Apple Quality Assessment Using Machine Learning Models

Ananya Ratakonda*, Vibha Chandrasekar†, Mandy Zhu¶, Marina Mata de la Barata Barcons‡, and Maithreyi Narayanan§

*Fourth-year Computer Science Student at UC Davis - Team lead

†Fifth-year Computer Science Student at UC Davis - Team member

¶ Third-year Statistics Student at UC Davis - Team member

‡Fourth-year Computer Science Engineering Student at UC Davis - Team member

§Fourth-year Cognitive Science Psychology Student at UC Davis - Team member

Email: (aratakonda, vibchand, mmbmata, mnarayan, mjizhu)@ucdavis.edu

I. Introduction

Apples are everywhere, from kitchens to grocery stores worldwide. Apple quality is crucial in helping customers purchase only the highest-quality produce. For farmers, researchers, and retailers, having an easy and quick method to assess apple quality is essential to ensure a future consumer's satisfaction.

II. Background

The agricultural industry faces the ongoing challenge of assessing apple quality, which directly impacts consumer loyalty and market value. When customers are given inconsistent apples, they are less likely to continue purchasing from the same source. Additionally, without a standardized quality evaluation process, apple prices can fluctuate, impacting producers and retailers. Today, most quality inspections are visual checks that are subject to human error and are time-consuming. These methods are neither accurate nor efficient enough to thoroughly test apple sizes, surface quality and other necessary features. Inaccurate quality assessments have many negative impacts that may affect farmers, consumers, and the overall agricultural distribution network.

Current research in machine learning (ML) models for agricultural quality control shows the impactful change ML algorithms can make. A fruit detection model such as the YOLO-based system for on-tree fruit detection was able to achieve an accuracy of 90% [1], demonstrating the effectiveness of these technological models. Furthermore, a quality algorithm for fruits such as bananas and tomatoes has been able to achieve over 90% accuracy in its quality assessments [1], further highlighting how reliable ML algorithms can be. These results provide clear evidence as to why it is effective to use ML models to improve the agricultural industry. This project builds upon this idea by utilizing Multilayer Perception, Support Vector Machine, and Random Forest models to assess apple quality.

To address the lack of standardized apple quality assessments, an automated approach to apple quality assessment using machine learning is proposed. This project takes in attributes such as sweetness, crunchiness, juiciness, ripeness, and acidity and uses binary classification to predict if the quality of an apple is 'good' or 'bad'. The models developed are: Multilayer Perceptron (MLP), Support Vector Machine (SVM), and Random Forest (RF). By utilizing these models, the solution aims to deliver an accurate and efficient method of evaluating the quality of apples, benefiting the apple supply chain.

The remainder of the report follows the following structure: Section 3: Literature Review of prior research and methodologies relevant to apple quality assessment. Section 4: Dataset Description of the apple quality dataset and data preprocessing techniques used, such as outlier removal. Section 5: Exploratory Data Analysis (EDA) revealing findings within feature distribution, trends, and correla-

tions. Section 6: Proposed Methodologies outlining the approach to creating these models. Section 7: Experimental Results and Evaluation provides a comparison among performance metrics of the models. Section 8: Conclusion and Discussion, which provides recommendations for future real-world applications of these models. Lastly, Sections 9, 10, and 11 outline project timelines, team contributions, and a link to the code for this project.

III. Literature Review

Before developing the machine learning models, the first step was to understand prior methods and experimentation regarding this topic. While there has been extensive research on image classification of various fruits using different models like CNNs, there is a lack of research focusing on other internal attributes of fruits, like ripeness, acidity, and sweetness, rather than just their external appearance. Other qualitative measurements, like weight and size, that can not easily be determined with images alone, are also important factors in determining the quality of fruits.

Starting off, we examined early research regarding the classification of apple quality. Singh, S., Singh, N.P. (2018) developed an image classification model that determines where the apple is good or rotten. They utilized k-fold cross validation with various models such as Logistic Regression, SVM, k-Nearest Neighbors, and Linear Discriminant. After evaluating their models, they came to the conclusion that SVM was the best choice with an accuracy of 98.9% [2]. Although these were used for image classification, we can still use this knowledge to guide the selection of the machine learning models for this project moving forward.

