Food Classification Using Nutritional Data (Machine Learning)

Name: - Rahul Raj

Batch: - DS-C-WE-E-B89

1) Introduction

In the modern era of digital health and smart nutrition, dietary awareness has become increasingly important due to the rise in lifestyle-related diseases such as obesity, diabetes, and heart disorders. With the rapid growth of food databases and nutritional information, there is a growing demand for intelligent systems capable of analysing food data and automatically classifying food items based on their nutritional composition. Machine Learning (ML) plays a vital role in building such intelligent classification systems by learning complex patterns from data and making accurate predictions. Food classification using nutritional attributes is a complex task because multiple food items may share similar nutritional values while belonging to different categories. Traditional classification methods based on fixed rules often lack flexibility and fail to generalize well across diverse food types. To address this challenge, this project proposes a multi-class machine learning approach that classifies food items into seven categories—Banana, Burger, Donut, Ice Cream, Pasta, Pizza, and Sushi—using nutritional features such as calories, protein, fat, carbohydrates, sugar, fibre, sodium, cholesterol, glycaemic index, water content, and serving size. This project evaluates and compares the performance of seven widely used machine learning classification algorithms: **Logistic Regression**, **Decision Tree**, **Random Forest**, **K-Nearest Neighbours (KNN)**, **Support Vector Machine (SVM)**, **Gradient Boosting Classifier**, and **XGBoost**. With the help of these model we have got almost + 90% accuracy in each model. Logistic Regression provides a strong baseline model for multi-class classification, while Decision Trees offer interpretability by learning hierarchical decision rules. Random Forest improves robustness by combining multiple decision trees, whereas KNN classifies food items based on similarity in nutritional space. Support Vector Machine is employed for its ability to handle high-dimensional data using optimal decision boundaries. Gradient Boosting and XGBoost are advanced ensemble methods that iteratively improve model performance by focusing on misclassified samples. The project involves comprehensive data preprocessing steps including handling missing values, removing duplicate records, outlier treatment, encoding categorical variables, and feature scaling. Exploratory Data Analysis (EDA) is conducted to understand data distributions, skewness, and inter-feature relationships. Model performance is evaluated using accuracy, precision, recall, F1-score, and confusion matrices to ensure reliable comparison. The outcomes of this project demonstrate how different machine learning models behave on nutritional data and provide insights into selecting the most effective classification technique. Such a system can be applied in smart dietary applications, food logging systems, health monitoring platforms, and educational tools to promote informed food choices and nutritional awareness.

2) Objective

- **Exploratory Data Analysis (EDA)**

  Exploratory Data Analysis (EDA) plays a crucial role in understanding and preparing the nutritional dataset for machine learning modelling in the NutriClass project. The initial step in EDA involved identifying and handling missing values present in the dataset. Since incomplete nutritional information can lead to biased results or errors during model training, rows containing null values were removed. Given the sufficiently large size of the dataset, this approach ensured data consistency without significantly affecting the overall data distribution. Duplicate records were also identified and eliminated to prevent repeated food entries from influencing the learning process, as duplicates can lead to overfitting and distorted evaluation metrics. After cleaning the dataset, nutritional attributes such as calories, protein, fat, carbohydrates, sugar, sodium, and cholesterol were analysed across different food categories using box plots. Box plots were chosen because they effectively display the median, interquartile range, and presence of outliers, allowing clear comparison of nutritional variation among food items. The analysis revealed noticeable differences in nutritional profiles, with processed foods like burgers and pizza showing higher calorie, fat, and sodium levels, while foods such as bananas demonstrated lower and more stable nutritional values. These visual insights helped in identifying outliers, understanding data spread, and confirming that nutritional features provide meaningful distinctions between food categories. Overall, the EDA process ensured that the dataset was clean, reliable, and well-understood, forming a strong foundation for effective pre-processing and accurate machine learning classification.

- **Scaling and Encoding**

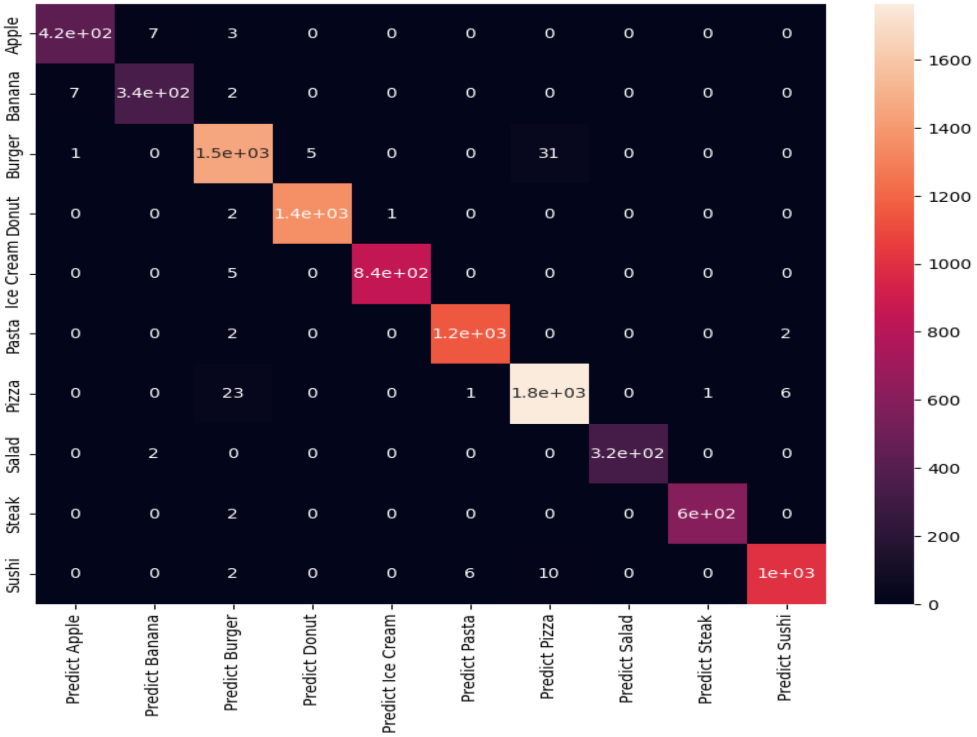
  In the pre-processing stage of this project, feature scaling and data encoding were performed to ensure that the dataset was compatible with machine learning algorithms and to improve overall model performance. Scaling was applied to numerical nutritional features such as calories, protein, fat, carbohydrates, sugar, sodium, and cholesterol because these attributes exist on very different value ranges. For example, sodium values can be in the hundreds while protein values are often in single digits. Without scaling, features with larger magnitudes may dominate distance-based and optimization-based models such as K-Nearest Neighbours (KNN), Support Vector Machine (SVM), and Logistic Regression, leading to biased predictions. Additionally, scaling helps reduce the impact of outliers by bringing all numerical features to a common scale, allowing the model to learn balanced patterns from the data. Standardization techniques were used to transform the features so that they have a mean of zero and a standard deviation of one.

