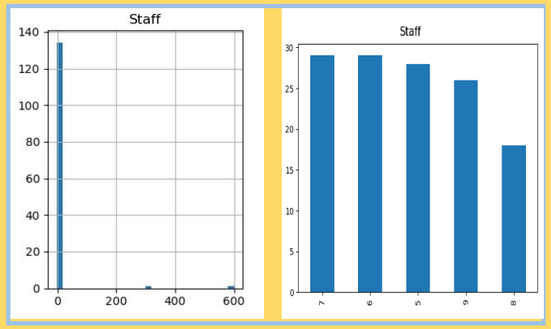
1. ITNPBD6 Assignment 1. Rodolfo Croes

2. Project Methodology

1. Read in the data provided and plot histograms to see where data cleaning needs to be done
2. Select 'Performance' as the target and the other variables as features, then split the data into 30% testing and 70% training. Then continue with step 4-7 on both the testing and training data.
3. Because the data in both the training and test split are fairly balanced, no oversampling or under sampling methods are needed for this project.
4. Clean the data by removing outliers in the variables 'Country', 'Location' and 'Staff'. As well as, remapping some data points in the variable 'Car Park'
5. Removing some features from the data for either having too many unique data points ('Store ID', 'Town' and 'Manager name'), or having too little variance in data points ('Country').
6. Using one-hot encoding, I encoded the Categorical variables of the data. (See the tables in part 3 for which variables were considered Categorical)
7. Using the min max Scaler, I encoded the Numerical Variables of the data. (See the tables in part 3 for which variables were considered Numerical)
8. For each model (Logistic Regression, Decision Tree and Multi-layer Perceptron Neural Network), I fitted them to the training data.
9. Apply 5 folds cross validation to each model on the training data to see which has the best mean cross validation score and change around the hyper parameters via brute force testing.
10. After choosing the best model with the best hyperparameters, I use the test data to plot a confusion matrix of said model.

4. Data Preparation

When analyzing the histograms of the Staff variable, there are a few outliers in the data, those being the stores that are listed to have –2, 300 and 600 staff. To clean this, I chose to keep the stores that had the value of staff as more than 0 and less than 10. The reason for the upper bound is because the maximum staff in a store when excluding the outliers is 9. The histogram for the uncleaned Staff data is the rightmost plot and the cleaned Staff data's histogram can be seen on the leftmost plot.



3. Variables

For all the variables I used to train the models, you can see them on the tables on the right. The reason I chose these variables and discarded the others is because the other variables had too many unique values thus making them less useful to predict the performance of a store. Another reason why I chose to discard some variables is if the majority of the data points in the variable were the same. This then leads to the variable not being helpful to the model when differentiating between good or bad performance.

Variable	Type
Staff	Numerical
Floor Space	Numerical
Car Park	Categorical
Demographic Score	Numerical
Location	Categorical
40min Population	Numerical
30min Population	Numerical

Variable	Type
20min Popu-	Numerical
10min Popu-	Numerical
Store Age	Numerical
Clearance	Numerical
Competition	Numerical
Competition Score	Numerical
Performance	Categorical

5. Model Training and Hyper Parameters

In the table next to this, we can see the models, their hyperparameter and the grid search cv score in %.

For the hyperparameters, I used different values for different variables that can be found on each model's API pages[1][2][3] .Then to find the best values for each I used grid search cross validation which took a dictionary of parameters and values and tested them. Then I called the method to return the best parameters and it's corresponding cross validation score. These methods can be found on the GridSearchCV's API page[4].

Model	Hyper Parameters	Grid Search CV Score(%)
LogisticRegression	'max_iter': 100, 'multi_class': 'auto', 'penalty': 'l2', 'solver': 'lbfgs'	75.7%
DecisionTreeClassifier	'criterion': 'entropy', 'max_depth': 5, 'min_samples_leaf': 10	68.0%
MLPClassifier	'hidden_layer_sizes': (50,50,50,50), 'learning_rate': 'invscaling', 'max_iter': 500, 'solver': 'adam'	78.0%

6. Final Model and Results

The final model that I have chosen is the MLP Neural Network model because it has the best grid search CV score compared to the other models. This model was trained by fitting it to 70% of the total data after cleaning and preprocessing. The other 30% of the cleaned and preprocessed data was used to test this model and produced the following confusion matrix:

Actual/Predicted	Good	Bad
Good	12	5
Bad	4	18

Based off of the confusion matrix above, we can see that this model is better at telling whether a store will perform poorly compared to when a store will perform well. The reason for this is because the true positives to false positive ratio is 75%, whereas the true negative to false negative ratio is 78%, thus showing a better negative prediction pattern compared to a positive prediction pattern.

7. References

[1] Anonymous sklearn.linear_model.LogisticRegression. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html#sklearn.linear_model.LogisticRegression [Accessed: "Mar 25, 2023"].

[2] Anonymous sklearn.neural_network.MLPClassifier. Available: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html#sklearn.neural_network.MLPClassifier [Accessed: "Mar 25, 2023"].

[3] Anonymous sklearn.tree.DecisionTreeClassifier. Available: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html#sklearn.tree.DecisionTreeClassifier [Accessed: "Mar 25, 2023"].

[4] Anonymous sklearn.model_selection.GridSearchCV. Available: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html#examples-using-sklearn-model-selection-gridsearchcv [Accessed: "Mar 30, 2023"].