More recently in October 2024, Cengel, T.A., Gencturk, B., Yasin, E.T. et al. (2024) published their research regarding apple quality, discussing in-depth about the qualities of their dataset, the different models they developed, and their evaluation of those models. The different types of models created were Multilayer Perceptron (MLP), Random Forest (RF), Decision Trees (DT), and a Support Vector Machine (SVM). According to their research, the model with the highest accuracy was the Multilayer Perceptron (MLP). Although the accuracy of the models were all high, MLP had the highest accuracy of 95.63%, and DT had the lowest accuracy of 81% [3]. The evaluation of the accuracy and recall values for each model in this paper provided a foundation for deciding to implement MLP, SVM, and RF models in this project, with the goal of achieving high accuracy after removing outliers from our dataset. In addition, we will implement hyper-parameter tuning to further optimize our models.

IV. Dataset Description

This study uses a data set of approximately 4,000 apple samples, each characterized by eight numerical features and one categorical target variable, ”Quality”. The numerical features include A_id, Size, Weight, Sweetness, Crunchiness, Juiciness, Ripeness, and Acidity. The A_id attribute uniquely identifies each sample and functions as the dataset’s index. The remaining numerical features represent the physical and sensory characteristics of the apples, and these values have already been normalized. The target variable, “Quality”, classifies each apple as ’good’ quality or ’bad’ quality, making it the focal point for subsequent classification tasks.

Initial data pre-processing ensured consistency and data readiness. The “Acidity” feature, originally stored in a non-numeric format, was converted to a floating-point data type to align with the dataset’s numerical structure. This adjustment ensured consistent data types across features, facilitating accurate computations and visualizations. A thorough inspection of the dataset confirmed the absence of null or missing values, ensuring completeness without the need for imputation.

The dataset’s structure and distributions were assessed using histograms (Figure-1) for each numerical feature. The features exhibited approximately normal distributions, confirmed by the alignment of their mean and median values. This indicated minimal skew and confirmed that the dataset was standardized, eliminating the need for additional scaling or normalization. These observations streamlined the pre-processing workflow and allowed attention to shift toward addressing outliers.

Outliers were identified and addressed to enhance data quality and to minimize their influence on downstream analysis. Boxplots highlighted the presence of extreme values across features, which

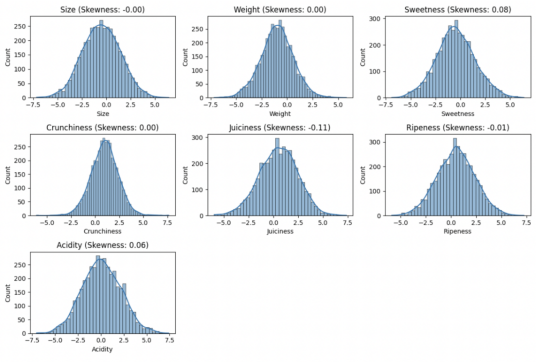

Fig. 1. Feature Skewness Visualization using Histograms

was solved by using the Interquartile Range (IQR) method to identify outliers and remove them. A total of 210 samples (about 5%) of the dataset was removed, which amounts to 20 to 54 outliers per feature. Removing these outliers improved the dataset's quality, making the analysis more accurate and reliable while ensuring the results were trustworthy.

Post-preprocessing histograms were replotted to validate the effectiveness of outlier removal. While the features retained their normal distributions, variability was significantly reduced, resulting in a cleaner and more consistent dataset. After this process, the dataset was reduced from 4,000 to 3,790 samples.

By addressing inconsistencies, ensuring completeness, and removing outliers, the dataset was prepared for reliable exploratory analysis and classification modeling. These pre-processing steps established a strong foundation for extracting meaningful insights.

V. Exploratory Data Analysis

The exploratory data analysis (EDA) process began with the creation of a pairplot (Figure-2) to examine the relationships between numerical features and their connection to the categorical target variable, "Quality." In the pairplot, blue represented good-quality apples, while orange represented bad-quality apples. Additionally, density plots for each feature provided an overview of their individual distributions. For example, features such as sweetness and juiciness showed overlapping distributions between good and bad quality apples, indicating these variables might not strongly differentiate quality. However, certain features, such as acidity and crunchiness, displayed slightly distinct patterns, with good-quality apples clustering more tightly around specific values compared to bad-quality apples. From this visualization, it was observed that no features displayed visibly strong correlations with one another.