  Encoding was applied to categorical variables because machine learning models require numerical inputs to perform computations. Categorical features such as **Food Name** cannot be processed directly in textual form. Therefore, encoding techniques such as one-hot encoding were used to convert categorical values into numerical representations without introducing ordinal relationships. This ensured that the models could correctly interpret categorical information while maintaining data integrity. Overall, scaling and encoding were essential pre-processing steps that enabled efficient model training and improved classification accuracy.

- **Machine Learning Classification Model**

Logistic Regression

Logistic Regression is a widely used classification algorithm that models the probability of a data point belonging to a particular class using a logistic function. Although originally designed for binary classification, it can be extended to multi-class problems using techniques such as one-vs-rest. In this project, Logistic Regression serves as a baseline model to evaluate how well simple linear decision boundaries can separate food categories based on nutritional attributes. The model assumes a linear relationship between the input features and the log-odds of the output classes. Nutritional features such as calories, fat, sugar, and sodium contribute to estimating the probability of a food item belonging to a specific category. Logistic Regression is computationally efficient and interpretable, allowing insights into feature importance through model coefficients. However, it may struggle with complex, non-ear relationships present ideal-world food data. Despite this limitation, Logistic Regression provides a strong benchmark to compare more advanced machine learning models used in this project. This model predict accuracy 98.7%.

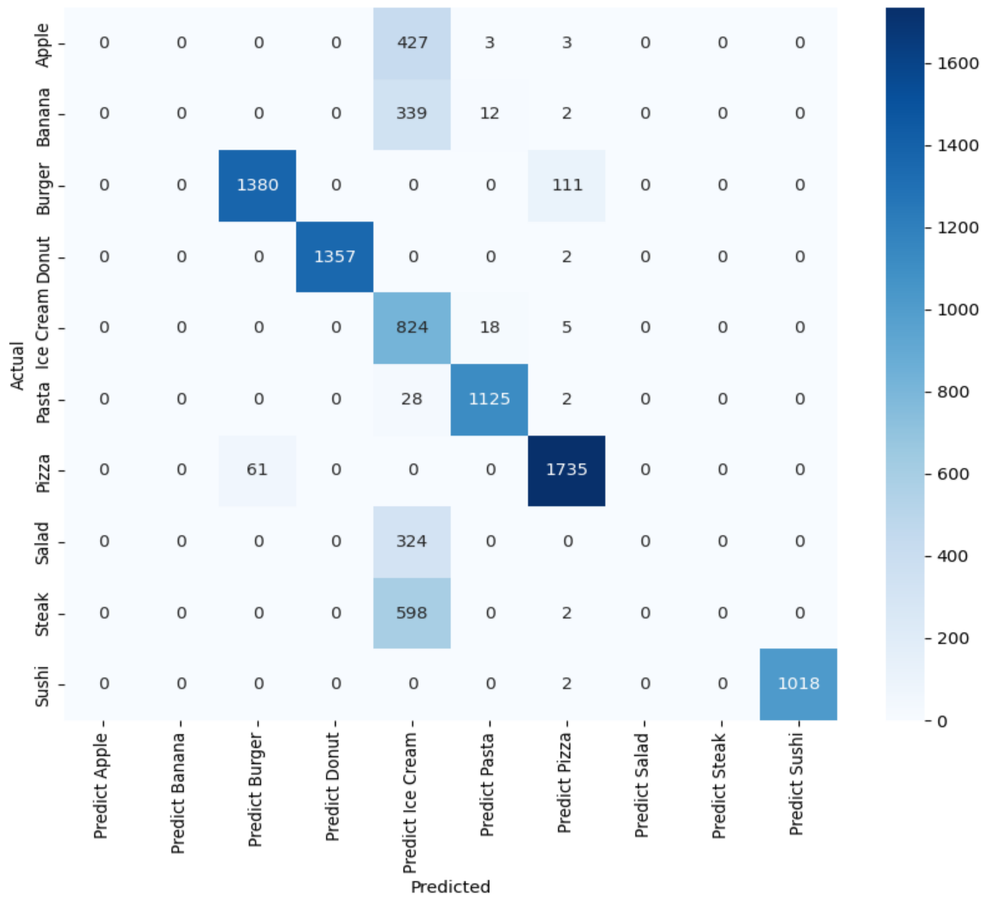


	Predict Apple	Predict Banana	Predict Burger	Predict Donut	Predict Ice Cream	Predict Pasta	Predict Pizza	Predict Salad	Predict Steak	Predict Sushi
Apple	4.2e+02	7	3	0	0	0	0	0	0	0
Banana	7	3.4e+02	2	0	0	0	0	0	0	0
Burger	1	0	1.5e+03	5	0	0	31	0	0	0
Donut	0	0	2	1.4e+03	1	0	0	0	0	0
Ice Cream	0	0	5	0	8.4e+02	0	0	0	0	0
Pasta	0	0	2	0	0	1.2e+03	0	0	0	2
Pizza	0	0	23	0	0	1	1.8e+03	0	1	6
Salad	0	2	0	0	0	0	0	3.2e+02	0	0
Steak	0	0	2	0	0	0	0	0	6e+02	0
Sushi	0	0	2	0	0	6	10	0	0	1e+03

Fig1: Confusion metrics of Logistic

Decision Tree Classifier

The Decision Tree classifier is a non-linear, rule-based model that splits data into subsets based on feature conditions. It constructs a tree structure where each internal node represents a decision on a feature, and each leaf node represents a class label. In this project, the Decision Tree model helps capture non-linear relationships between nutritional features and food categories. For example, foods with high fat and sodium values may follow a different decision path than low-calorie foods. Decision Trees are easy to interpret and visualize, making them suitable for understanding classification logic. However, they are prone to overfitting, especially when the tree grows too deep. To address this, parameters such as maximum depth and minimum samples per split can be controlled. Despite their limitations, Decision Trees are effective for learning complex patterns and provide a foundation for ensemble models like Random Forest and Gradient Boosting. This model predict accuracy 98.8%.



