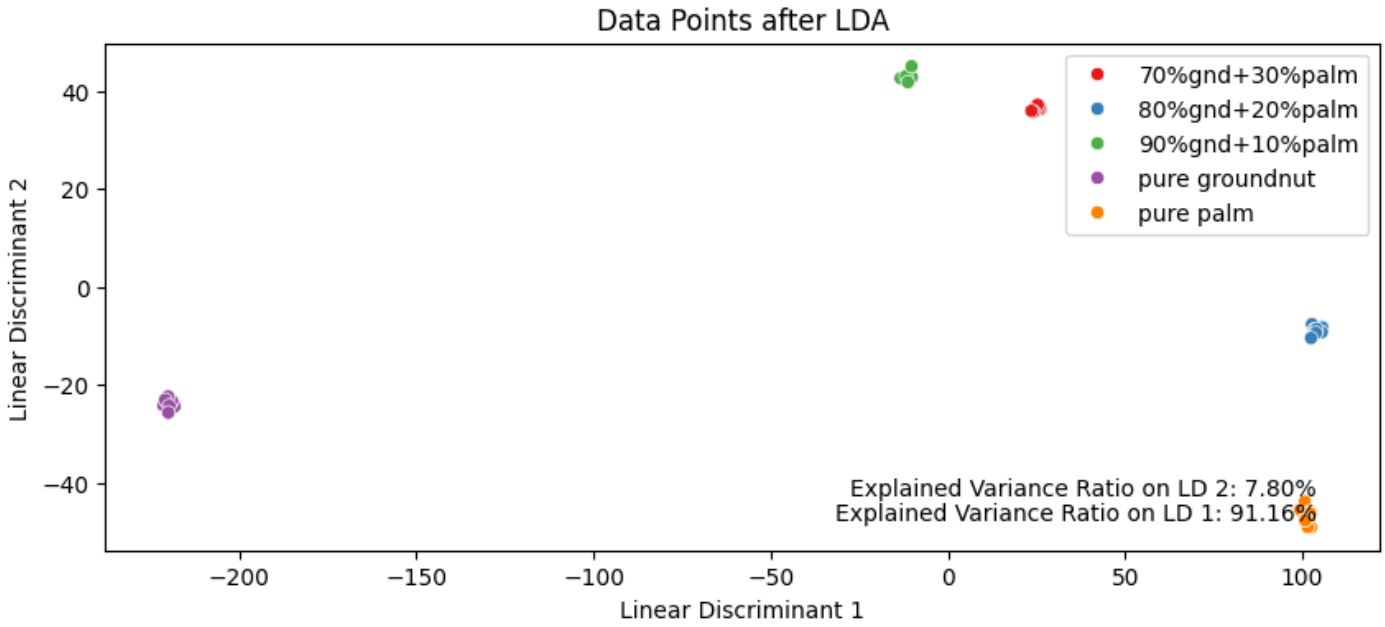
Now the time to put the graph of Data points clustered by PCA in which we see each category clustered in a similar way.



5. Linear Discriminant Analysis:

Linear Discriminant Analysis (LDA) is a dimensionality reduction technique and a classification algorithm. It finds linear combinations of features that best separate multiple classes while maximizing the distance between class means and minimizing within-class variance. It's commonly used in pattern recognition and machine learning for classification tasks.

The graph of LDA with explained variance ratio is given below:



By seeing the graph of LDA we can say that LDA is most preferable for this data rather than PCA. But PCA cannot ignore.

B). Machine Learning Algorithm

Now we have to analyse the working of machine learning models and get the accuracy of each model. After getting the accuracy of each model we will find the best fit model and worst fit model.

Machine learning models hold significant importance across various fields by analyzing data patterns to make predictions, optimize processes, and uncover insights. They enable advancements in healthcare, finance, transportation, and more, revolutionizing decision-making and automation. With their adaptability and scalability, machine learning models drive innovation and efficiency, shaping the future of technology.