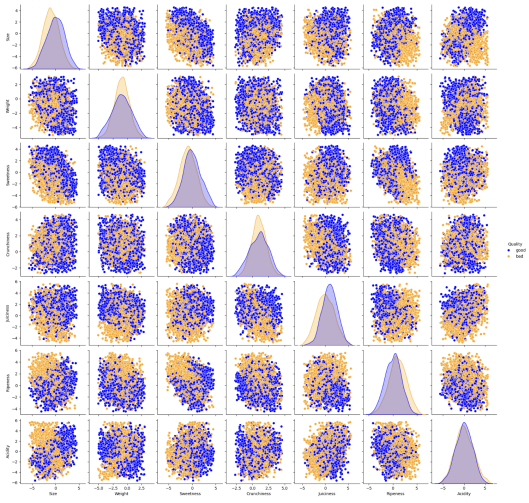

Fig. 2. Pair-plot Between Features and Target

To quantitatively assess the relationships among the numerical features, a correlation matrix using Pearson correlation coefficients was computed. This was visualized as a heatmap (Figure-3). The heatmap revealed that all features exhibited weak correlations, and no two features showed a strong linear relationship. This finding suggests that each feature provides distinct information for analysis. The distribution of each numerical feature was further investigated using histograms (Figure-4). These histograms confirmed that all features followed a normal distribution with minimal skew. This is an important observation as it eliminates the need for transformations or adjustments in the data prior to modeling.

Additionally, To validate the normality of the numerical features, Q-Q plots were used to compare the quantiles of the data against those of a normal distribution. These plots confirmed that all features closely aligned with normality, further validating the insights from the histograms. For example, the Q-Q plot for "Sweetness" (Figure-4) shows that most data points closely follow the red

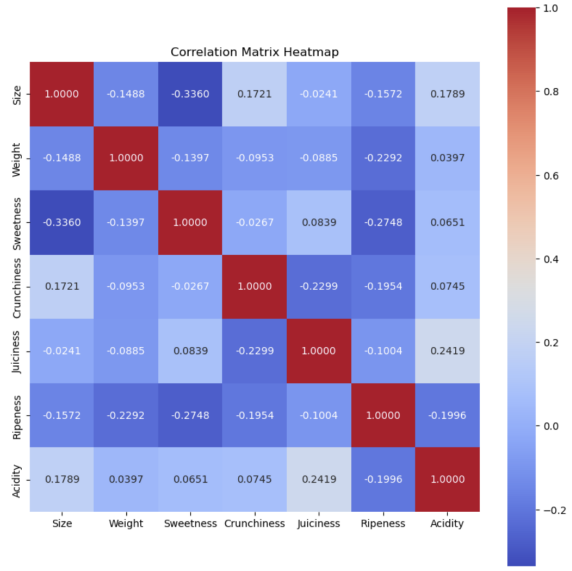

Fig. 3. Correlation Matrix of Features

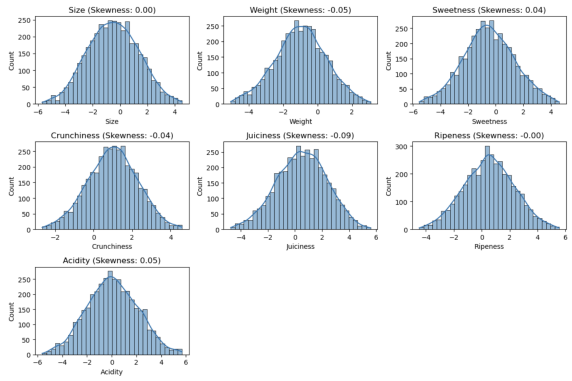

Fig. 4. Histograms of Features after Removing Outliers

line, with only minor deviations at the extremes. The red line denotes a perfect normal distribution. This indicates that the ”Sweetness” feature is approximately normal, with only small deviations at the lower and upper extremes. The Q-Q plots for the other features showed similar results, with the majority of points aligning well with the red line. This consistency across features confirms that the numerical data is largely normal, supporting its use in models that assume normality. The insights gained through these visualizations and analyses provided a comprehensive understanding of the dataset. The absence of strong correlations among features and the balanced distribution of the target variable laid a solid foundation for subsequent modeling efforts. Figures were generated to complement these findings and inform the next steps in model development.