	Predict Apple	Predict Banana	Predict Burger	Predict Donut	Predict Ice Cream	Predict Pasta	Predict Pizza	Predict Salad	Predict Steak	Predict Sushi
Apple	0	0	0	0	427	3	3	0	0	0
Banana	0	0	0	0	339	12	2	0	0	0
Burger	0	0	1380	0	0	0	111	0	0	0
Donut	0	0	0	1357	0	0	2	0	0	0
Ice Cream	0	0	0	0	824	18	5	0	0	0
Pasta	0	0	0	0	28	1125	2	0	0	0
Pizza	0	0	61	0	0	0	1735	0	0	0
Salad	0	0	0	0	324	0	0	0	0	0
Steak	0	0	0	0	598	0	2	0	0	0
Sushi	0	0	0	0	0	0	2	0	0	1018

Fig2: Confusion metrics of Decision Tree Model

Random Forest Classifier

Random Forest is an ensemble learning technique that combines multiple decision trees to improve classification accuracy and reduce overfitting. Each tree in the forest is trained on a random subset of data and features, and the final prediction is obtained through majority voting. In this project, Random Forest effectively captures complex interactions among nutritional features while maintaining robustness against noise and outliers. By averaging the predictions of multiple trees, Random Forest reduces variance and improves generalization compared to a single Decision Tree. It also provides feature importance scores, which help identify influential nutritional attributes such as calories, fat, and sodium. Although Random Forest models are computationally heavier, they perform well on structured tabular data like nutritional datasets. This model significantly improves classification stability and accuracy compared to simpler classifiers. This model predict accuracy 99.3%.

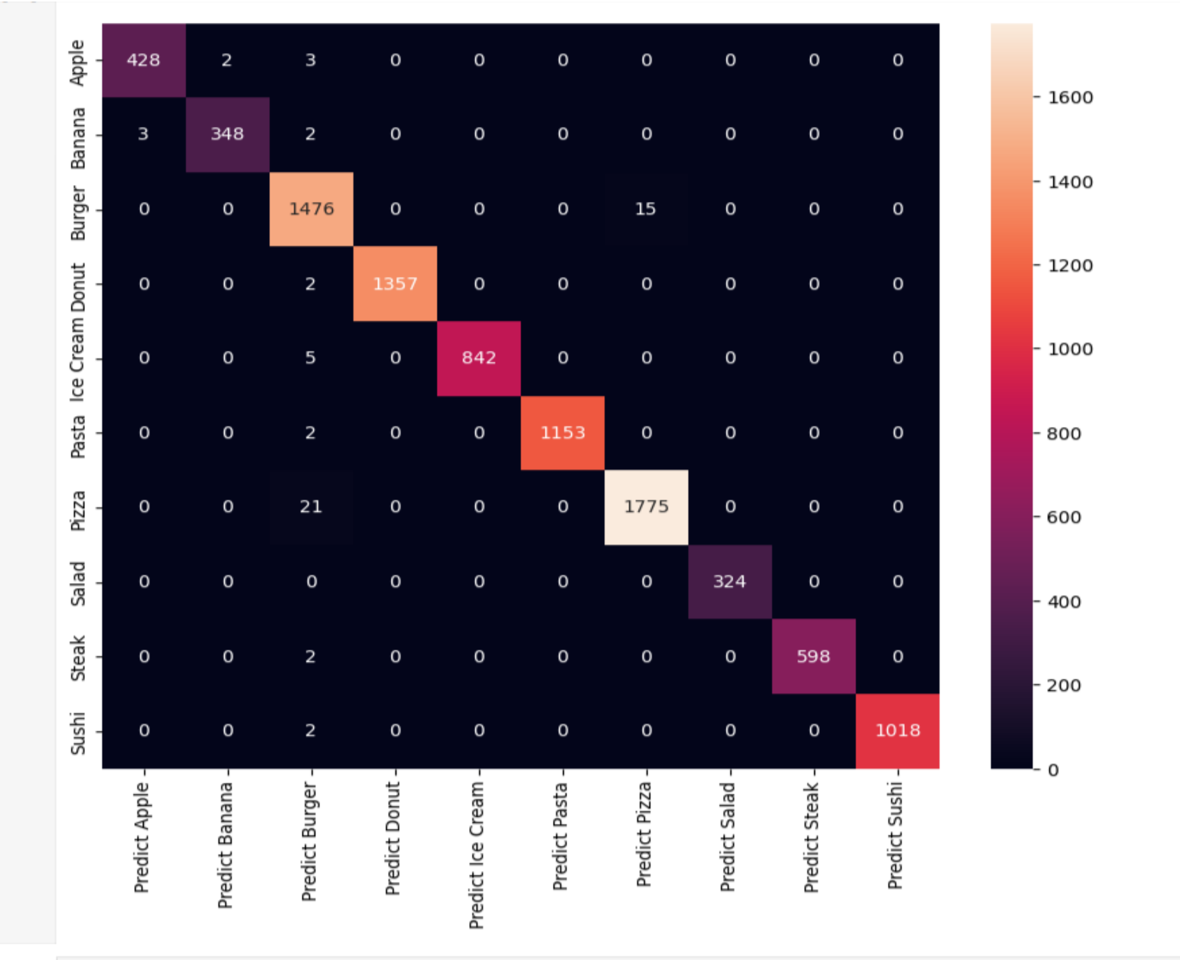


Fig3: Confusion metrics of Random Forest Classifier

K-Nearest Neighbours (KNN)

K-Nearest Neighbours is a distance-based classification algorithm that assigns a class label to a data point based on the majority class among its nearest neighbours. In this project, KNN classifies food items by comparing their nutritional profiles to similar food records. Distance metrics such as Euclidean distance are used to measure similarity, making feature scaling an essential pre-processing step. KNN is simple and intuitive but computationally expensive for large datasets since it stores all training data. Its performance depends heavily on the choice of the number of neighbours (k) and distance weighting. While KNN can capture local patterns in nutritional data, it is sensitive to noise and irrelevant features. Despite these limitations, KNN provides valuable insights into how similarity-based learning performs on food classification tasks. This model predict accuracy 99.1%.

