

Foods for the Goals: A Comparative Study of Machine Learning Algorithms for Fitness Goals-Based Food Classification

Quiel Andrew I. Quiwa
College of Engineering
University of the Philippines Diliman
Quezon City, Philippines
qiquiwa@up.edu.ph

Joel Paolo C. Salvador
College of Engineering
University of the Philippines Diliman
Quezon City, Philippines
jcsalvador3@up.edu.ph

1. PROBLEM STATEMENT

Food consumption is a critical factor affecting an individual's level of fitness. While people have varying fitness goals depending on their lifestyle or daily activities, the quality and quantity of food consumed can influence how well those goals are achieved. Unfortunately, access to information on which types of food best support specific fitness goals is limited and eating healthy is often associated with higher costs.

This project explores different classifier algorithms to classify food into three fitness labels: (1) Muscle gain which is associated with high protein intake for caloric surplus; (2) Weight loss which is associated with reduced carbohydrate and fat intake to achieve caloric deficit; and (3) Endurance which is focused on high carbohydrate diet to maximize energy reserves for long exercises.

Given these labels: three primary features will be used to classify food: (1) protein, (2) fat, and (3) carbohydrate. Adjacently, the project will also explore other macronutrient trends of food within each fitness label to better understand the characteristics that differentiate these categories. Through this classifier, the project aims to improve access to information on food options that match a person's fitness goal. Potential use cases in the future can also be expanded towards addressing macronutrient deficiencies, as well as specific dietary restrictions of individuals.

2. RELATED LITERATURE

One of the barriers to eating healthy, as identified by Solan, is the misguided assumptions of people, who usually view eating healthy as being too expensive and too difficult to prepare in order for them to adopt nutritious diets [1]. Nutrition plans, including macronutrient breakdowns, should be tailored to each individual, depending on multiple factors, including fitness goals [2].

For muscle gain, the recommended daily protein intake is at least 2.2g protein/kg of an individual's weight [3]. For weight loss, a daily caloric deficit of 500 kcals is identified to reduce body fat while preserving muscle [4]. For endurance, about 7-12g carbohydrates/kg of an individual's weight is required for heavy training [5].

Similar use cases have recently been explored by machine learning researchers. Al Qohar, et. al classified Indone-

sian food based on their nutritional profiles using KNN, SVM, random forest, and multi-layer perceptron (MLP) [6]. Senthilmurugan, et. al, evaluated a SVM classifier for fast food nutrition data [7]. Ajami and Teimourpour developed a food recommender system using AdaBoost [8]. Ahuja, et. al developed a food classification algorithm using KNN and Naive Bayes [9].

Overall, this project will leverage on these literatures to improve access to information on fitness goals-based food, while also diving on the macronutrient trends among the foods within the fitness classes.

3. DATASET

The dataset that would be used is the Food Nutrition Dataset by Utsav Dey on Kaggle. It is a comprehensive database of different foods containing information about their macro and micronutrients.

In addition, all other features, including the three mentioned previously, will be used to identify the macronutrient trends of foods within each fitness class.

4. METHODOLOGY

1. Data Pre-processing

For data pre-processing, feature columns were first identified to ensure that only relevant nutritional attributes were included in the model. The features identified were primarily macronutrients, particularly protein, caloric value, carbohydrates, sugars, sodium, dietary fiber, and fat. These are the features that were used to assign the labels for the entries. Later in the experiment, additional feature columns were also included to test the model with additional features but were not used in labeling.

After the feature columns for consideration were decided, an exploratory data analysis was conducted using the pandas library to gain an initial understanding of the dataset. This included checking for the shape of the dataset, dropping rows with empty values for the selected features, and assigning the dataframes for the features and labels to their respective matrix variables. Some basic EDA statistics were also explored by describing the dataframe to gain a general understanding of dataset:

The values above assured the project team that the dataset is generally within the acceptable and expected ranges before proceeding with model development.

2. Threshold-based labeling

Since the objective of the project is to compare classifiers using supervised learning methods, and the dataset used did not initially have labels, label assignment rules were applied on each row based on studies related to macronutrient features gathered from nutrition-based journal articles. Although the project recognizes that the best results can be achieved through proper consultation with a certified or professional nutritionist, the labels were decided with the following logic:

1. The rows were labeled as "Muscle Gain" if the food macronutrients contained Protein $\geq 20\text{g}$, or if Protein $\geq 15\text{ g}$ and Calories $\geq 250\text{ kcal}$. Nutrition research articles suggest that Muscle Protein Synthesis rates or the process that turns muscle into protein were maximized by ingesting a 20g dose of whey protein after a protein-rich breakfast [cite]. Additionally, there is research showing that MPS can be maximized in young adults with an intake of 20-25g of high quality protein [cite]. Caloric value was also included in this label since several journals have referenced that energy surplus is required to facilitate muscle hypertrophy [cite]. Although no exact value has been determined, the project selected anything greater than 250kcal despite having only about 15g protein could still be labeled as food for muscle gain.
2. The rows were labeled as "Endurance" if the food macronutrient values were Carbohydrates $\geq 30\text{ g}$, or Sugars $\geq 12\text{ g}$, or (Carbohydrates $\geq 20\text{ g}$ and Sodium $\geq 140\text{ mg}$). This was based on articles citing endurance requiring 30-60 g of carbohydrate intake for endurance events and regular ingestion of sodium beverages for proper hydration [cite]. Sugars such as glucose, sucrose, or carbohydrates that are simple sugars were also identified to contribute to endurance so this was also selected in the criteria [cite]. This is why food which were not above 30g of carbohydrates, but had significant amounts of sugars and sodium were also included in the labeling rules.
3. Lastly, the rows were labeled as "weight loss" if the food macronutrient values were Calories $< 150\text{ kcal}$, Fat $< 5\text{ g}$, and (Carbohydrates $< 20\text{ g}$ or Dietary Fiber $\geq 3\text{ g}$ or Protein $\geq 10\text{ g}$). Lower calorie intake is a requirement to achieve caloric deficit for weight loss, and 150kcal is a typical value for smaller portions of snacks and general food items. However, other macronutrient thresholds were also identified as requirements for this deficit to be considered healthy for weight loss. A study found that combining increased fruit and vegetable intakes with decreased fat intake, is an effective strategy for managing body weight while controlling hunger [cite]. Additionally, there are also studies showing that fiber intake is associated with lower body weight [cite]. Hence, an added requirement for this label is to have lower fat content and either one of having low carbohydrates, high dietary fiber, and relatively higher protein. Across all of these additional criteria, the low calorie measure remains a requirement.

The python-based code logic for setting the threshold explained is shown below:

```
def assign_label(row):  
    # Muscle Gain  
    if row["Protein"] >= 20 or (row["Protein"] >= 15 and row["Calories"] >= 250):  
        return "Muscle Gain"  
  
    # Endurance  
    if row["Carbohydrates"] >= 30 or row["Sugars"] >= 12 or (row["Carbohydrates"] >= 20 and row["Sodium"] >= 140):  
        return "Endurance"  
  
    # Weight Loss  
    if (row["Caloric Value"] < 150) and (row["Fat"] < 5) and (row["Carbohydrates"] < 20 or row["Dietary Fiber"] >= 3):  
        return "Weight Loss"  
  
    # Default  
    return "No Category"
```

Value counts were implemented to ensure that the threshold based labeling for the dataset included enough values for each label:

3. Standardization for Model Comparison

To ensure that the models are compared properly and accordingly, a standardized model development procedure was applied across the different algorithms. GridSearchCV was used, except for Naive Bayes, to identify the best combination of the parameters before each model was evaluated.

All models were trained and assessed using a 5-fold cross-validation, which splits the dataset into 5 equal parts, and produces 5 separate performance scores. This 5-fold cross-validation ensures that every data point in the dataset is used both for training and testing, which reduces bias. The 5 performance scores generated were then averaged out to get its mean performance.

The metrics computed for each model were as follows: (1) accuracy, (2) precision, (3) recall, and (4) f1-score. These scores provide a complete picture of how well each model performed across the three different categories.

Lastly, confusion matrices were also generated for each classifier to help visualize the misclassifications that occurred in each model.

4. Libraries Used

a. SciKit Learn

The following modules and classes from SciKit Learn were used in this project:

b. Pandas – This library was primarily used for exploratory data analysis, including reading the dataset, generating initial insights out of the datasets, and providing tabular visualizations on the data.

- c. Matplotlib Library - This was used to generate output data visualizations, mainly the confusion matrix plot.
- d. Seaborn - This library was also used to generate the necessary data visualizations on top of the matplotlib library.
- e. OS and Glob - This library was used to load the dataset.

5. Four Classifier Models

a. Naive Bayes Classifier

For the Naive Bayes Classifier, Gaussian Naive Bayes was used as the algorithm in this project since nutritional components food in the dataset are continuous numerical values. Naive Bayes, unlike other models, does not require tuning so no parameter optimization using GridSearchCV was performed for this model.