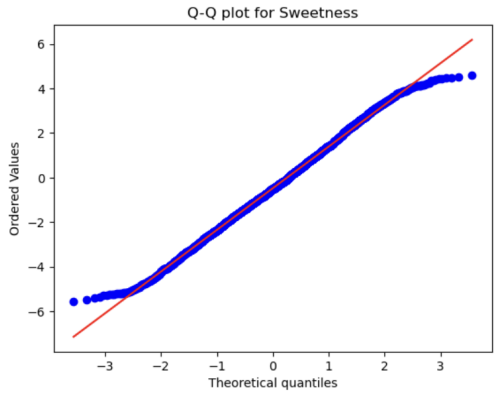

Fig. 5. Q-Q Plot Assessing the Normality of Sweetness

VI. Proposed Methodology

This project seeks to develop an automated system for apple quality assessment using machine learning models, following a structured and detailed workflow. The process began with dataset preprocessing, where outlier removal and conversion of non-numeric features were performed. This ensured that the dataset was clean and ready for modeling. The data was then split into training and testing subsets, following an 80:20 ratio to enable effective model evaluation. Exploratory Data Analysis (EDA) was conducted to analyze trends, feature distributions, and compute correlations, allowing for the identification of the most important features for use in the models.

Finding the most suitable features to include in the prediction models was the main goal of feature selection. Each feature in the dataset contributed equally to the analysis. Therefore, all features were kept except for a_id, which was found to be insignificant. This approach ensured that each retained feature provided distinct and valuable information for the models. After the dataset was successfully prepared and key features were identified, the next stage involved developing machine learning models to evaluate apple quality. For this step, three models—Random Forest (RF), Support Vector Machine (SVM), and Multilayer Perceptron (MLP)—were developed and evaluated.

The Random Forest (RF) model was chosen because it can handle a variety of features and uses ensemble learning to improve accuracy and minimize over-fitting. For reference, random forest is an ensemble learning algorithm that works by creating a forest of decision trees (models), where each tree is trained independently on random samples of the data, to produce more accurate predictions. This technique uses sampling with replacement. What is unique about random forest is that it selects the best split from a random subset of features when splitting a node, adding diversity to the trees and making the model more generalizable. To maximize performance, hyper-parameters like the depth and number of trees were adjusted.

The Support Vector Machine (SVM) model was put into use because of its performance in earlier studies and appropriateness for classification tasks. For context, a SVM is a supervised machine learning model often used for classification tasks. It works by finding the optimal hyperplane that separates data points from different classes. The model's parameters, including kernel type, regularization, and gamma, were improved through grid search and using the Radial Basis Function (RBF) kernel, which handles non-linearly separable data well.

Lastly, the Multilayer Perceptron (MLP) model was implemented due to its ability to capture complex, non-linear relationships in the data through its neural network structure. Multi-layer Perceptron (MLP) is an Artificial Neural Network that is used to develop models for classification and pattern recognition [3]. MLP in particular has an input layer, an output layer, and a hidden layer in between. It uses feedforward passes and backpropagation to adjust the weights of the nodes. This model was created to see if we could achieve a higher accuracy compared to the SVM and Random Forest models. Hyperparameters such as learning rate, number of hidden layers, and activation functions were adjusted using grid search.

The next step was testing the models carefully to ensure accurate predictions on apple quality. Cross-validation was implemented to check how well the models performed on unknown data, and revealed whether the models were learning patterns or simply memorizing the training data.

$$Accuracy : \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision : \frac{TP}{TP + FP} \tag{2}$$

$$Recall(TPR) : \frac{TP}{TP + FN} \tag{3}$$

$$F1 : \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{4}$$

$$FPR : \frac{FP}{FP + TN} \tag{5}$$

Additionally, several key performance metrics were analyzed for each model, as shown in the equations above. Accuracy (Equation 1) measured how often the model made correct predictions overall, offering a general assessment of performance. Although accuracy is highly important, it can be misleading due to its lack of error type identification. Precision (Equation 2) revealed the proportion of true positive predictions among all positive predictions made by the model, highlighting how reliable the model was in labeling instances as positive. A high precision result indicated a reliable model with few false positives. Recall (Equation 3) signaled how many of the actual positive cases the model successfully identified, revealing its sensitivity to true positives. The model's ability to capture a large number of positive cases is reflected by a high recall outcome. The F1-score (Equation 4) combines precision and recall to provide a balanced measure of performance, considering both precision and recall simultaneously. Finally, the ROC plot, a visual representation that compares the true positive rate (TPR, also known as recall) and false positive rate (FPR) was used. The FPR is the proportion of false positive cases over all the true positive cases. The ROC plot displays area under the curve (AUC), which is useful for comparing model performance. A higher AUC value indicates that the model has better performance as it has a high rate of identifying true positive cases and minimizing false positive cases.