Accuracy of a model:

The accuracy of a machine learning (ML) model measures its ability to make correct predictions or classifications. It represents the ratio of correct predictions to the total predictions made by the model. High accuracy indicates that the model performs well in capturing patterns and generalizing from the training data to unseen data.

Precision:

Precision in the context of machine learning refers to the fraction of relevant instances among the total instances predicted as positive by the model. It assesses the model's ability to correctly identify positive cases while minimizing false positives. A high precision indicates that when the model predicts a positive result, it is likely to be correct.

Recall:

Recall in the realm of machine learning denotes the fraction of relevant instances that the model correctly identifies among all relevant instances. It gauges the model's ability to capture all positive instances, minimizing false negatives. A high recall indicates that the model effectively identifies most positive cases, even if it leads to some false positives.

F-1 Score:

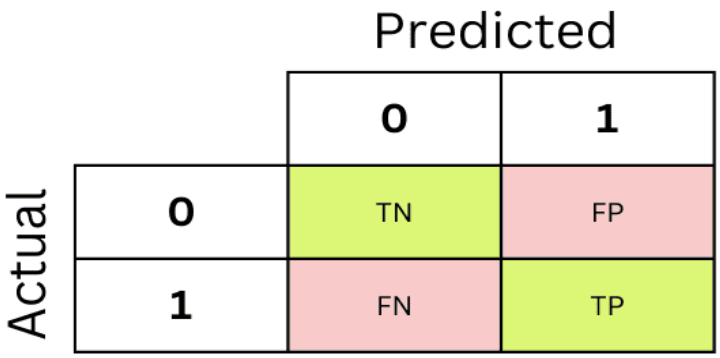
The F-1 score is a metric commonly used in classification tasks that combines both precision and recall into a single value. It's calculated as the harmonic mean of precision and recall:

$$f_1 - score = \frac{2 \times precision \times recall}{precision + recall}$$

It balances both precision and recall, providing a single score that represents the model's performance. In the context of a confusion matrix, the F1 score gives an overall assessment of how well the model is performing across both positive and negative classes, considering both false positives and false negatives.

Confusion Matrix:

A confusion matrix is a table used to summarize the performance of a classification model. It presents a comprehensive view of how well the model is performing by comparing actual and predicted classes. It contains information about true positives, true negatives, false positives, and false negatives, which are essential for evaluating the model's accuracy, precision, recall, and other performance metrics.



$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

The below tables give the accuracy, training and testing information of all machine learning models:

Accuracy Score Table for Original dataset

Algorithm	Accuracy	No. of samples for training	No. of samples for testing
Linear Regression	0.96	32	18
Logistic Regression	0.77	19	31
K-Nearest Neighbour	0.72	17	33
Decision Tree	0.96	23	27
Random Forest	0.72	12	38
Probability Neural Network	0.92	8	42

Accuracy Score Table with PCA

Algorithm	Accuracy	No. of samples for training	No. of samples for testing
Linear Regression	0.79	32	18
Logistic Regression	0.77	19	31
K-Nearest Neighbour	0.75	17	33
Decision Tree	0.96	23	27
Random Forest	0.97	12	38
Probability Neural Network	0.92	8	42

Accuracy Score Table with LDA

Algorithm	Accuracy	No. of samples for training	No. of samples for testing
Linear Regression	0.96	32	18
Logistic Regression	0.73	4	46
K-Nearest Neighbour	0.75	17	33
Decision Tree	0.95	10	40
Random Forest	0.78	4	46
Probability Neural Network	0.67	4	46

Model Summery Table of good results

Algorithm	Accuracy	Precision	Recall	F1 - Score
Decision Tree	0.96	0.97	0.97	0.97
Random Forest	0.97	0.97	0.98	0.97
Probability Neural Network	0.92	0.95	0.93	0.93

According to the given results table:

- **Accuracy:** Random Forest achieves the highest accuracy (0.97), followed closely by Decision Tree (0.96), while Probability Neural Network has a slightly lower accuracy (0.92).