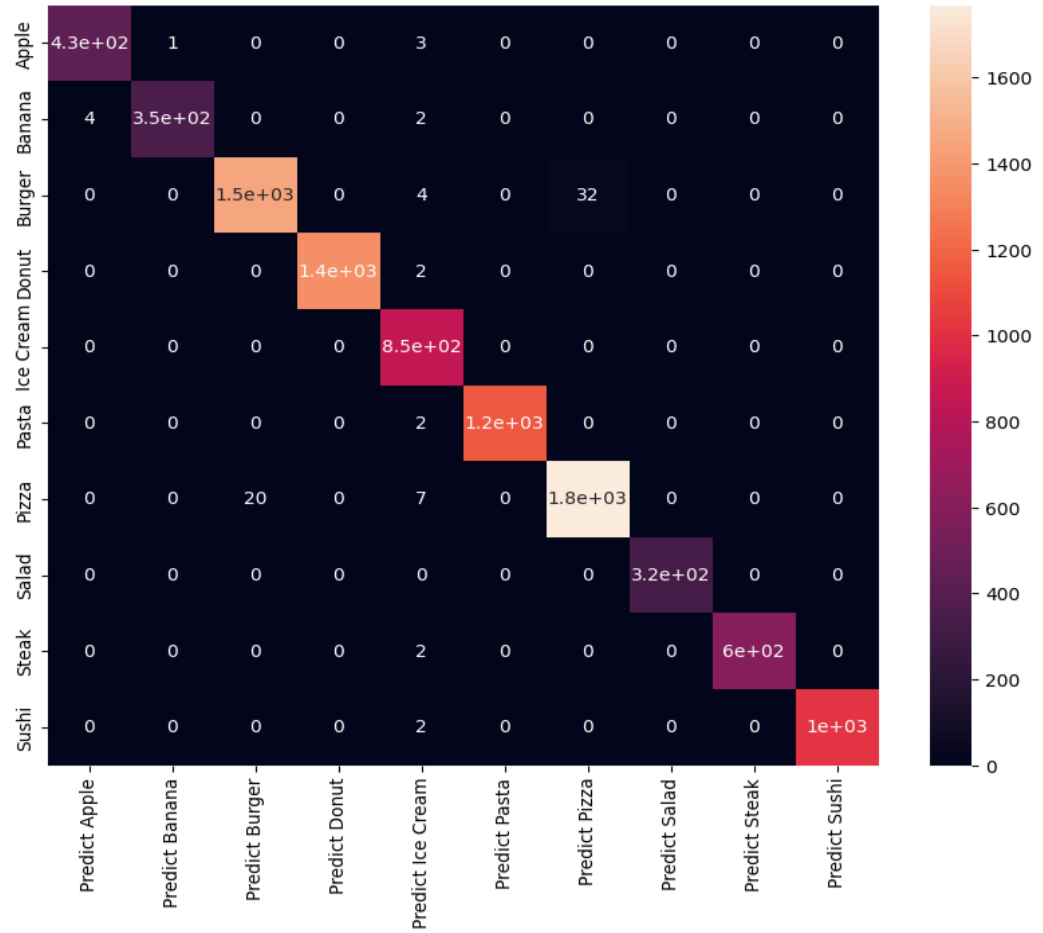


Fig4: Confusion metrics of K-Nearest Neighbours

Support Vector Machine (SVM)

Support Vector Machine is a powerful classification algorithm that finds an optimal hyperplane to separate data points into different classes. It works well in high-dimensional spaces and can handle non-linear boundaries using kernel functions. In this project, SVM is used to classify food items by maximizing the margin between different food categories based on nutritional attributes. SVM is particularly effective when classes are well-separated, but it requires careful tuning of parameters such as kernel type and regularization strength. Feature scaling is essential for SVM to perform optimally. While SVM offers strong theoretical guarantees, it can be computationally expensive for large datasets. Nonetheless, it is a valuable model for capturing complex decision boundaries in nutritional data. This model predict accuracy 95.3%.

