Project Deliverable 2

Group Members: John Paul Minimo

Topic: Model to predict apple quality based on sweetness, crunchiness, and juiciness

Dataset Apple Quality (kaggle.com)

Evaluation Metrics

To effectively evaluate our machine learning models in predicting apple quality, several metrics are considered essential:

Accuracy: This metric is useful given our nearly balanced dataset. However, it does not always provide a complete picture, especially with imbalanced classes.

Precision: This is crucial since we are attempting to minimize false positives. In our project, we aim to avoid classifying bad apples as good. As bad apples are considered a loss.

Baseline Model - Logistic Regression

For our baseline model, I opted to go for logistic regression. The results for this model were:

Accuracy: 0.730

Precision: 0.768

The dataset I used had the classes as 50-50, and were balanced, so I opted to use the most-frequent-class baseline.

Methodology

Preprocessing/Cleaning - The dataset was removed of any errors/null entries before any type of processing/training began.

Data Splitting: The dataset was split into training and test sets with a 70/30 ratio, ensuring sufficient data for both training and evaluating the models. Our dataset included 4000 entries, so the classic 70/30 ratio seemed like a good option.

Cross-validation: For each of the models, I used 5-fold cross-validation during hyperparameter tuning to ensure that the models were generalizing well over different parts of the data.

Hyperparameter Tuning: Used GridSearchCV to optimize parameters for each model, focusing on maximizing the F1 score and precision.

Models and Hyperparameters Tested:

Logistic Regression: Evaluated with penalties (L1 and L2) and varying strengths of regularization (C values were the same as the ones tested in our gridsearch homework).

Random Forest: Tuned for number of trees, maximum features, maximum depth, and minimum samples split.

Artificial Neural Network (ANN): Configured with different numbers of neurons and layers, using ReLU for hidden layers and sigmoid for the output layer.

Results

The performance of various models is outlined below, with more information that could be found inside of the notebook.

Model	Precision	Accuracy	Relative to Baseline (Accuracy)
Logistic Regression	0.73	0.77	+0%
Random Forest	0.89	0.89	+15%
ANN	0.73	0.74	-4%

Best Model

Random Forest outperformed all of the other models and significantly surpassed the baseline’s metrics, seeing a 15% gain in accuracy when compared to our baseline model.

Comparison with Literature

The models in literature typically used datasets that consisted of images, and computer vision methods were used. If we were to compare our results with those models, it would be unfair, due to the difference in nature of the datasets in the first place. However, if we were to ignorantly look at the results alone and pretend they were the same, the random forest model is very competitive, but not reaching the level of 99% accuracy that some of the papers had reached.

However, comparing it to other models in the kaggle discussion boards, my model seems to perform well, exceeding quite a bit of the models/code posted using the same dataset.

Discussion

I believe hyperparameter tuning really helped when creating a model that would perform well. I believe one of the reasons as to why ANN did not work as well is due to the fact that I was not able to use GridSearchCV/truly test different parameters for ANN.

I also believe that I could’ve done a better job with feature engineering, maybe adding a section excluding different sections of the dataset could help our final results.

Conclusion

This project allowed me to learn more about the process of what a real data scientist would undergo in the real world. It also taught me how I should start working from scratch, performing research on a problem, reading different solutions that had already been attempted or have succeeded, and attempting to improve on those attempts.

This project also showed me that performing these tasks/training models requires critical thinking and research, and to truly succeed you should have attempt to have a deep understanding in order to know what went wrong and what went right.

As for improving my project, I think I could have done a better job on feature engineering, maybe removing some unneeded and highly correlated features to make it more efficient,, Also during writing this document, I feel that normalizing the data could have helped,

Project Deliverable 1 Below:

Problem Description: Determining Apple Quality

For agricultural companies, determining apple quality is vital in order to have good apple sales. A bad quality apple is determined through many different factors, such as color, texture, flavor, and whether or not it is rotten/spoiled. Not only do these aesthetic features matter, but also can provide insight to the apple's nutritional value (bad quality could be a good indicator of being less nutritional), and whether there is a disease affecting that certain type of apple. Selling bad apples to the public could reduce a company's reputation to the customer/distributor, so selling good quality apples is a must.

