12/16/2020

ML MODELLING AND EVALUATION-1002

MARCOS BITTENCOURT

Roshandeep Singh Saini
100766638

Contents

1.1 Executive Summary

An analysis is conducted to predict whether a given dish is vegetarian or not, depending on the ingredients used in the dish. In simpler terms it is expected that that a dish with same ingredients, would result in a similar dish. Predictive modelling techniques would be used to find a solution which best classifies the dish as vegetarian or non-vegetarian in terms of diet.

1.2 Problem Statement

The problem statement is "To classify a dish as vegetarian and non-vegetarian based on the ingredients".

Based on the ingredients used to prepare an Indian dish, it can be classified as a part of a vegetarian diet or a non-vegetarian diet. To gain insights on how dishes with similar ingredients can be classified similarly or differently based on region, flavor, and course.

1.3 Analytics Rationale Statement

The rationale of the analysis is to accurately predict whether a dish can be called a vegetarian dish or a non-vegetarian dish depending on the contents, flavor profile, or the course of meal. Dishes with similar ingredients can be profiled as a different course of meal as per Indian cuisine. Hence, the aim is to quantify the similarity and disparity.

1.4 Data

The data has been acquired from Kaggle open datasets. It is a raw dataset named "indian_food", which represents entirety of the testing data for August 5th to October 5th, 2020. It contains details about Indian food, the features used to classify them.
It has 255 records. These variables are: -

- Independent Variables- Actual measurement parameters of an Indian Dish which are name, ingredients, prep_time, cook_time, flavor_profile, course, state, region.
- Dependent Variable- Classification of dish as part of the diet(vegetarian and non-vegetarian).

1.5 Output Variable Structure

The output variable will have data divided in 2 classes “Vegetarian” and “Non-Vegetarian” as this is a binomial classification problem. The hypothesis is that the ingredients contained in Vegetarian and Non-Vegetarian dishes would have close cosine similarity.

1.6 Data Analysis Approach

The methodology used will be to develop a classification model using machine learning approach to segregate Indian dishes into “vegetarian” and “non-vegetarian”. It is a scenario of Supervised Learning. This would require classification algorithms SVM, Decision Tree and an advanced modelling through TensorFlow(Keras) etc. depending on the complexity and the type of data received. This would be determined in the EDA and modelling phase of the project. The software tools used will be: -

- Python – for EDA (Exploratory Data Analysis), Data Cleaning, Model Building and Testing
- Jupyter Notebook – IDE for development
- Tableau/Power BI – for detailed visualizations

1.7 Exploratory Data Analysis Action Plan

The EDA was done on the cannabis dataset according to the stated EDA Action Plan and steps were taken to clean and transform the data in corresponding to the problem statement and assumptions

1.7.1 EDA Summary

Label Conversion - Initial steps correspond to transforming the data to fit the problem statement, tokenization and vectorization of “ingredients” feature, binary data encoding for the target variable “diet” as Class 0 for “Non-Vegetarian”, Class 1 for “Vegetarian”.

NaN Values – EDA shows there is no null values in the dataset but features region and state have a value “ -1 ” which is replaced NaN so that “-1” is not treated as a separate value but an unknown value. It is not replaced with a synthetic value because it is a small dataset and even 1 value addition to a any group would create an unbalance.

Categorical Variables Encoding – Features like flavor_profile, course, state, and region are categorical variables which needed to be one-hot encoded before model building. A new binary variable is added for each unique integer value for model to work on numerical input where for categorical variables where no such ordinal relationship exists.