In this project, the Naive Bayes mainly served as a baseline for comparison of the different models. Because the algorithm relies on strong independence assumptions between features, it generally performs well on simple, cleanly separated data. However, Naive Bayes is less flexible than the other three algorithms. As such, this served as the baseline for improvement comparison of the other advanced models (Support Vector Machines, Multilayer Perceptron, and AdaBoost) over a straightforward Naive Bayes classifier.

b. Support Vector Machines

In this project, a Support Vector Machine classifier was also explored. Given that nutritional data do not form clean linear boundaries, the SVM model used Radial Basis Function (RBF) as the kernel. RBF is useful since it allows non-linearly separable variables to be mapped into a higher-dimensional space where classification becomes easier.

Two parameters were tuned for SVM, namely, C and gamma. C controls how strict or flexible the margin is, and gamma determines how far the influence of each training point reaches. These parameters shape the decision boundary of the RBF kernel.

To tune the parameters, GridSearchCV was used to test different combinations of the values of $c = 0.1, 1, 10$, and of the values of $\gamma = 0.01, 0.1, 1$, with the goal of finding the combination of the parameters that yields the highest macro-averaged F1 score.

The best combination of c and gamma from the grid search was then used to evaluate the final SVM model.

c. AdaBoost Classifier

In addition to Naive Bayes and SVM, an AdaBoost Classifier was also used to explore whether using multiple weak learners at different learning rates could improve the classification performance for this nutrition dataset.

Initially, an arbitrary base value of 100 weak learners with the learning rate set to 1 was tested. This was discovered to have a decent performance at Mean F1 = 0.968522. Hence,

tuning these parameters involved testing values at the +/- 100 range from the base value of the number of weak learners and +/- 0.9 range from the base value of the learning rates. The GridSearchCV class was also used to calculate all possible values of the parameters by iterating across all combinations of the range provided. The best combination of weak learners and learning rates calculated using GridSearchCV was used for the final model

Additionally, this AdaBoost Classifier utilized the SAMME algorithm since the SciKit Learn library's default AdaBoost algorithm (SAMME.R) has been deprecated. SAMME or "Stagewise Additive Modeling using Multi-class Exponential Loss" is an extension of AdaBoost used to handle problems with more than two classes using the following adjusted computation:

AdaBoost weight computation:

$$\text{alpha} = 0.5 * \log((1 - \text{error}) / \text{error})$$

SAMME adjusted weight computation:

$$\text{alpha} = \log((1 - \text{error}) / \text{error}) + \log(K - 1)$$

K is the number of classes, so that the extra term strengthens the confidence of accurate weak classifiers in multiclass settings. More classes mean each weak learner is less likely to guess correctly by chance. The term $\log(K-1)$ adjusts for that, rewarding strong learners more.

d. Multi-Layer Perceptron Classifier

Finally, the project tested the dataset using a Multi-Layer Perceptron Classifier. In this case, up to three hidden layers of backpropagation were initially tested using SciKit Learn's MLPClassifier class. The methodology did not explore more than three hidden layers since the result of the model at two hidden layers was already found to be decent, and increasing the hidden layers to three no longer resulted in significant improvement on the model.

The neuron sizes for each hidden layer here were tested following a range of 50 to 200 neurons. The alpha values were tested on different ranges from 0.0001 to 0.01. The learning rates were tuned from 0.0005 to 0.0025. Once again, the GridSearchCV class was used to calculate the possible values of the parameters by iterating across all possible combinations of the hidden layer sizes, alpha values, and learning rates within the range and values provided. The best combination was used for the final model.

6. Results, Analysis and Discussion

a. Model Evaluation

i. List down the scores

ii. Highlight the highest F1-score

iii. Justify why F1 was used

IF ANALYSIS ONLY.

Based on the results, the model that achieved the highest accuracy score of XX was XXX model. On the other hand, XXX model reached the highest precision score, while XXX model gained the highest recall score.

However, in this project, the models will be evaluated based on the highest F1-score. This score avoids any issue brought by the not evenly distributed counts of food included in each class, especially on accuracy alone. F1-score is also a function of both precision and recall, which makes it a reliable metric to be used in determining how well the model performs in each class (e.g. how well it avoids false positives and false negatives across all classes).

<To insert: confusion matrix analysis in latex>

As such, the best model is Adaboost as it achieved the highest F1-score of XXX.

b. Data Analysis / Visualization

To better understand the three fitness goal categories that we have, especially on the different nutritional components that separate these three, exploratory analyses were conducted. To note, this is not to build another classifier, but to explain why the models behave the way they do and which nutrients actually drive the classification.

ANOVA (Analysis of Variance)

A one-way ANOVA was performed across all nutrient features of the dataset to identify the specific nutrients that show statistically significant differences among the three classes with a threshold of $p < 0.05$.