Evaluation Metrics

How is apple quality found without machine learning?

Currently, without machine learning, agricultural companies must use manual labor in order to detect whether or not apple quality is bad. This manual inspection process can be tedious and sometimes erroneous especially if we are dealing at such a large scale of apples.

Why Machine Learning?

By using machine learning, we can provide a more elegant solution to agricultural companies. Currently, the process of determining apple quality is destructive, hence the many studies to attempt and remedy this problem. By using machine learning, this allows companies to possibly reduce the amount of labor needed when determining apple quality, while also increasing the rate of apples being checked at a given time and giving a non-destructive option to determining quality..

High Level Approach

The first step to solving this problem with machine learning is to first conduct some data exploration. Cleaning the data and performing some preprocessing will ensure that no invalid records are being fed into training the model. The next step would be to complete some feature engineering. By doing some prior research on which information might be unimportant, we can reduce the dimensionality and increase the accuracy of our model, possibly making it perform efficiently and more accurately, when compared to just throwing all possible valid features at our model. Next, would be to test out different possible classification models, should we be using KNN, random forest, etc?

We can use gridsearch to automate the testing of these models and hyperparameters. Then we can compare them amongst each other, we can use accuracy between training and test sets to determining for overfitting. Next we can use f1 score, and accuracy, and a confusion matrix to understand if our model is actually decent at predictions. For this type of problem, we would want to have less type 1 errors, as this means that our model is letting bad apples pass. Different models might have different performance in this metric, so we would want a model with good precision when determining bad apples. After performing this analysis, we should have a final model for our problem.

Literature Review

Apple quality identification and classification by image processing based on convolutional neural networks[1]

This study trained a convolutional neural network model to determine the apple quality using pictures of the apple. For their dataset, they gathered pictures of apples each with varying conditions in order to show different possible conditions, each either being considered, premium, or poor. Their model had achieved an overall accuracy of 95.33% when running against an independent testing dataset.

Using hyperspectral imaging technology and machine learning algorithms for assessing internal quality parameters of apple fruits[2]

This study uses hyperspectral imaging in order to build their dataset of Pink Lady apples. By taking images of different apples at different stages, they were able to build their dataset of 300. After testing ANN, KNN, Decision trees, they were able to come to the conclusion that for their mode of data, ANN is the most effective, with an R^2 value of 0.91 for their ANN model.

Vision-based apple quality grading with multi-view spatial network[3]

The dataset for the study was obtained using a unique scheme with multiple fixed cameras capturing different views of apples. Lightweight Convolutional Neural Networks were used for feature extraction. This Multi-View Spatial Network achieved a classification accuracy of 99.23%.

Literary Analysis Conclusion

One important thing I realized after completing this analysis is how little research is completed on datasets that involve a tabular mode of data. Most if not all of the research I found is completed using visual data which involve complex imaging techniques. What I will do differently with my project is train a model using tabular data, as some companies or small farmers may not have the access to this type of techniques.

Dataset Discussion: Apple Quality (kaggle.com)

The source of this dataset was retrieved from an agricultural company. The features include size, weight, sweetness, crunchiness, juiciness, ripeness, acidity, and quality, with 4001 records. The distribution of classes is nearly 50/50 with 2004 instances of good apples (50.2%), and 1996 apples being labeled bad (49.8%).

References

[1] Apple quality identification and classification by image processing based on convolutional neural networks - PMC (nih.gov)

[2] Using hyperspectral imaging technology and machine learning algorithms for assessing internal quality parameters of apple fruits - ScienceDirect

[3] Vision-based apple quality grading with multi-view spatial network - ScienceDirect