VII. Experimental Results and Evaluation

A. Random Forest (RF) Model

At first, the Random Forest model had achieved an accuracy of 87.6%, a F1-score of 87.66%, a precision score of 85.86%, and a recall score of 89.54%. To improve the model's performance, grid search was implemented. The following hyperparameters were used: max_depth (the maximum depth of the tree), max_features (the maximum number of features used to determine the best split when building a tree), and n_estimators (the number of trees in the forest). From the grid search, the model with the best hyperparameters had a max_depth of 20, max_features of 'sqrt', and n_estimators of 400. With these hyperparameters, the model produced an accuracy score of 88.65%, F-1 score of 88.68%, precision score of 87.08%, and a recall score of 90.35%. Hyperparameter tuning helped improve accuracy, F-1 score, and recall score.

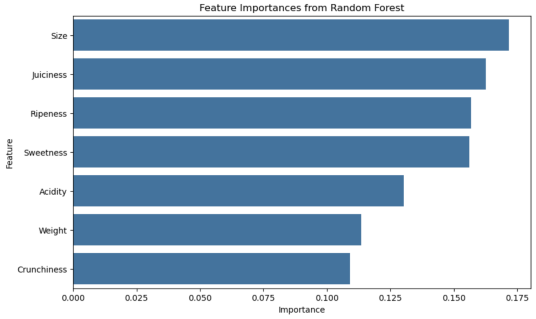

Fig. 6. Bar plot of Random Forest feature importances

Additionally, we used the Random Forest model to plot the feature importance, as shown in Figure-6. The bar plot shows that Size had the highest importance score and Crunchiness with the lowest importance score. However, when looking at the feature importance scale holistically, there is not a significant difference between the importance scores among all the features, as the range is fairly small (0.11 to 0.17). This supports our decision to keep all of the features in our dataset when building our classification models.

B. Support Vector Machine(SVM) Model

Because our exploratory data analysis showed that the data did not appear to be linearly separable, the Radial Basis Function (RBF) kernel was chosen for this SVM model, as it is effective for non-linearly separable data. Initially, the model achieved an accuracy of 87.60% and F1-score of 88.70%. To optimize the model, a grid search was performed over the following hyperparameters: kernel, C (the regularization parameter), and gamma (controls the influence of each training point on the decision boundary). The grid search led to the following optimized model parameters: C = 10, Gamma = 'auto' (automatically set to inverse of the number of features in our dataset), and Kernel = 'RBF'. After optimization, the model's accuracy improved to 91.42%, and the F1-score increased to 91.34%. Additionally, the model achieved a precision of 90.74% and a recall of 91.96%.

After optimization, the false positives decreased, and the false negatives increased, and the F1 score still increased by 0.0264. In apple quality classification, false positives (classifying the apples as 'good' quality when they are actually 'bad') are more harmful than false negatives. This is because selling or consuming 'bad' quality apples poses a greater risk to consumers and sellers than failing to classify some apples as 'good'. Thus, it is acceptable to tolerate the tradeoff of lowering false positives and increasing false negatives.

C. Multi-layer Perceptron (MLP) Model

After splitting the dataset into training and testing sets, we first determined the baseline accuracy of the MLP classifier without any hyperparameter tuning, which was an accuracy of 91%. Although this accuracy is already high, to further improve this model, we tuned various hyperparameters such as the activation functions, learning rate, solver (used for optimization), maximum iterations, and number of hidden layers.

After manual search, the next step was to run a grid search to determine which combination of hyperparameters resulted in the highest accuracy rate. Because the computational time for grid search increased as more hyperparameters were used, the optimization process for this model took significantly longer than the SVM or Random Forest mod-

els. The model eventually achieved an accuracy of 93.9%. This was achieved using the following parameters: the tanh activation function, SGD solver, an adaptive learning rate with an initial learning rate of 0.001, a maximum of 2000 iterations, and hidden layers with sizes 32, 64, 128.