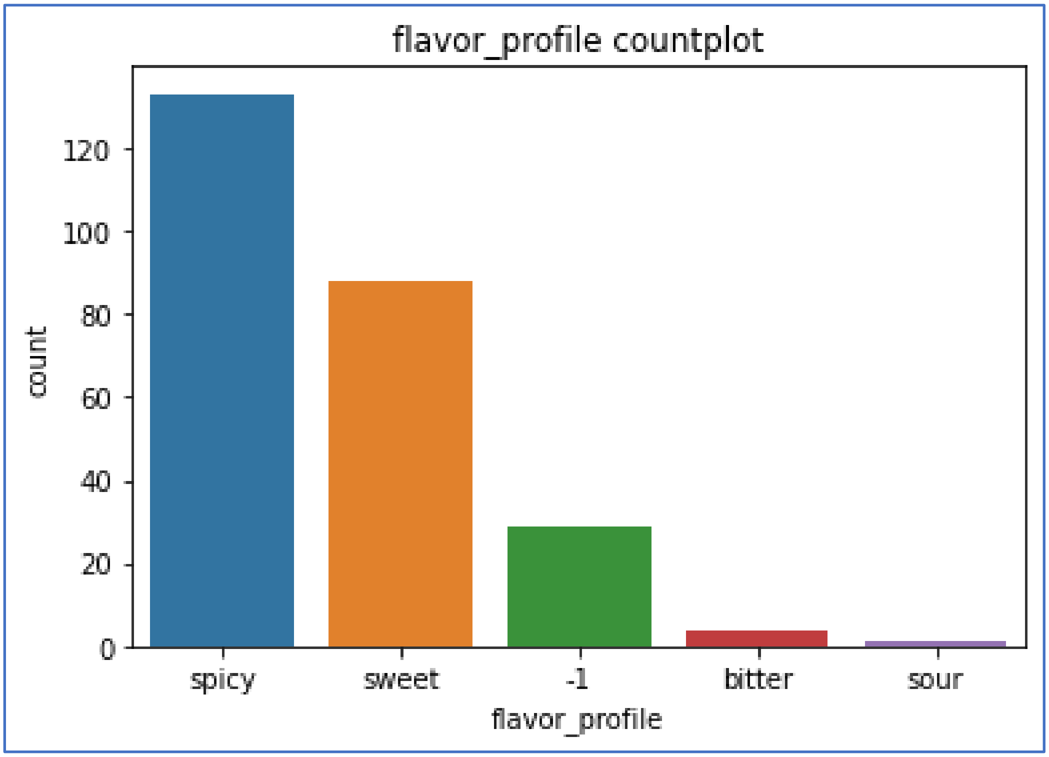


FIGURE 1 FLAVOR PROFILE DISTRIBUTION

Dependent Variable Labeling - The dependent variable “diet” was labelled and converted to numeric to be better used by the model Class 0 for “Non-Vegetarian”, Class 1 for “Vegetarian”.

Correlation - The correlation heatmap was used to check the correlation between the features and no highly correlated feature were indicated. All correlations are less than 0.8, indicating low correlation.

Duplicates – There were no duplicates in the dataset all the dishes were unique. It was comparatively a very clean dataset

Imbalance - The clean dataset has an imbalance of 226:29 for Vegetarian: Non-Vegetarian. This could have been handled by SMOTE technique before model building but because of the underlying architecture of SMOTE it would lead to overfitting of Non-vegetarian data points. The only solution for this is more data.

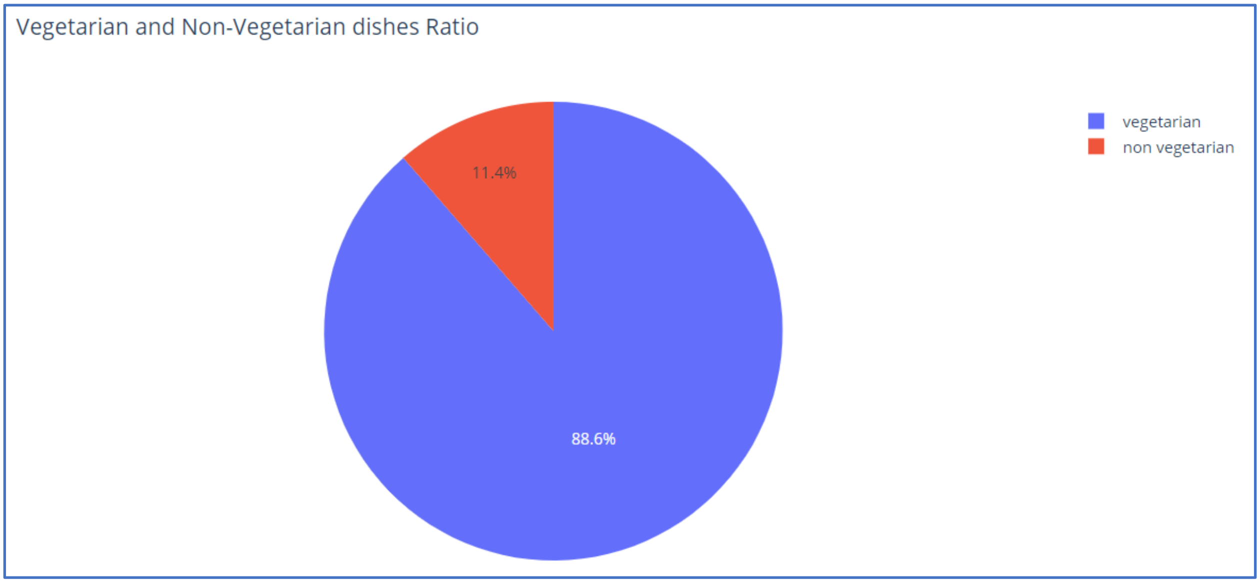


FIGURE 2 IMBALANCE IN DIET DISTRIBUTION

Vectorization - The ingredients need to be tokenized for further analysis. This is the most important feature of the model where the diet is predicted mainly using the ingredients. The food ingredients are taken, and vectors are created for each and every dish. This is similar to label encoding. This is done so that the algorithm can process the contents of the dish in the prediction process. The vectors are of the shape (255, 337) or (no of dishes, no of total ingredients)

Cosine Similarity - Ingredient vectors are used to check the cosine similarity between 2 dishes. Mathematically, it measures the cosine of the angle between two vectors projected in a multi-dimensional space. The cosine similarity is advantageous because even if the two vectors are far apart by the Euclidean distance, chances are they may still be oriented closer together. The smaller the angle, higher the cosine similarity.

