Dataset 1: Apple Quality
Link: https://www.kaggle.com/datasets/nelgiriyewithana/apple-quality

Dataset 2: Sales Records (100K)
Link: https://excelbianalytics.com/wp/downloads-18-sample-csv-files-data-sets-for-testing-sales/

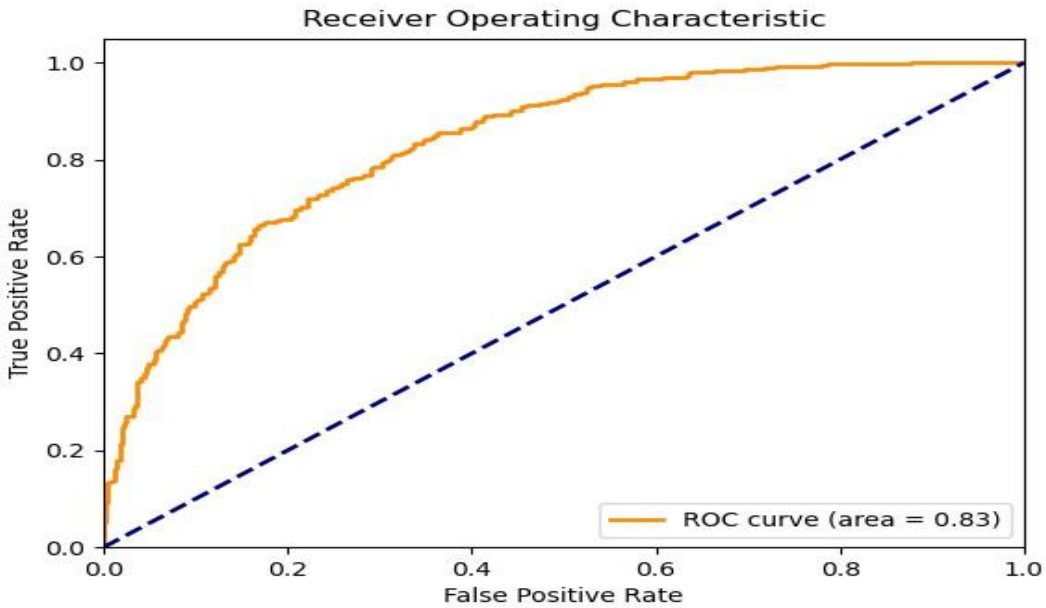
In machine learning data is considered a fundamental component that fuels the algorithms that learn from it, enabling them to make predictions or decisions. Consequently, access to the right data is critical for these processes to occur. There are two main avenues for acquiring data: a) utilizing pre-existing datasets, or b) gathering data specifically for building a model. In this discussion we will examine two datasets. The first is a dataset, hosted on Kaggle, which includes several characteristics of apples such as size, weight, and sweetness. The second dataset is a synthetically generated data on global sales records. We will examine the features of each dataset, identify which ML algorithms would be appropriate, and assess their strengths and weaknesses in relation to the datasets.

Dataset 1 – Apple Quality

For the first dataset, *Apple Quality*, it is hypothesized that the included features would allow for one to categorize the quality of the apple as 'good' or 'bad'. Given that we are expected to categorize, group, or label a datapoint, it means we will be using a classifier algorithm. The cardinality value of two for the predictor column could make this dataset a good candidate for Binary Logistic Regression. Alternatively, we can opt to use a K-nearest Neighbors approach. Both algorithms are evaluation using confusion matrix metrics.

1. **Binary Logistic Regression (Classification)**
   This method is appropriate for several reasons. For starters, the class is balanced, with enough datapoints in each category. The predictor 'Quality' has 2004 counts for 'good' and 1996 counts for 'bad'. Each predictor was found to be normally distributed and exhibited low VIF, making them suitable candidates for regression analysis. Furthermore, we have the advantage of having fewer than ten predictors, all numeric type. Logistic Regression is very sensitive to outlier points; therefore, boxplots were used to check for any occurrences and appropriate measures were taken.