From the ANOVA, it showed that there are several nutrients that vary strongly between the three fitness goals, this includes phosphorus, protein, caloric value, carbohydrates, potassium, vitamin B3, magnesium, and nutrition density, among others. This means that the three fitness goal categories vary largely in these nutrients.

<include here the list of the ANOVA results in latex>

These nutrients were expected to be the top nutrients with very high F-scores since muscle gain foods are high in protein, and caloric values, endurance foods are high in carbohydrates and sugar, and weight loss foods are low in fat, carbohydrates, and calories. These are just some of the observations that were translated into the assigning of the label code block of the project.

Using ANOVA, it also noted that there are other significant features in each fitness goal category on top of the ones used in our label assignment code block.

<include here the list of significant features in latex>

Extended Feature Columns Included in the Dataset used in the Model

These significant features listed by ANOVA were then tested to be included in the dataset used in the 4 algorithms, with

the main question to answer “will the performance of the models differ if we introduce the extended feature columns?”

After doing such, the answer we got is it did not make the performance of the models better, but rather, made it worse. The primary reason here is that the feature space of the model increased but we did not update the code block setting the criteria for the food classification. As such, the additional feature columns became noise to the models as they were never used in label definition.

Logistic Regression

Lastly, in this project, logistic regression was also used, not as an additional model to be compared, but as a tool to interpret the results. This is to understand how the nutrients influence the likelihood of that food being labeled into one of the categories.

The regression model was fitted both with all features in the dataset, and with the significant features identified from ANOVA. A positive coefficient means that it increasing that nutrient in the food will make its probability of being assigned to a certain class higher, while a negative coefficient makes it otherwise.

The results of logistic regression fitted using all features are as follows: for muscle gain foods, caloric value, and protein, among other nutrients showed strong positive weights for this class; for endurance foods, carbohydrates and sugars had positive values, and; for weight loss foods, protein, calories, and fat had negative weights. These results support the criteria used in assigning the labels.

For the logistic regression that fitted only the significant features found in the ANOVA above, it showed a somewhat similar set of results to the logistic regression fitted using all features. This reinforced the findings of the earlier logistic regression.

Combining the results of ANOVA and Logistic Regression helped validate the logic used in classifying the entries inside the dataset.

5. REFERENCES

- [1] M. Solan, “Why is eating healthy so hard?,” Harvard Health, Feb. 01, 2022. <https://www.health.harvard.edu/nutrition/why-is-eating-healthy-so-hard>
- [2] Better Health Channel, “Sporting performance and food,” Vic.gov.au, Apr. 17, 2023. <https://www.betterhealth.vic.gov.au/health-performance-and-food>
- [3] J. Iraki, P. Fitschen, S. Espinar, and E. Helms, “Nutrition Recommendations for Bodybuilders in the Off-Season: A Narrative Review,” Sports, vol. 7, no. 7, p. 154, Jun. 2019, doi: <https://doi.org/10.3390/sports7070154>.
- [4] E. R. Helms, A. A. Aragon, and P. J. Fitschen, “Evidence-based recommendations for natural bodybuilding contest preparation: nutrition and supplementation,” Journal of the International Society of Sports Nutrition, vol. 11, no. 1, May 2014, doi: <https://doi.org/10.1186/1550-2783-11-20>.

- [5] "DIETARY CARBOHYDRATE AND THE ENDURANCE ATHLETE: CONTEMPORARY PERSPECTIVES," Gatorade Sports Science Institute, 2022. <https://www.gssiweb.org/research/article/dietary-carbohydrate-and-the-endurance-athlete-contemporary-perspectives>
- [6] "View of Machine Learning Techniques for Classifying Indonesian Foods and Drinks by Nutritional Profiles," Shm-publisher.com, 2025. <https://www.shmpublisher.com/index.php/joiser/article/view/528/277> (accessed Nov. 17, 2025).
- [7] M. Senthilmurugan, Nagendar Yamsani, C. Mary, Loganayagi S, and A. Akilandeswari, "Evaluation of Support Vector Machine and Kernel Neural Network Classification for Fast Food Nutrition Data," 2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS), pp. 150–154, Aug. 2023, doi: <https://doi.org/10.1109/ICAISS58487.2023.10250603>.
- [8] A. Ajami and B. Teimourpour, "A Food Recommender System in Academic Environments Based on Machine Learning Models." Accessed: May 24, 2024. [Online]. Available: <https://arxiv.org/pdf/2306.16528.pdf>
- [9] A. Ahuja, V. Kedia, M. Ansari, B. Grover, and Shubha Puthran, "Food Classification using Machine Learning Algorithms," Advances in computer science research, pp. 1034–1046, Jan. 2025, doi: https://doi.org/10.2991/978-94-6463-858-5_86.