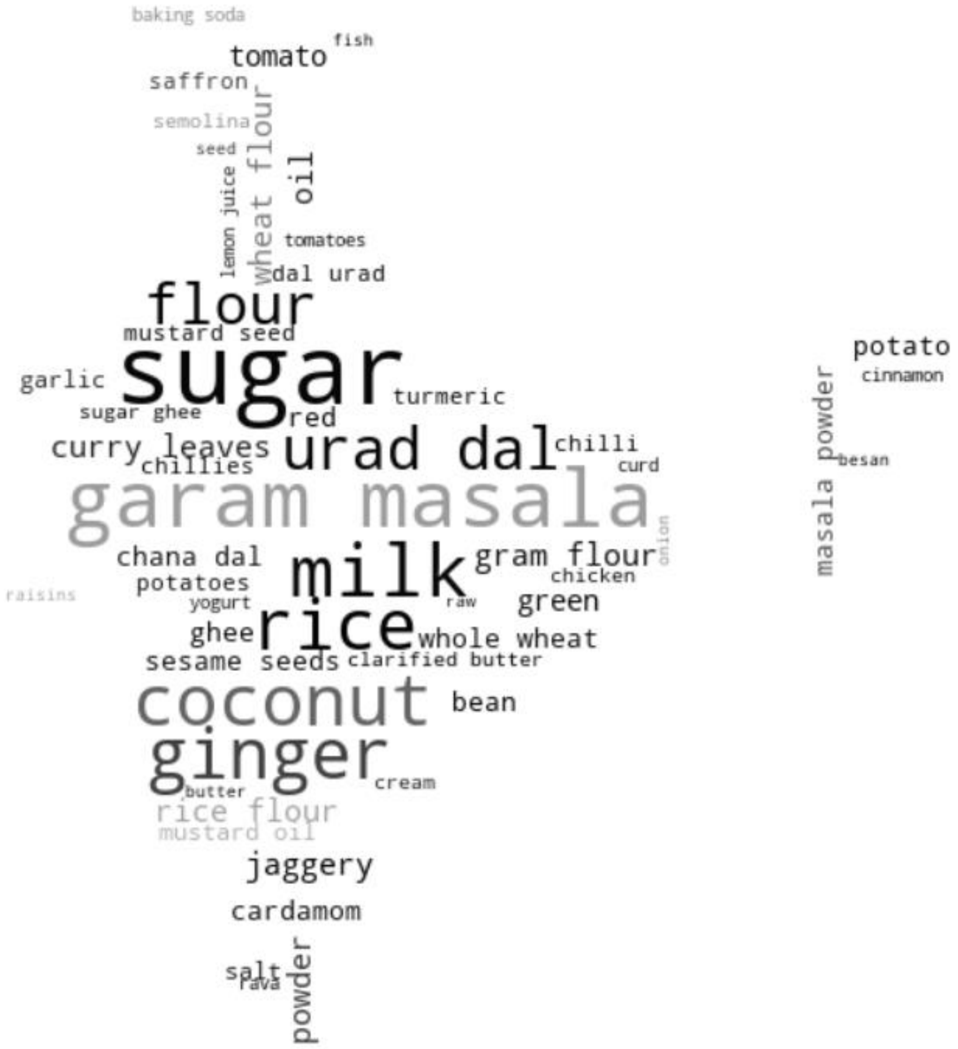


FIGURE 3 INGREDIENTS USED IN INDIAN DISHES

1.7.2 EDA Insights

The three key insights from the EDA process are following:

- No high correlation is seen between the independent variables, but on model building PCA might be required. There is little or no multicollinearity within the independent variables. Logistic Regression can be used as a base model to get an approximation of outputs.

- In the Heatmap 0 - 66th (these numbers correspond to index of "data frame") ingredients vectors have high cosine similarity each other. Cosine similarity to calculate similarity of ingredient vectors. If cosine similarity between two foods is high, it can be inferred that dishes are similar.
- There is a large imbalance between the two classes (Vegetarian and Non-Vegetarian) which might affect the model.
- As seen in the EDA dishes at location 9, 14, and 15 are sweet dishes and have very similar ingredients. Therefore, the cosine similarity between the ingredient vectors is high, indicating actual closeness between the dishes. Also seen in the EDA dish at location 30 is a savory dish compared to a sweet dish. Therefore, having a small cosine similarity between them, Indicating no actual closeness between the contents of the dishes.