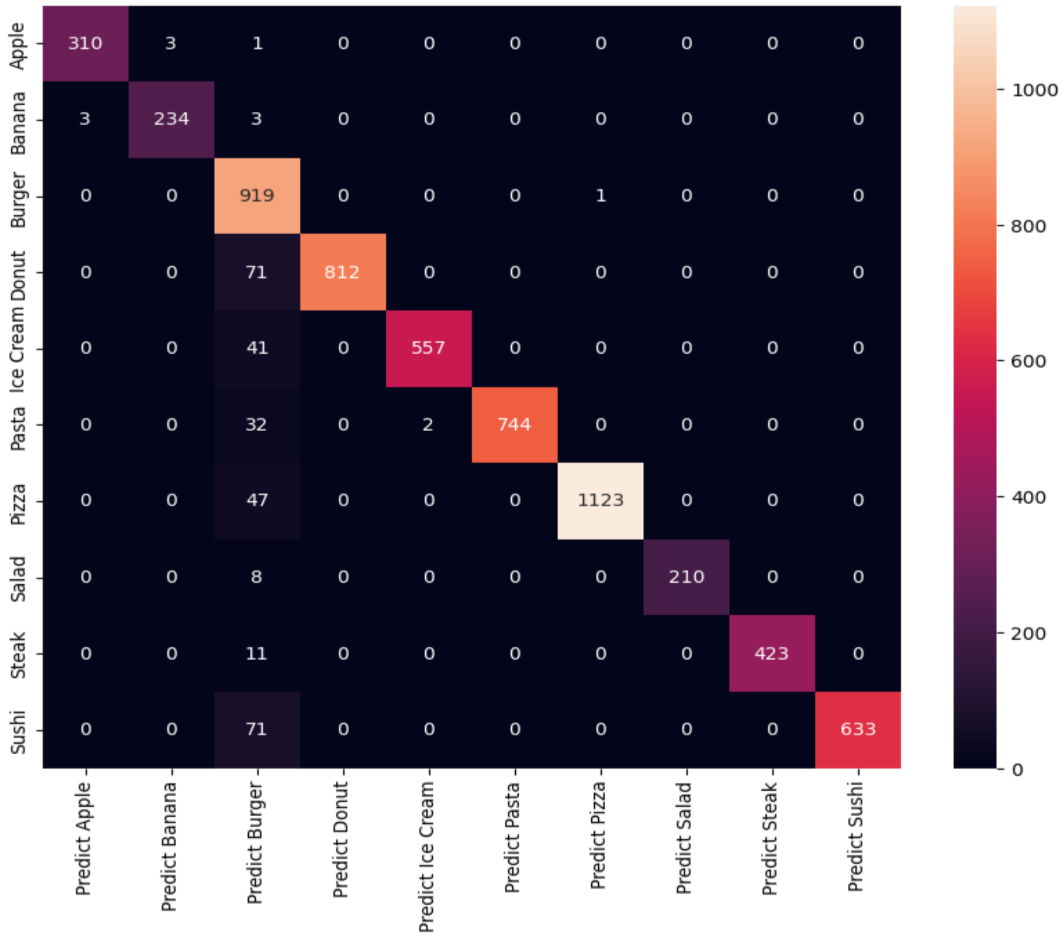
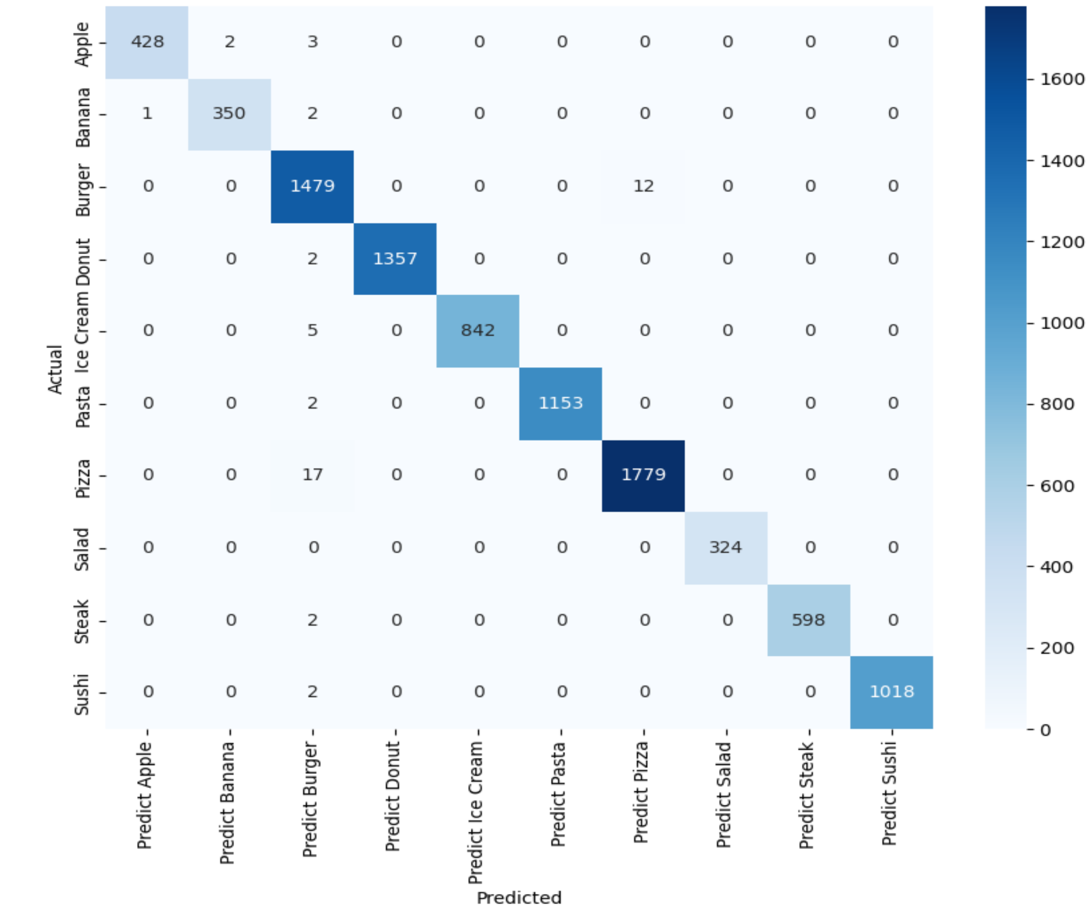


Fig5: Confusion metrics of Support Vector Machine

Gradient Boosting Classifier

Gradient Boosting is an ensemble technique that builds models sequentially, where each new model corrects the errors of the previous one. In this project, Gradient Boosting effectively learns complex patterns by focusing on misclassified food items during training. The model combines multiple weak learners, typically shallow decision trees, to create a strong classifier. Gradient Boosting handles mixed feature types well and provides high predictive accuracy. However, it requires careful parameter tuning to avoid overfitting. This model is particularly useful for structured datasets and demonstrates strong performance in multi-class food classification tasks. This model predict accuracy 95.3%.