After experimenting with the number of iterations, we observed that it took nearly 2000 iterations for the SGD solver to converge, which led to the decision to set the maximum iterations to 2000. Through grid search, another observation was that the SGD solver performed better than the Adam solver. Additionally, we found that the 'adaptive' learning rate with a value of 0.001 performed the best. An adaptive learning rate dynamically adjusts the learning rate during model training. For hidden layers, two configurations were used: [16, 32, 128], and [32, 64, 128]. The grid search showed that the latter performed better.

Overall, the MLP model achieved an accuracy of 93.93%, F1 score of 93.88%, precision of 93.14%, and recall of 94.64%.

D. Evaluation of All 3 Models

Overall, looking at all three models we can see that all of them have relatively high metric scores as seen in Figure-7. Firstly, Random Forest model achieved an accuracy of 88.65%, F-1 score of 88.68%, precision score of 87.08%, and recall score of 90.35%. The SVM model achieved an accuracy of 91.42%, F-1 score of 91.34%, precision score of 90.74%, and recall score of 91.96%. Finally, the MLP model achieved an accuracy of 93.93%, F-1 score of 93.88%, precision score of 93.14%, and recall score of 94.64%.

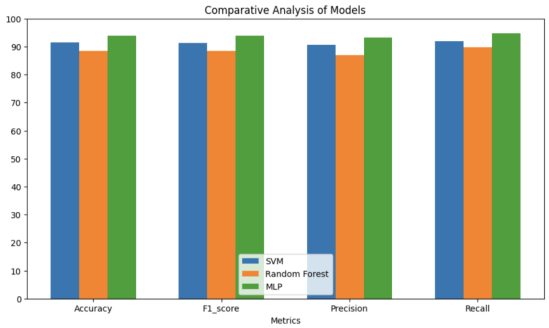

Fig. 7. Evaluation Metrics Comparative Bar Plot

In terms of accuracy, the MLP model had the highest accuracy (93.93%) among the three models. The SVM model had the second highest accuracy (91.42%), and the Random Forest model had the lowest accuracy (88.65%) out of the three models. This suggests that the MLP model performed the best in predicting true 'good' and 'bad' quality apples correctly.

For the F-1 scores, the MLP model had the highest F-1 score (93.88%) among the three models. The SVM model had the second highest F-1 score (91.34%), and the Random Forest model had the lowest F-1 score (88.68%) out of the three models. This suggests that the MLP model performed the best in identifying true 'good' quality apples correctly while minimizing false positives (apple is predicted as 'good' when the apple is actually 'bad').

Regarding the precision score, the MLP model had the highest precision score (93.14%) among the three models. The SVM model had the second highest precision (90.74%), and the Random Forest model had the lowest precision score (87.08%) out of the three models. This suggests that the MLP model has the best accuracy in its positive predictions. In other words, the proportion of correctly predicted 'good' quality apples out of all the predicted 'good' apples is high.

For the recall scores, the MLP model had the highest recall score (94.64%) among the three models. The SVM model had the second highest recall (91.96%), and the Random Forest model had the lowest recall (90.35%) out of the three models. This suggests that the MLP model performed the best in identifying true 'good' quality apples correctly while minimizing false negatives (apple is predicted as 'bad' quality when the apple is actually 'good' quality).

Furthermore, in the ROC plot in Figure-8, the Random Forest model had an area under the curve (AUC) value of 0.95, SVM had 0.96, and MLP had 0.98. MLP had the largest AUC value, indicating that the model performed better at identifying actual "good" quality apples while minimizing false positives. Because the MLP model consistently outperformed the SVM and Random Forest model in terms of accuracy, F-1, precision, recall, and AUC value, we choose MLP to be our optimal

model.