1.8 Model Summary and Evaluation

The model used as a part of the Data Analysis Approach are Logistic Regression, and Neural Network. The steps for the model building are:

- The clean data prepared from the EDA process is divided into 2 part in the ratio of 80:20.
- For the Neural Network Modelling 2 sets to data is used. Firstly, the dataset without the ingredients vector data frame(created during EDA) appended to the original dataset. Secondly a dataset with the ingredient vector data frame appended. The later data frame has a dimension of (255, 405).
- The split data is first scaled to reduce the variance due to different magnitudes of the data. It also normalizes the features and improves the model.
- A standard model is created with the clean data. To evaluate the results the tools used are **Confusion Matrix**, **Classification report**, **and Learning Curve.**

1.8.1 Logistic Regression

Model Description

Logistic Regression is a Machine Learning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 or 0. In this case the dependent variable “diet” has been encoded to binomial as 1 and 0 for “vegetarian” and “non-vegetarian” respectively.

Justification For Model Selection

The reason for model selection is that Logistic Regression is simple model in terms of complexity and time constraints. Perfect to be used as a base model. It works well typically with a large sample size and a smaller number of features. For this reason PCA is conducted on the dataset and the dimensions are reduced to 175 dimensions. Logistic regression works on the assumption that there is minimal or no multicollinearity among the independent variables which make is suitable for the analysis as the dataset available show no high multicollinearity among the independent variables.

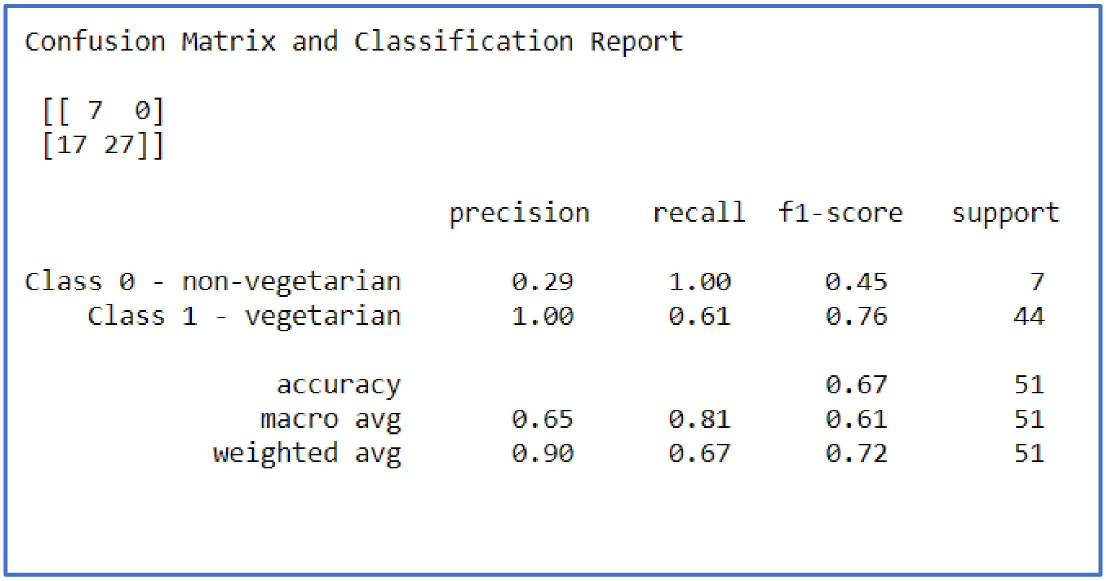
Model Evaluation

The entire dataset was divided into training set and testing set. The **Confusion Matrix** and **Classification Report** for the **Standard Model** gives the following results as seen in *Figure 4*:

- For Class 0 (Non-Vegetarian), 7 identified correctly 17 identified incorrectly.
- For Class 1 (Vegetarian), 27 identified correctly 0 identified incorrectly.
- **True Positives** – Number of correctly predicted positive values is 7.
- **True Negatives** - Number of correctly predicted negative values is 27.
- **False Positives** – Number of negative values incorrectly predicted as positive values is 17.
- **False Negatives** – Number of positive values incorrectly predicted as negative is 0.