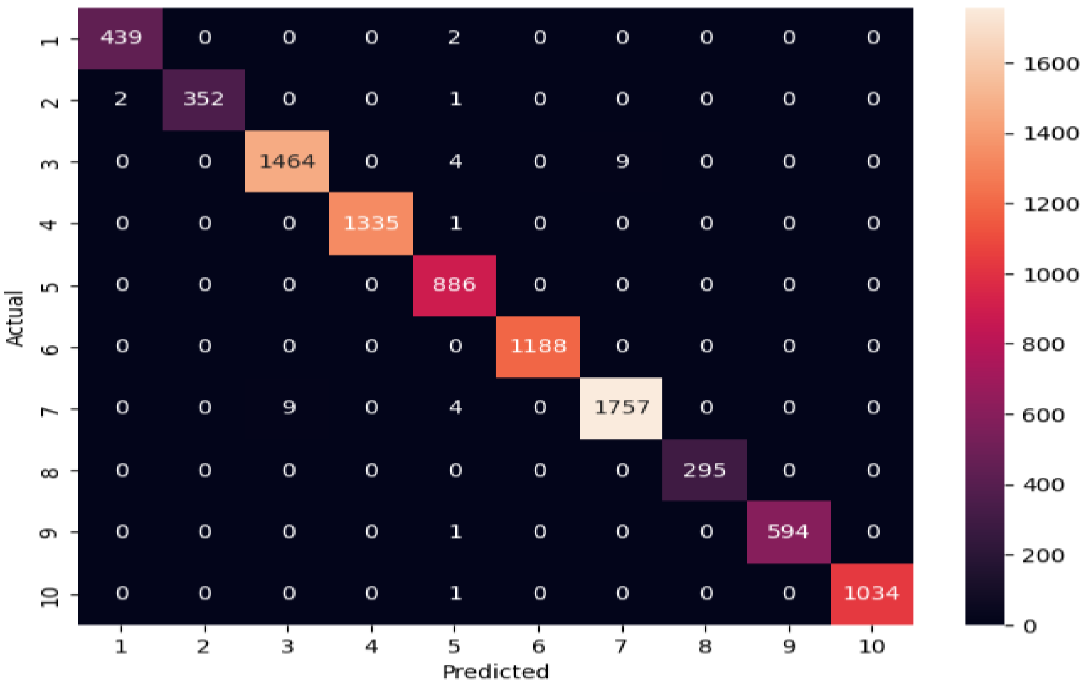
	Predict Apple	Predict Banana	Predict Burger	Predict Donut	Predict Ice Cream	Predict Pasta	Predict Pizza	Predict Salad	Predict Steak	Predict Sushi
Apple	428	2	3	0	0	0	0	0	0	0
Banana	1	350	2	0	0	0	0	0	0	0
Burger	0	0	1479	0	0	0	12	0	0	0
Donut	0	0	2	1357	0	0	0	0	0	0
Ice Cream	0	0	5	0	842	0	0	0	0	0
Pasta	0	0	2	0	0	1153	0	0	0	0
Pizza	0	0	17	0	0	0	1779	0	0	0
Salad	0	0	0	0	0	0	0	324	0	0
Steak	0	0	2	0	0	0	0	0	598	0
Sushi	0	0	2	0	0	0	0	0	0	1018

Fig6: Confusion metrics of Gradient Boosting Classifier

XGBoost Classifier

XGBoost (Extreme Gradient Boosting) is an advanced and optimized implementation of gradient boosting designed for high performance and scalability. In this project, XGBoost demonstrated superior performance by effectively handling non-linear relationships, feature interactions, and class imbalance within the nutritional dataset. It uses regularization techniques to prevent overfitting and supports parallel computation, making it highly efficient. Precision measures the proportion of correctly predicted food items for a given class out of all predicted items for that class.

Recall measures the proportion of correctly predicted food items out of all actual items belonging to that class. F1-Score is the harmonic mean of precision and recall, providing a balanced evaluation metric. XGBoost achieved higher F1-scores across most food categories, indicating balanced precision and recall. This makes it particularly suitable for real-world applications where both false positives and false negatives must be minimized.



Actual \ Predicted	1	2	3	4	5	6	7	8	9	10
1	439	0	0	0	2	0	0	0	0	0
2	2	352	0	0	1	0	0	0	0	0
3	0	0	1464	0	4	0	9	0	0	0
4	0	0	0	1335	1	0	0	0	0	0
5	0	0	0	0	886	0	0	0	0	0
6	0	0	0	0	0	1188	0	0	0	0
7	0	0	9	0	4	0	1757	0	0	0
8	0	0	0	0	0	0	0	295	0	0
9	0	0	0	0	1	0	0	0	594	0
10	0	0	0	0	1	0	0	0	0	1034

Fig7: Confusion metrics of XG Boost

- **To compare model performance using confusion or evaluation metrics**

  Comparing model performance using evaluation metrics is an essential step in selecting the most effective machine learning classifier for the NutriClass project. Since the problem involves multi-class food classification, relying solely on accuracy is insufficient, as it does not fully capture model behaviour across all food categories. Therefore, multiple evaluation metrics were used to provide a comprehensive performance comparison among the models. Accuracy was used as an initial indicator to measure the overall correctness of predictions; however, additional metrics such as precision, recall, F1-score, and confusion matrix were employed to gain deeper insights into classification quality. Precision helps identify how accurately a model predicts a specific food category, while recall measures the model's ability to correctly identify all instances of a given class. The F1-score, which is the harmonic mean of precision and recall, was particularly important in evaluating models under class imbalance conditions. Confusion matrices were used to visualize correct and incorrect predictions for each food category, allowing identification of commonly misclassified classes. By comparing these metrics across Logistic Regression, Decision Tree, Random Forest, KNN, SVM, Gradient Boosting, and XGBoost models, it became possible to evaluate strengths and weaknesses systematically. This evaluation process enabled the selection of the most robust model, ensuring high predictive performance, reliability, and suitability for real-world food classification applications.

3) Tool Used

Jupyter Notebook

It is an open-source interactive computing environment widely used for data analysis, machine learning, and scientific computing. In this project, Jupyter Notebook served as the primary development platform, allowing seamless integration of code execution, data visualization, and documentation in a single interface. Its cell-based structure enables step-by-step execution of Python code, making it easier to experiment with data pre-processing techniques, visualize results, and debug errors efficiently. Jupyter Notebook also supports Markdown, which was used to add explanations, headings, and observations alongside the code, improving project readability and presentation. The interactive nature of Jupyter Notebook made it ideal for iterative model development and comparative analysis of multiple machine learning algorithms.

Python

The Python programming language was used as the foundation of this project due to its simplicity and extensive ecosystem for data science and machine learning. Python provides strong support for numerical computation, data handling, and model implementation, making it suitable for handling large-scale nutritional datasets.

NumPy

NumPy is a fundamental Python library used for numerical operations and array manipulation. In this project, NumPy enabled efficient handling of multi-dimensional arrays, mathematical computations, and data transformations required during scaling, normalization, and feature engineering. Its optimized performance ensures faster processing of large datasets.

Pandas

Pandas was extensively used for data loading, cleaning, and manipulation. It provided data structures such as Data Frames and Series, which allowed easy handling of tabular nutritional data. Pandas was used to remove null values and duplicate records, encode categorical variables, select relevant features, and prepare input data for machine learning models. Its flexibility made data pre-processing systematic and efficient.

Matplotlib

Matplotlib is a visualization library used to generate static plots and graphs. In this project, Matplotlib was used to create box plots, histograms, and performance graphs, helping visualize nutritional distributions and identify outliers. It played a crucial role in exploratory data analysis and result interpretation.