- **Precision:** Precision measures the proportion of true positive predictions among all positive predictions made by the classifier. Both Decision Tree and Random Forest have the same precision (0.97), while Probability Neural Network has slightly lower precision (0.95).

- **Recall:** Recall measures the proportion of true positive instances that were correctly identified by the classifier. Random Forest has the highest recall (0.98), followed by Decision Tree (0.97), and Probability Neural Network has the lowest recall (0.93).

- **F1-Score:** F1-score is the harmonic mean of precision and recall, providing a balanced measure of a classifier's performance. Decision Tree and Random Forest achieve the same F1-score (0.97), while Probability Neural Network has a slightly lower F1-score (0.93).

Overall, Random Forest appears to perform the best across all metrics, followed closely by Decision Tree, while Probability Neural Network lags slightly behind in terms of accuracy, recall, and F1-score. However, the choice of the best algorithm depends on various factors such as dataset characteristics, computational resources, interpretability, and specific requirements of the problem at hand.

7. Conclusions

Now we'll put the conclusions of whole the study:

- Development of a cost-effective and efficient method for assessing groundnut oil quality through multi-sensor signal analysis, offering a viable alternative to traditional, expensive techniques like GC-MS and HPLC.

- Customized multi-sensor instrument utilizing 8 MOS sensors enables detection of common chemical components in groundnut oil, facilitating accurate prediction of fatty acid content through signature pattern analysis.

- Application of Multivariate Data Analysis, particularly probability neural network, ensures precise processing of multi-sensor data, enhancing the reliability of fatty acid content prediction.

- Comprehensive analysis of blended palm oil and groundnut oil products utilizing clustering and classification techniques, revealing compositional patterns and quality characteristics within the blends.

- Identification of natural product groupings based on compositional similarities through clustering algorithms, providing valuable insights into the categorization of blended oil products.

- Evaluation of classification models to predict blend categories and assess algorithm performances, enabling stakeholders to make informed decisions regarding product classification and quality control in the food industry.

- Also in conclusion, by using accuracy of all ML models, Random Forest emerges as the top-performing algorithm based on the provided metrics, followed closely by Decision Tree, while Probability Neural Network exhibits slightly lower performance. However, the final choice of algorithm should consider various factors, including performance, computational resources, interpretability, and specific requirements of the classification task.

Overall, the study presents a significant contribution to the oil industry by offering a practical solution for routine analysis of groundnut oil quality and providing insights into the characterization of blended oil products, thereby facilitating improved decision-making processes for stakeholders.

8. Applications and Future Scope

- ## Applications:

1. **Quality Control in the Oil Industry:** The research provides a practical solution for assessing the quality of groundnut oil, a crucial parameter in the oil industry. The developed multi-sensor instrument can be implemented in oil production facilities for routine analysis, enabling timely quality control measures to maintain product standards.

2. **Cost-Effective Analysis in Continuous Production Lines:** By replacing costly and complex methods like GC-MS and HPLC with a customized multi-sensor instrument, the research offers a cost-effective alternative for routine analysis in continuous production lines. This can streamline production processes and reduce operational costs for oil manufacturers.

3. **Enhanced Product Classification and Quality Control**: The analysis of blended palm oil and groundnut oil products using clustering and classification techniques contributes to enhanced product classification and quality control within the food industry. Stakeholders can utilize the findings to ensure product consistency and meet regulatory standards.

4. **Food Safety and Consumer Protection:** Implementing the developed multi-sensor instrument and analysis techniques can enhance food safety and consumer protection by ensuring the quality and authenticity of groundnut oil products. This is particularly important in regions where adulteration and food fraud are prevalent concerns.

5. **Supply Chain Management**: The research findings can be utilized in supply chain management to track and monitor the quality of groundnut oil from production to consumption. This can help stakeholders identify and address quality issues early in the supply chain, improving overall product integrity and customer satisfaction.