The model has overall **Precision** of **90%** and overall **Accuracy** of **67%**

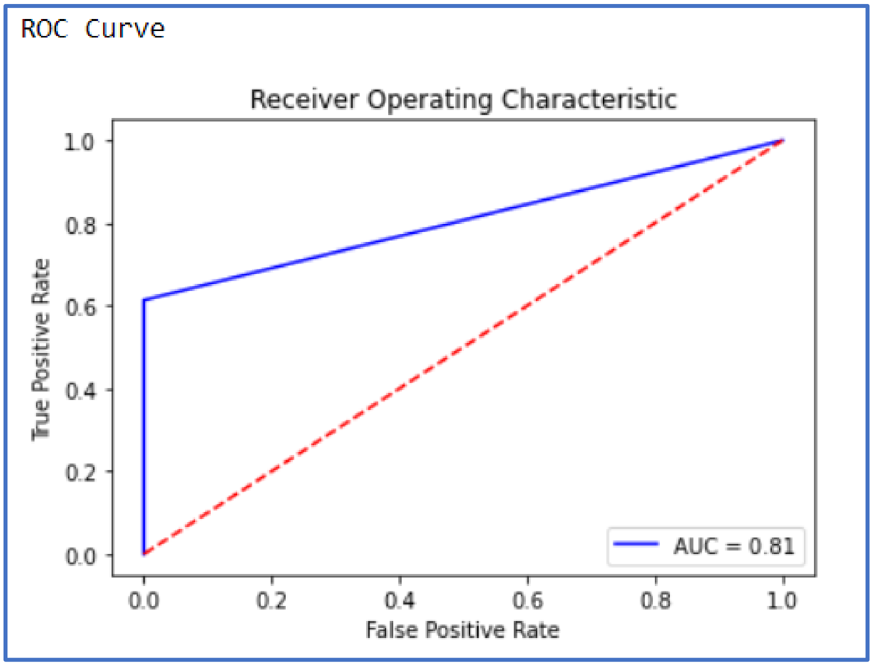
The Precision for Class 0 / Non-Vegetarian is 29% due to large number of False positives.



```
Confusion Matrix and Classification Report

 [[ 7  0]
 [17 27]]

                           precision    recall  f1-score   support

Class 0 - non-vegetarian       0.29      1.00      0.45         7
    Class 1 - vegetarian       1.00      0.61      0.76        44

                accuracy                           0.67        51
               macro avg       0.65      0.81      0.61        51
            weighted avg       0.90      0.67      0.72        51
```

FIGURE 4 LOGISTIC REGRESSION CLASSIFICATION REPORT AND CONFUSION MATRIX

ROC/AUC for Logistic Regression has a value of .81 as seen in *Figure 5*, it means there is 81% chance that model will be able to distinguish between positive class and negative class. The confusion matrix above shows that there are a larger number of False Positives which negatively impact the performance.



ROC Curve

FIGURE 5 ROC CURVE, AUC

1.8.2 Neural Network

Model Description

Neural networks are a set of algorithms, modeled loosely after the human brain, that are designed to recognize patterns. Neural networks help us cluster and classify. In a neural network a single perceptron (neuron) can be imagined as a Logistic Regression. Artificial Neural Network, or ANN, is a group of multiple perceptron's/ neurons at each layer. ANN is also known as a Feed-Forward Neural network because inputs are processed only in the forward direction.

It consists of 3 layers – Input, Hidden and Output. The input layer accepts the inputs, the hidden layer processes the inputs, and the output layer produces the result. Essentially, each layer tries to learn certain weights.

Justification For Model Selection

The reason for model selection is that Neural Networks is an advance model in terms of complexity and time constraints. It outperforms Logistic regression because it requires less formal statistical training, ability to implicitly detect complex nonlinear relationships between dependent and independent variables, ability to detect all possible interactions between predictor variables. In this case using the ingredient vectors with all the independent variables