Seaborn

Seaborn is a statistical data visualization library built on top of Matplotlib. It was used to create advanced visualizations such as KDE plots and confusion matrix heat maps. Seaborne enhances visual clarity and aesthetic appeal, making it easier to analyse data distribution and model performance.

Scikit-learn

Scikit-learn is the core machine learning library used in this project. It provides a wide range of tools for pre-processing, model training, and evaluation. Scikit-learn was used for feature scaling, imputation, encoding, train-test spliting, and implementing machine learning models such as Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbours, Support Vector Machine, and Gradient Boosting Classifier. It also offers evaluation metrics including accuracy, precision, recall, F1-score, and confusion matrix.

4) Observation

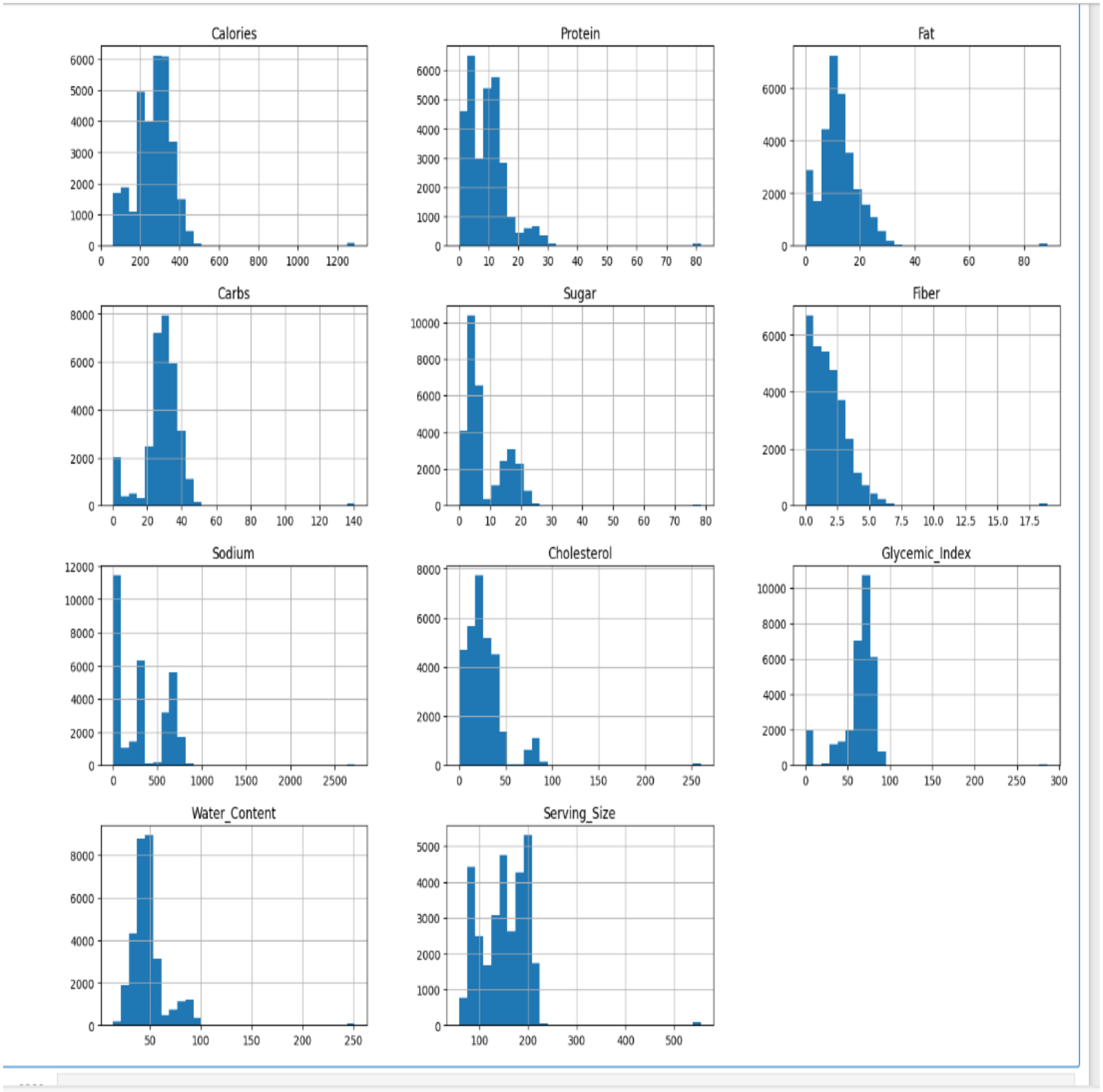
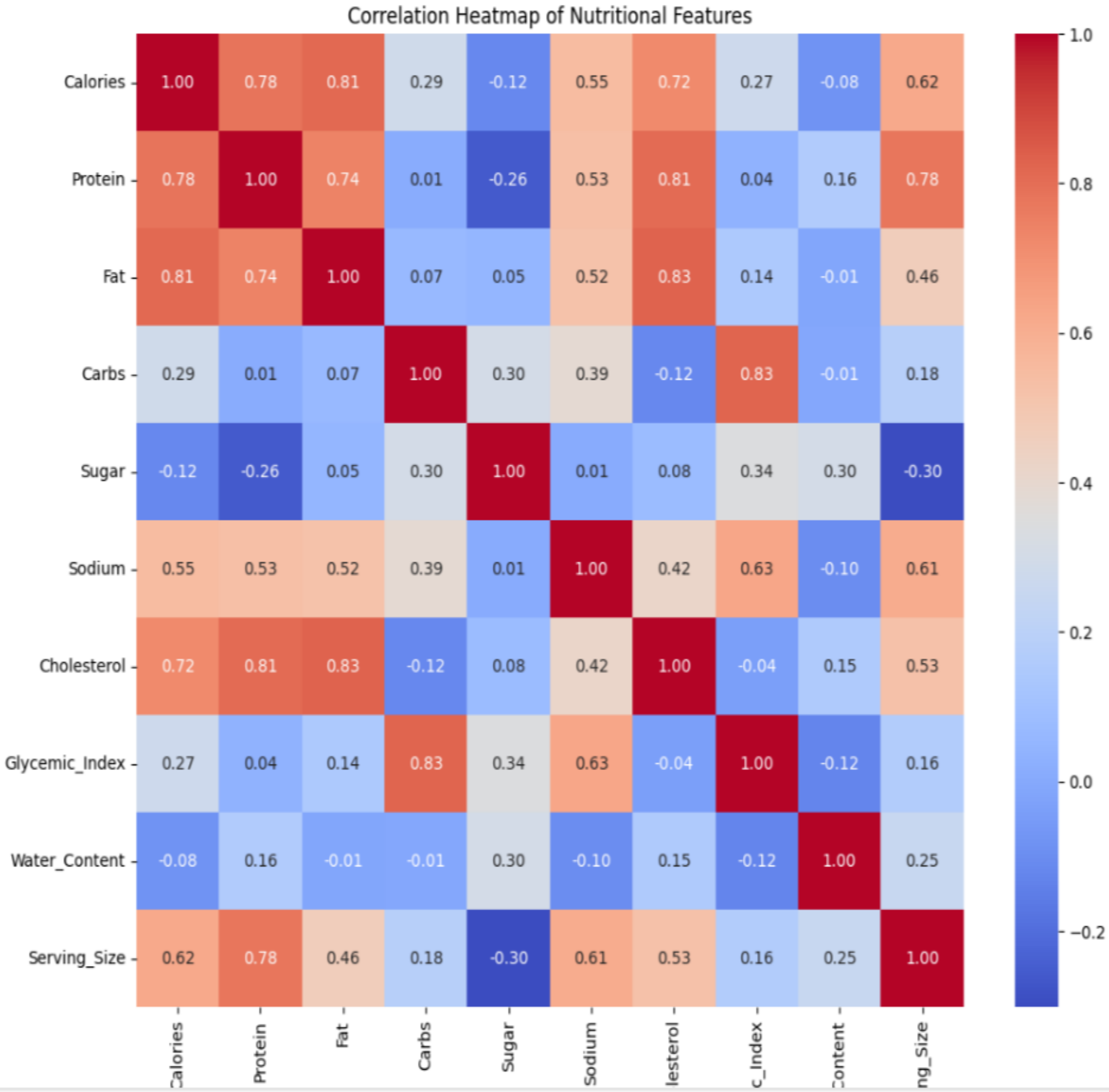