   Model Results – 74.62% Accuracy, AUC = 0.83

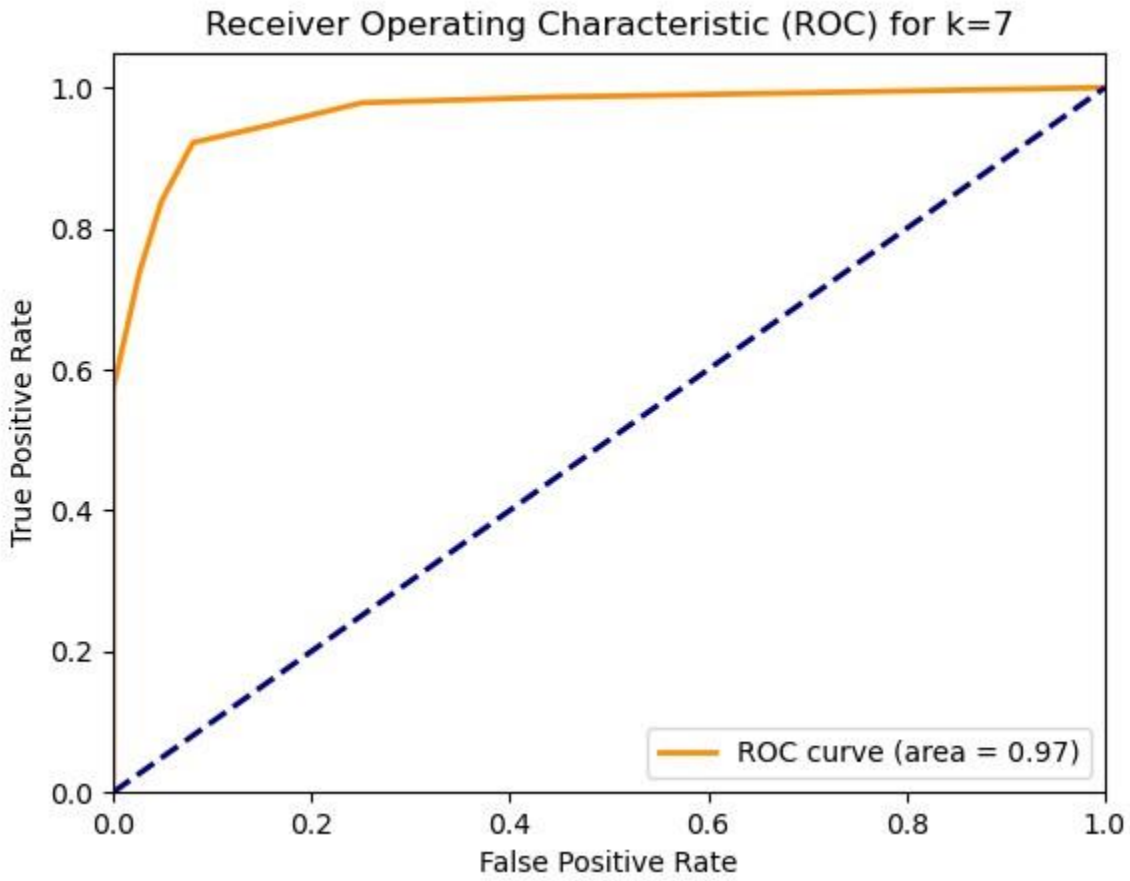


	precision	recall	f1-score	support
0	0.758270	0.733990	0.745932	406.00000
1	0.734644	0.758883	0.746567	394.00000
accuracy	0.746250	0.746250	0.746250	0.74625
macro avg	0.746457	0.746437	0.746250	800.00000
weighted avg	0.746634	0.746250	0.746245	800.00000

2. K-Nearest Neighbors (Classification)

Using K-Nearest Neighbors algorithm is a good choice due to its simplicity. It's a lightweight algorithm and, just like Logistic Regression, its easily interpretable. As mentioned before, the class is balanced, which makes it a good candidate for KNN. A weakness of KNN is its slow compute performance with too many datapoints; therefore, our dataset of 4000 observations is suitable. Moreover, the preprocessed dataset without outliers was used as k-NN is unable to handle it on its own.