1.8.2.1 Neural Network Without Ingredient Vectors

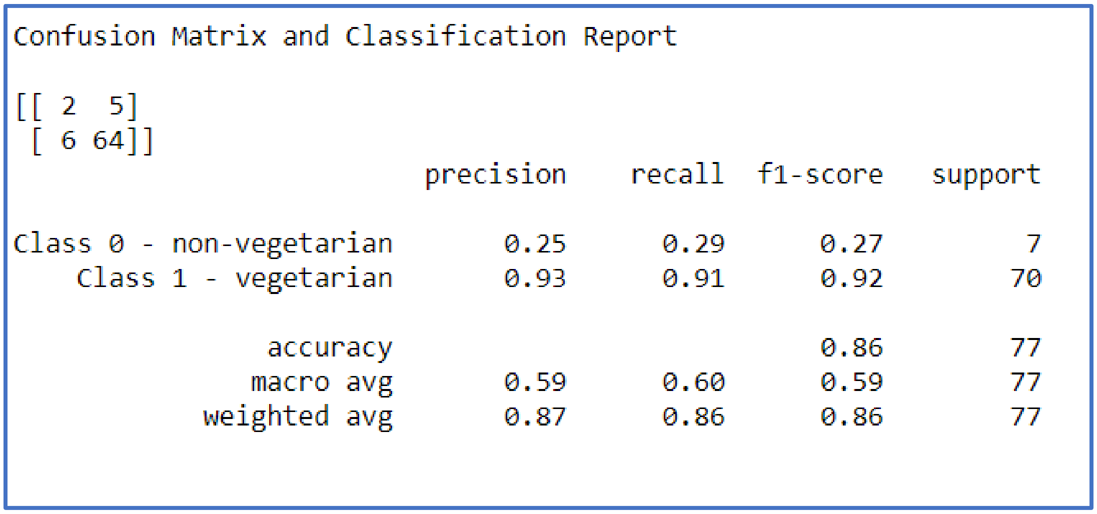
Model Evaluation

The entire dataset was divided into training set and testing set. The **Confusion Matrix** and **Classification Report** for the **Standard Model** gives the following results as seen in *Figure 5*:

- For Class 0 (Non-Vegetarian), 2 identified correctly 6 identified incorrectly.
- For Class 1 (Vegetarian), 64 identified correctly 5 identified incorrectly.
- **True Positives** – Number of correctly predicted positive values is 2.
- **True Negatives** - Number of correctly predicted negative values is 64.
- **False Positives** – Number of negative values incorrectly predicted as positive values is 6.
- **False Negatives** – Number of positive values incorrectly predicted as negative is 5.

The model has overall **Precision** of **87%** and overall **Accuracy** of **86%**

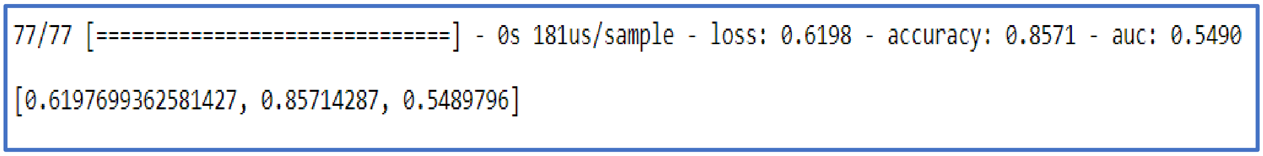
The Precision for Class 0 / Non-Vegetarian is 25% due to large number of False positives.



```
Confusion Matrix and Classification Report

[[ 2  5]
 [ 6 64]]
                          precision    recall  f1-score   support

Class 0 - non-vegetarian       0.25      0.29      0.27         7
    Class 1 - vegetarian       0.93      0.91      0.92        70

                accuracy                           0.86        77
               macro avg       0.59      0.60      0.59        77
            weighted avg       0.87      0.86      0.86        77
```

FIGURE 5 NEURAL NETWORK CLASSIFICATION REPORT AND CONFUSION MATRIX

ROC/AUC for Logistic Regression has a value of .54 as seen in *Figure 6*, it means there is 54% chance that model will be able to distinguish between positive class and negative class. The confusion matrix above shows that there are a larger number of False Positives which negatively impact the performance.



```
77/77 [==============================] - 0s 181us/sample - loss: 0.6198 - accuracy: 0.8571 - auc: 0.5490

[0.6197699362581427, 0.85714287, 0.5489796]
```

FIGURE 6 AUC SCORE

The performance of the model is plotted. As it can be seen

In the first plot the validation loss decreases with the training loss. At the end of the graph there are a few fluctuations. This is the optimum amount up to which the validation loss can be reduced. As it can be seen ahead that the position at which the loss is minimum is 40. So, it cannot be reduced further.