6. **Nutritional Labeling and Claims Verification**: Accurate prediction of fatty acid content allows for precise nutritional labeling of groundnut oil products, helping consumers make informed dietary choices. Additionally, the analysis techniques can aid in verifying nutritional claims made by manufacturers, ensuring compliance with regulatory standards.

- # Future Scope:

1. **Expansion to Other Oil Varieties**: The research can be extended to assess the quality of other types of oils beyond groundnut oil. By adapting the multi-sensor instrument and analysis techniques, the method can be applied to assess the quality of various edible oils, providing a versatile solution for the oil industry.

2. **Integration with Industry 4.0 Technologies:** Future research can explore the integration of the developed multi-sensor instrument with Industry 4.0 technologies such as Internet of Things (IoT) and Artificial Intelligence (AI). This integration can enable real-time monitoring and predictive maintenance in oil production facilities, optimizing production processes and minimizing downtime.

3. **Exploration of Sensor Fusion Techniques:** Further research can investigate sensor fusion techniques to enhance the accuracy and reliability of fatty acid content prediction. By combining data from multiple sensor types, such as MOS sensors and optical sensors, researchers can improve the sensitivity and specificity of the analysis method.

4. **Application in Food Authentication:** The clustering and classification techniques employed in the analysis of blended palm oil and groundnut oil products can be extended to authenticate and classify other food products. This could include detecting adulteration or ensuring the authenticity of food products in the supply chain.

Overall, the research lays the foundation for innovative applications in quality control, production optimization, and food authentication within the oil and food industries, with potential for further advancements and integration with emerging technologies.

9. References

1) Oil Fatty Acid Measurement (OFAM) system for Blended Groundnut Oil Arun Jana, Devdulal Ghosh, Subhankar Mukherjee, Alokesh Ghosh, Amitava Akuli, Nabarun Bhattacharyya, Centre for Development of Advanced Computing (C-DAC), Kolkata, India

2) Jonnala, R. S. ; Dunford, N. T. ; Dashiell, K. E., 2005. New high oleic peanut cultivars grown in the Southwestern United States. J. Am. Oil Chem. Soc., 82(2): 125–128

3) Andersen, P.C., Hill, K., Gorbet, D.W., Brodbeck, B.V. 1998. Fatty acid and amino acid profiles of selected peanut cultivars and breeding lines. J. Food Comp. Anal. 11:100-111

4) Kirk, R. S. and Sawyer, R. (1991). Pearson's Composition and Analysis of Foods, 9th ed. (student edition), England: Addision Wesley Longman Ltd. 33-36.

5) Esuoso, K.O., & Odetokun, S.M. (1995). Proximate chemical composition and possible industrial utilization of Blighia sapida seed and seed oils.

6) Fang, G., Goh, J. Y., Tay, M., Lau, H. F., & Li, S. F. (2013). Characterization of oils and fats by 1H NMR and GC/MS fingerprinting: classification, prediction and detection of adulteration. Food chemistry, 138(2-3), 1461–1469. https://doi.org/10.1016/j.foodchem.2012.09.136

7) HAJIMAHMOODI, M., VANDERHEYDEN, Y., SADEGHI, N., JANNAT, B., OVEISI, M., & SHAHBAZIAN, S. (2005). Gas-chromatographic fatty-acid fingerprints and partial least squares modeling as a basis for the simultaneous determination of edible oil mixtures. Talanta, 66(5), 1108–1116. doi:10.1016/j.talanta.2005.01.011

8) Hilali, M., Charrouf, Z., Soulhi, A. E. A., Hachimi, L., & Guillaume, D. (2007). Detection of Argan Oil Adulteration Using Quantitative Campesterol GC-Analysis. Journal of the American Oil Chemists' Society, 84(8), 761–764. doi:10.1007/s11746-007-1084-y

9) Cunha, S.C., & Oliveira, M.B. (2006). Discrimination of vegetable oils by triacylglycerols evaluation of profile using HPLC/ELSD. Food Chemistry, 95, 518-524.