Model Results – Accuracy 92.06%, AUC= 0.96.65



k	Accuracy	F1 Score	Sensitivity	AUC
1	0.892446	0.892431	0.884097	0.892049
2	0.879641	0.878695	0.803235	0.932736
3	0.896287	0.896293	0.892183	0.944974
4	0.893726	0.893271	0.840970	0.955069
5	0.907810	0.907784	0.897574	0.960016
6	0.902689	0.902473	0.867925	0.963464
7	0.920615	0.920633	0.921833	0.966564
8	0.918054	0.917935	0.892183	0.967869
9	0.919334	0.919357	0.921833	0.966606
10	0.907810	0.907718	0.886792	0.966120

Verdict: Although both algorithms are easy to interpret, k-Nearest Neighbors did a much better job at classifying good and bad apples. In a business case scenario KNN could be the best approach.

Dataset 2 – Sales Records

For the second dataset, *Sales Records*, several approaches can be taken. First, linear regression can be applied to predict numeric values on several columns, those being 'Total Revenue', 'Total Profit', and 'Units Sold'. Alternatively, we can employ a classification model to predict 'Category' or 'Order Priority', both of which would imply the use of a multinomial classification approach. For the sake of remaining inline with the previously selected algorithms, we will evaluate the use of a Multiple Linear Regression Model for 'Units Sold', and KNN for 'Order Priority'.

3. **Multinomial Logistic Regression (Classification)**
   This algorithm falls primarily within the realm of statistical learning and represents an enhanced version of Binary Logistic Regression. The main distinction lies in the use of multiple classes found in the target variable. This method also assumes parametric conditions where data is normally distributed. In cases where the data is not normally distributed, transformation techniques, such as using logarithmic and power transformations, can be used to normalize the data. Boxplots have revealed potential outliers that should be addressed unless they are inherent characteristics of the data. In the process of evaluating for collinearity, several inconsistencies were identified; 'Unit Price' and 'Unit Cost' exhibit serial correlation, as are 'Total Revenue', 'Total Cost', and 'Total Profit'.

   Model Results – 25.16% Accuracy

	precision	recall	f1-score	support
C	0.243541	0.350325	0.287332	4924.00000
H	0.249163	0.149789	0.187099	4967.00000
L	0.249646	0.246899	0.248265	4998.00000
M	0.266640	0.260223	0.263392	5111.00000
accuracy	0.251650	0.251650	0.251650	0.25165
macro avg	0.252247	0.251809	0.246522	20000.00000
weighted avg	0.252366	0.251650	0.246559	20000.00000

4. **K-nearest Neighbors (Classification)**
   The Sales Records dataset contains 100 thousand observations, all of which are non-null. Given the distributions found in the dataset, no transformation would need to be done because KNN is not sensitive to parametric assumptions. However, outlier points will need to be addressed. Lastly, this dataset would need to scaled for KNN to perform as intended.

   Model Results – 27.25% Accuracy

	precision	recall	f1-score	support
C	0.270852	0.360669	0.309373	5024.00000
H	0.271361	0.300100	0.285008	4995.00000
L	0.273924	0.245944	0.259181	4993.00000
M	0.276113	0.182839	0.219998	4988.00000
accuracy	0.272550	0.272550	0.272550	0.27255
macro avg	0.273062	0.272388	0.268390	20000.00000
weighted avg	0.273058	0.272550	0.268467	20000.00000

Verdict: Neither algorithm delivered strong performance. It's possible that alternative algorithms, like decision trees or random forests, might yield better results. Alternatively, it could be that predicting 'Order Priority' is not feasible given the patterns in the data. A more suitable approach might have been to predict a continuous outcome, such as 'Total Revenue' or 'Total Profit'.

Conclusion

Both datasets are drastically different from each other. Because of this, different algorithms were chosen. Whether it's for predicting a numerical value or classifying an observation, each algorithm has its own strengths, weaknesses, and prerequisites. In a business case use, it is up to the Data Scientist and subject matter experts to decide what would be the best choice, contingent on business requirements and goals. Furthermore, as we saw with Dataset 2, there might now a pattern at all to predict an outcome.