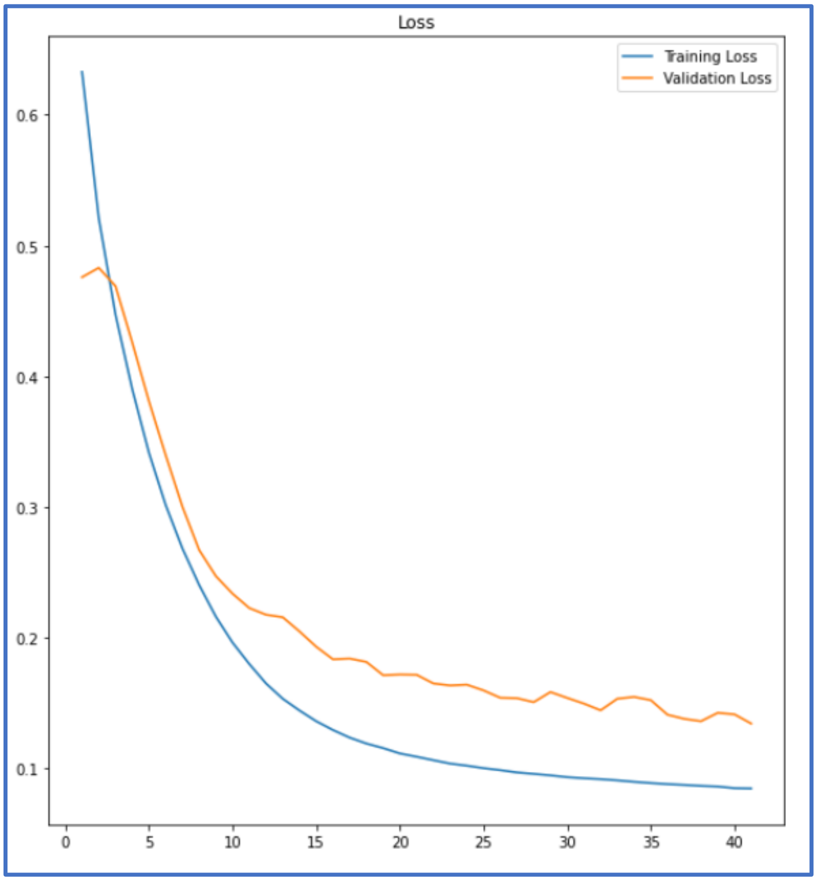


FIGURE 7 LEARNING CURVE

1.8.2.2 Neural Network With Ingredient Vectors

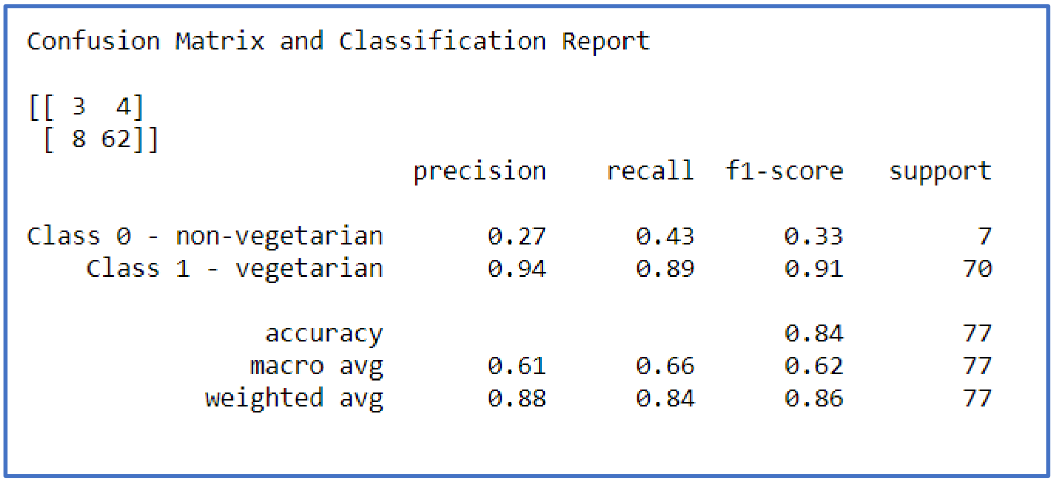
Model Evaluation

The entire dataset was divided into training set and testing set. The **Confusion Matrix** and **Classification Report** for the **Standard Model** gives the following results as seen in *Figure 8*:

- For Class 0 (Non-Vegetarian), 3 identified correctly 8 identified incorrectly.
- For Class 1 (Vegetarian), 62 identified correctly 4 identified incorrectly.
- **True Positives** – Number of correctly predicted positive values is 3.
- **True Negatives** - Number of correctly predicted negative values is 62.
- **False Positives** – Number of negative values incorrectly predicted as positive values is 8.
- **False Negatives** – Number of positive values incorrectly predicted as negative is 4.

The model has overall **Precision** of **88%** and overall **Accuracy** of **84%**

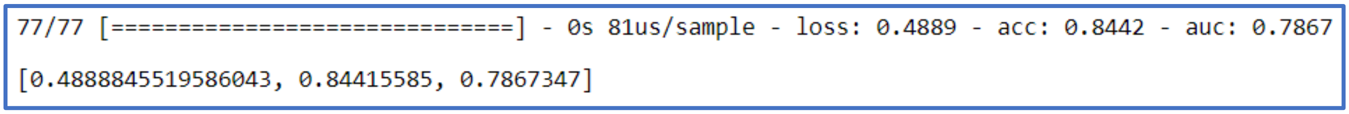
The Precision for Class 0 / Non-Vegetarian is 27% due to large number of False positives.



```
Confusion Matrix and Classification Report

[[ 3  4]
 [ 8 62]]
                          precision    recall  f1-score   support

Class 0 - non-vegetarian       0.27      0.43      0.33         7
    Class 1 - vegetarian       0.94      0.89      0.91        70

                accuracy                           0.84        77
               macro avg       0.61      0.66      0.62        77
            weighted avg       0.88      0.84      0.86        77
```