10) Marikkar, J. M. N., Ghazali, H. M., Che Man, Y. B., Peiris, T. S. G., & Lai, O. M. (2005). Distinguishing lard from other animal fats in admixtures of some vegetable oils using liquid chromatographic data coupled with multivariate data analysis. Food Chemistry, 91(1), 5–14. doi:10.1016/j.foodchem.2004.01.08

11) Aparicio, R., & Aparicio-Ruíz, R. (2000). Authentication of vegetable oils by chromatographic techniques. Journal of chromatography. A, 881(1-2), 93–104. https://doi.org/10.1016/s0021-9673(00)00355-1

12) Tilak T. Chandratilleke, Nima Nadim, Ramesh Narayanaswamy, Vortex structure-based analysis of laminar flow behaviour and thermal characteristics in curved ducts, International Journal of Thermal Sciences, Volume 59, 2012, Pages 75-86, ISSN 1290-0729, https://doi.org/10.1016/j.ijthermalsci.2012.04.014.

13) Xie, J., Liu, T., Yu, Y., Song, G., & Hu, Y. (2013). Rapid Detection and Quantification by GC–MS of Camellia Seed Oil Adulterated with Soybean Oil. Journal of the American Oil Chemists' Society, 90(5), 641–646. doi:10.1007/s11746-013- 2209-0

14) Suman, M., Riani, G., & Dalcanale, E. (2007). MOS-based artificial olfactory system for the assessment of egg products freshness. Sensors and Actuators B: Chemical, 125(1), 40–47. doi:10.1016/j.snb.2007.01.031

15) Zheng, X. Z., Lan, Y. B., Zhu, J. M., Westbrook, J., Hoffmann, W. C., & Lacey, R. E. (2009). Rapid identification of rice samples using an electronic nose. Journal of Bionic Engineering, 6(3), 290-297. 1

16) Lippolis, V., Pascale, M., Cervellieri, S., Damascelli, A., & Visconti, A. (2014). Screening of deoxynivalenol contamination in durum wheat by MOS-based electronic nose and identification of the relevant pattern of volatile compounds. Food Control, 37, 263-271.

17) Kang, G., Cho, S., Seong, P., Park, B., Kim, S., Kim, D., ... & Park, K. (2013). Effects of high pressure processing on fatty acid composition and volatile compounds in Korean native black goat meat. Meat science, 94(4), 495-499.

18) Capone, S., Tufariello, M., & Siciliano, P. (2013). Analytical characterisation of Negroamaro red wines by "Aroma Wheels". Food chemistry, 141(3), 2906-2915.

19) Defilippi, B. G., Juan, W. S., Valdés, H., Moya-León, M. A., Infante, R., & Campos-Vargas, R. (2009). The aroma development during storage of Castlebrite apricots as evaluated by gas chromatography, electronic nose, and sensory analysis. Postharvest Biology and Technology, 51(2), 212–219. doi:10.1016/j.postharvbio.2008.08.008

20) Giovanni Pacioni, Lorenzo Cerretani, Giuseppe Procida, Angelo Cichelli, Composition of commercial truffle flavored oils with GC–MS analysis and discrimination with an electronic nose, Food Chemistry, Volume 146, 2014, Pages 30-35, ISSN 0308-8146, https://doi.org/10.1016/j.foodchem.2013.09.016.

21) K. Anjani, Praduman Yadav (2017), "High yielding-high oleic non-genetically modified Indian safflower cultivars", Industrial Crops & Products, vol. 104, pp. 7 – 12

22) Praduman Yadav, · K. Anjani (2017), "Assessment of Variation in Castor Genetic Resources for Oil Characteristics", J Am Oil Chem Soc, DOI 10.1007/s11746-017-2961-7.