Fig8: Distribution of Nutritional Features



	Calories	Protein	Fat	Carbs	Sugar	Sodium	lesterol	c_Index	ontent	ng_Size
Calories	1.00	0.78	0.81	0.29	-0.12	0.55	0.72	0.27	-0.08	0.62
Protein	0.78	1.00	0.74	0.01	-0.26	0.53	0.81	0.04	0.16	0.78
Fat	0.81	0.74	1.00	0.07	0.05	0.52	0.83	0.14	-0.01	0.46
Carbs	0.29	0.01	0.07	1.00	0.30	0.39	-0.12	0.83	-0.01	0.18
Sugar	-0.12	-0.26	0.05	0.30	1.00	0.01	0.08	0.34	0.30	-0.30
Sodium	0.55	0.53	0.52	0.39	0.01	1.00	0.42	0.63	-0.10	0.61
Cholesterol	0.72	0.81	0.83	-0.12	0.08	0.42	1.00	-0.04	0.15	0.53
Glycemic_Index	0.27	0.04	0.14	0.83	0.34	0.63	-0.04	1.00	-0.12	0.16
Water_Content	-0.08	0.16	-0.01	-0.01	0.30	-0.10	0.15	-0.12	1.00	0.25
Serving_Size	0.62	0.78	0.46	0.18	-0.30	0.61	0.53	0.16	0.25	1.00

Fig9: Correlation Matrix of Nutritional Features

5) Future Scope: -

The NutriClass project presents several opportunities for further enhancement and real-world application. One of the key future improvements involves expanding the dataset to include a wider variety of food items, regional cuisines, and branded food products. A larger and more diverse dataset would improve model generalization and classification accuracy. Additionally, incorporating real-time data sources such as food APIs or nutritional databases can enable dynamic updates and improve model relevance over time. Advanced feature engineering techniques can be applied in the future, including nutrient ratio analysis, calorie density metrics, and personalized dietary indicators. Integrating deep learning models such as Artificial Neural Networks (ANNs) could further enhance performance, especially when handling complex nutritional patterns. Model optimization techniques such as automated hyper parameter tuning and ensemble stacking can also be explored to achieve higher predictive accuracy. The project can be extended into a complete application by integrating the model into mobile or web-based dietary platforms. This would allow users to log food items and receive instant classification and nutritional insights. Furthermore, personalization features can be introduced by considering user-specific health parameters such as age, activity level, and dietary restrictions.

From a healthcare perspective, the system can be adapted to assist nutritionists and dietitians in diet planning and disease management. Overall, the future scope of this project lies in transforming it from a standalone machine learning model into a scalable, intelligent nutrition recommendation system with real-world impact.

6) Conclusion:-

The NutriClass project successfully demonstrates the application of machine learning techniques for classifying food items based on their nutritional attributes. By utilizing structured nutritional data such as calories, protein, fat, carbohydrates, sugar, sodium, and other dietary indicators, the project highlights how data-driven approaches can effectively distinguish between different food categories. A systematic workflow involving exploratory data analysis, data cleaning, pre-processing, feature engineering, and model evaluation ensured the reliability and quality of the dataset used for model development.

Multiple machine learning classification models, including Logistic Regression, Decision Tree, Random Forest, K-Nearest Neighbours, Support Vector Machine, Gradient Boosting, and XGBoost, were implemented and compared using standard evaluation metrics. This comparative analysis provided valuable insights into the strengths and limitations of each model. Among them, ensemble-based methods, particularly XGBoost, demonstrated superior performance due to their ability to handle complex feature interactions, non-linearity, and class imbalance effectively. XG Boost accuracy is greater in all Machine Learning Model when removing all outliers and scaling.

The use of evaluation metrics such as accuracy, precision, recall, F1-score, and confusion matrices enabled a comprehensive assessment of model performance beyond simple accuracy measures. Overall, the project validates the effectiveness of machine learning in food classification and establishes a strong foundation for intelligent dietary applications. NutriClass can serve as a scalable framework for future nutrition-based decision-support systems, contributing meaningfully to health monitoring, diet planning, and smart food recommendation platforms.