FIGURE 8 NEURAL NETWORK CLASSIFICATION REPORT AND CONFUSION MATRIX

ROC/AUC for Logistic Regression has a value of .78 as seen in *Figure 9*, it means there is 78% chance that model will be able to distinguish between positive class and negative class. The confusion matrix above shows that there are a larger number of False Positives which negatively impact the performance.



```
77/77 [==============================] - 0s 81us/sample - loss: 0.4889 - acc: 0.8442 - auc: 0.7867

[0.4888845519586043, 0.84415585, 0.7867347]
```

FIGURE 8 AUC SCORE

The performance of the model is plotted. As it can be seen

In the first plot the validation loss decreases with the training loss. At the end of the graph there are a few fluctuations. This is the optimum amount up to which the validation loss can be reduced. As it can be seen ahead that the position at which the loss is minimum is 199. So, it cannot be reduced further.

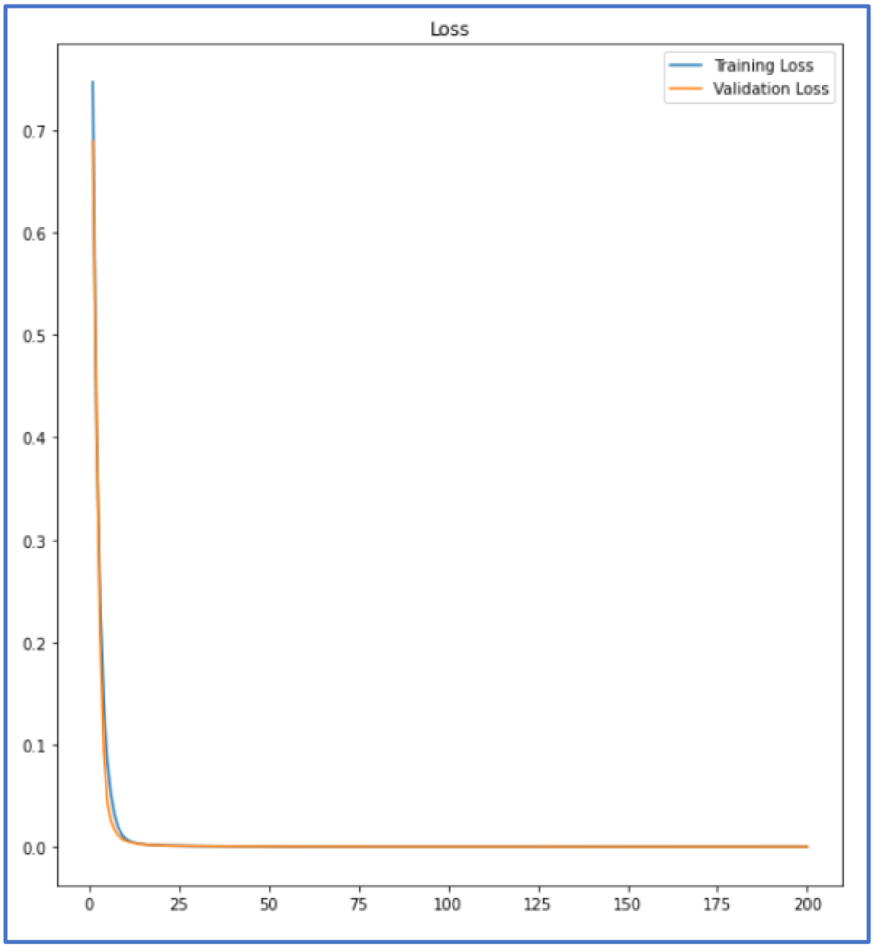


FIGURE 9 LEARNING CURVE

1.9 Insights

- The dataset used has some constraints as there are only 255 records with an imbalance in the number of samples provided. More balanced data could be used to better predict and answer the problem statement. The results of the model are able to classify the Vegetarian dishes with greater accuracy because of the bias
- The dataset was a clean dataset which did not require synthetics values being added to which is a good sign.
- There exists a high multi-collinearity among the food vectors as evident from the heatmap. This showed that some of the dishes are highly similar to the other dishes All the features are kept for the model analysis and overall dimensionality reduction is conducted which produces good results for Logistic regression. For the Neural Network model, the Input layer shape is changed as per the data dimensions
- Accuracy of 100% for Class 1 (Logistic Regression) and 95% for Class 1 (Neural network) show that the model works well for the study under consideration.
- According to the problem statement the model works well classifying the Vegetarian dishes but not accurate enough for non - vegetarian dishes because of the high imbalance.