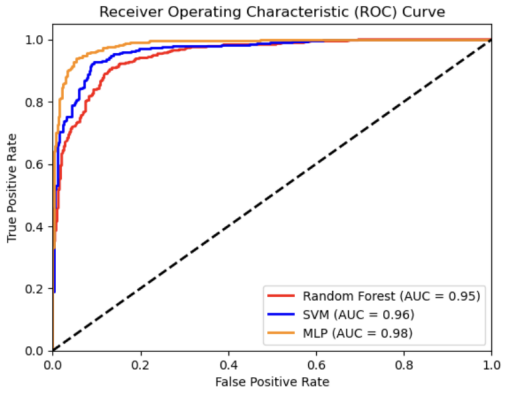

Fig. 8. ROC Comparison Plot

VIII. Conclusion and Discussion

This research demonstrated the potential of machine learning models in automating the assessment of apple quality, a task traditionally reliant on subjective and time-consuming visual inspections. Using the dataset of 4,000 apple samples, we applied thorough pre-processing techniques, including normalization, outlier removal, and dataset splitting. The exploratory data analysis confirmed the absence of strong correlations among features, ensuring that each attribute contributed unique information to the models. Additionally, the balanced distribution of the target variable, "Quality," supported robust model training and evaluation. The machine learning models implemented in this study—Multilayer Perceptron (MLP), Support Vector Machines (SVM), and Random Forest (RF)—provided valuable insights into their effectiveness for apple quality classification. The MLP model, inspired by previous research, demonstrated its ability to capture complex patterns in the data. With hyperparameter tuning, this model achieved the highest accuracy among the models tested. Similarly, the SVM model, optimized using grid search, showcased its effectiveness in handling nonlinearly separable data, while RF provided a reliable baseline due to its ensemble learning approach. Our findings indicate that the use of machine learning models can significantly improve the speed and accuracy of apple quality assessment. The models' performance highlights their potential to replace traditional methods, ensuring more consistent and objective quality evaluations. However, the study also revealed some limitations. For instance, while the dataset's normalization and balance were strengths, the features themselves showed weak correlations, which may have limited the models' predictive capabilities. Future work could explore the integration of additional features, such as environmental or storage conditions, to further enhance model performance. In conclusion, this study underscores the transformative impact of machine learning in agricultural quality assessment. By adopting automated systems like the ones developed in this project, stakeholders in the apple supply chain, including farmers and retailers, can ensure higher consumer satisfaction and operational efficiency. Moving forward, implementing these models in real-world settings and expanding their applications to other types of produce could further revolutionize the agricultural industry.

IX. Github Link

https://github.com/aratakon/ecs171_Project

X. Project Roadmap

Deadline	**Task**
10/18	Data Cleaning, Data Imputation, Removing outliers.
10/23	EDA: create visualizations like correlation plots, histograms, and Q-Q plots.
10/25	Perform feature selection. Finish literature review and select three models to develop: MLP, SVM, and RF.
10/28	Begin development of MLP, SVM, and RF models.
10/30	Complete and submit mid-progress report.
11/08	Finish training models and evaluate models on test data.
11/22	Finish model development and select best performing model after a comparative evaluation.
12/02	Complete final report rough draft. Develop HTML website.
12/09	Finalize project report. Ensure code is working.
12/10	Submit project report, demo, and source code.

XI. Member Contributions

Member	Contribution
Ananya	Literature review; Developed, optimized and evaluated the MLP model; Demo video; CSS; Lead meetings.
Vibha	Worked on EDA; Created, evaluated, and wrote about SVM model; Project road map; Worked on HTML website.
Marina	Worked on preprocessing including: normalization, outlier removal, and handling non numeric features; Worked on Introduction and Proposed Methodologies in final report.
Maithreyi	Worked on EDA and created necessary plots; Wrote Dataset Description and EDA sections in the final report.
Mandy	Worked on EDA; Created evaluated, and wrote about random forest model; Made ROC comparison plot and wrote about the evaluation of the models.

References

[1] Meshram, V., Patil, K., Meshram, V., Hanchate, D., & Ramkteke, S. D. (2021). Machine learning in agriculture domain: A state-of-art survey. Artificial Intelligence in the Life Sciences, 1, 100010. https://doi.org/10.1016/j.ailsci.2021.100010

[2] Singh, S., Singh, N.P. (2019). Machine Learning-Based Classification of Good and Rotten Apple. In: Khare, A., Tiwary, U., Sethi, I., Singh, N. (eds) Recent Trends in Communication, Computing, and Electronics. Lecture Notes in Electrical Engineering, vol 524. Springer, Singapore. https://doi.org/10.1007/978-981-13-2685-1_36

[3] Cengel, T.A., Gencturk, B., Yasin, E.T. et al. Apple (Malus domestica) Quality Evaluation Based on Analysis of Features Using Machine Learning Techniques. Applied Fruit Science (2024). https://doi.org/10.1007/s10341-024-01196-4