|  |
| --- |
| CSCU9T6 |
| World of Bargains |
| 2615649 |

|  |
| --- |
| University of Stirling  6/3/2020 |

Contents

[Introduction 0](#_Toc34423101)

[The task 0](#_Toc34423102)

[The techniques 0](#_Toc34423103)

[Why data mining is suitable 0](#_Toc34423104)

[Data pre-processing 1](#_Toc34423105)

[Terminology 1](#_Toc34423106)

[What is pre-processing? 1](#_Toc34423107)

[Country 2](#_Toc34423108)

[Staff attribute 2](#_Toc34423109)

[Carpark attribute 3](#_Toc34423110)

[Location attribute 3](#_Toc34423111)

[10 min population 3](#_Toc34423112)

[Summary 4](#_Toc34423113)

[Attribute analysis 5](#_Toc34423114)

[Town 5](#_Toc34423115)

[Country 5](#_Toc34423116)

[Store ID 5](#_Toc34423117)

[Managers name 5](#_Toc34423118)

[Staff numbers 6](#_Toc34423119)

[Floor space 6](#_Toc34423120)

[Window space 6](#_Toc34423121)

[Carpark 6](#_Toc34423122)

[Demographic score 7](#_Toc34423123)

[Location 7](#_Toc34423124)

[40, 30, 20, 10 min population 8](#_Toc34423125)

[Store age 8](#_Toc34423126)

[Clearance space 8](#_Toc34423127)

[Competition number 9](#_Toc34423128)

[Competition score 9](#_Toc34423129)

[Profit (Target variable) 9](#_Toc34423130)

[Performance (Target variable) 9](#_Toc34423131)

[Modelling 10](#_Toc34423132)

[Prediction 10](#_Toc34423133)

[Multilayer perceptron 10](#_Toc34423134)

[Attribute Selection Analysis 11](#_Toc34423135)

[Specific Model Analysis Terminology 12](#_Toc34423136)

[Best MLP model using 10 folds 13](#_Toc34423137)

[Trialling different fold numbers 13](#_Toc34423138)

[Best MLP result 13](#_Toc34423139)

[Classification 14](#_Toc34423140)

[Decision tree (J48) 14](#_Toc34423141)

[Attribute Selection Analysis 15](#_Toc34423142)

[Specific Model Analysis Terminology 15](#_Toc34423143)

[Best J48 model using 10 fold 16](#_Toc34423144)

[Trialling different fold numbers 16](#_Toc34423145)

[Best decision tree result 16](#_Toc34423146)

[Decision tree ROC curves: Best model vs All attributes, Default settings model 18](#_Toc34423147)

[Recommendations 20](#_Toc34423148)

[Conclusion of results 20](#_Toc34423149)

[Best MLP result 20](#_Toc34423150)

[Best decision tree result 20](#_Toc34423151)

[Recommended technique 20](#_Toc34423152)

[Recommended attributes 21](#_Toc34423153)

**World of bargains assignment**

# Introduction

After being contacted by the owner of a chain of over 100 stores in the UK, my task is to create a prediction model for the owner so that he can enter the details of a shop and receive back the amount of profit it should generate.

## The task

The task is to use the data mining software Weka to build a predictor capable of estimating the revenue that a store should generate based on its values for the variables listed in the data. This should be done using the fewest number of variables possible as it is expensive to gather these values. Also, as the population data is particularly expensive to gather, the owner has specifically asked for a recommendation on if he should continue gather it, or not.

A report will also be written to provide a detailed description of the process followed in finding the best model for predicting the profit and for performance classification, including any decisions made along the way and any recommendations for the owner.

## The techniques

The techniques the owner requested to be used are Multilayer perceptron or a decision tree as his IT director has some experience with these techniques.

The Multilayer perceptron model will be used to estimate the profit that a store is expected to generate. The result of this will allow the owner to assess the performance of his store to see if there can be any improvements made to maximise profit in underperforming stores.

The decision tree will be used to classify the shops in different levels of performance ‘poor’, ‘reasonable’, ‘good’ and ‘excellent’.

## Why data mining is suitable

Firstly, what is data mining? Data mining is the process of searching through large quantities of data to find patterns and trends that would otherwise be undiscoverable by humans/impossible to understand. It uses complex mathematical algorithms to do things like categorise data and predict things like profit of a business, which is what it will be used for in this case. A model will be built using the data provided to us and then be applied by the owner to predict the profit of his stores. The application of the prediction model will allow the owner to potentially make changes in the business in order to maximise profit.

In order to be able to effectively employ data mining, there must be a sufficient **quantity** of data to represent the problem and on this occasion, we do have enough data.

No matter how much data there is, if it is not relevant/**quality** data, it is of no use when data mining as no relevant knowledge will be gained. In this case we do have data that is important to what we are trying to predict so data mining can be used.

# Data pre-processing

## Terminology

**Outlier:** A values that is out with the range of expected/normal values

**Extreme values:** Values that fall far from the range of normal values

**Minority value:** A value that appears infrequently in the data

**Attribute:** A variable or field

**Mean:** The average, the sum of all values divided by the number of values.

**Record/Instance:** Refers to a row of data

**Noise:** Data that needs to be removed due to having meaningless information.

**Correlation:** A mutual relationship or connection between two or more things

The Weka output below shows a graphical representation of each attribute on the unedited dataset.

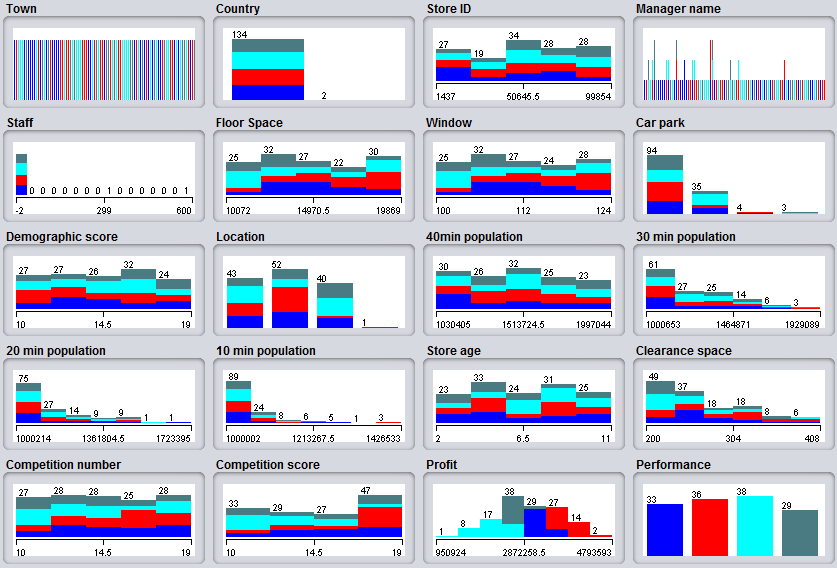


Figure 1 - Distribution of values before pre-processing

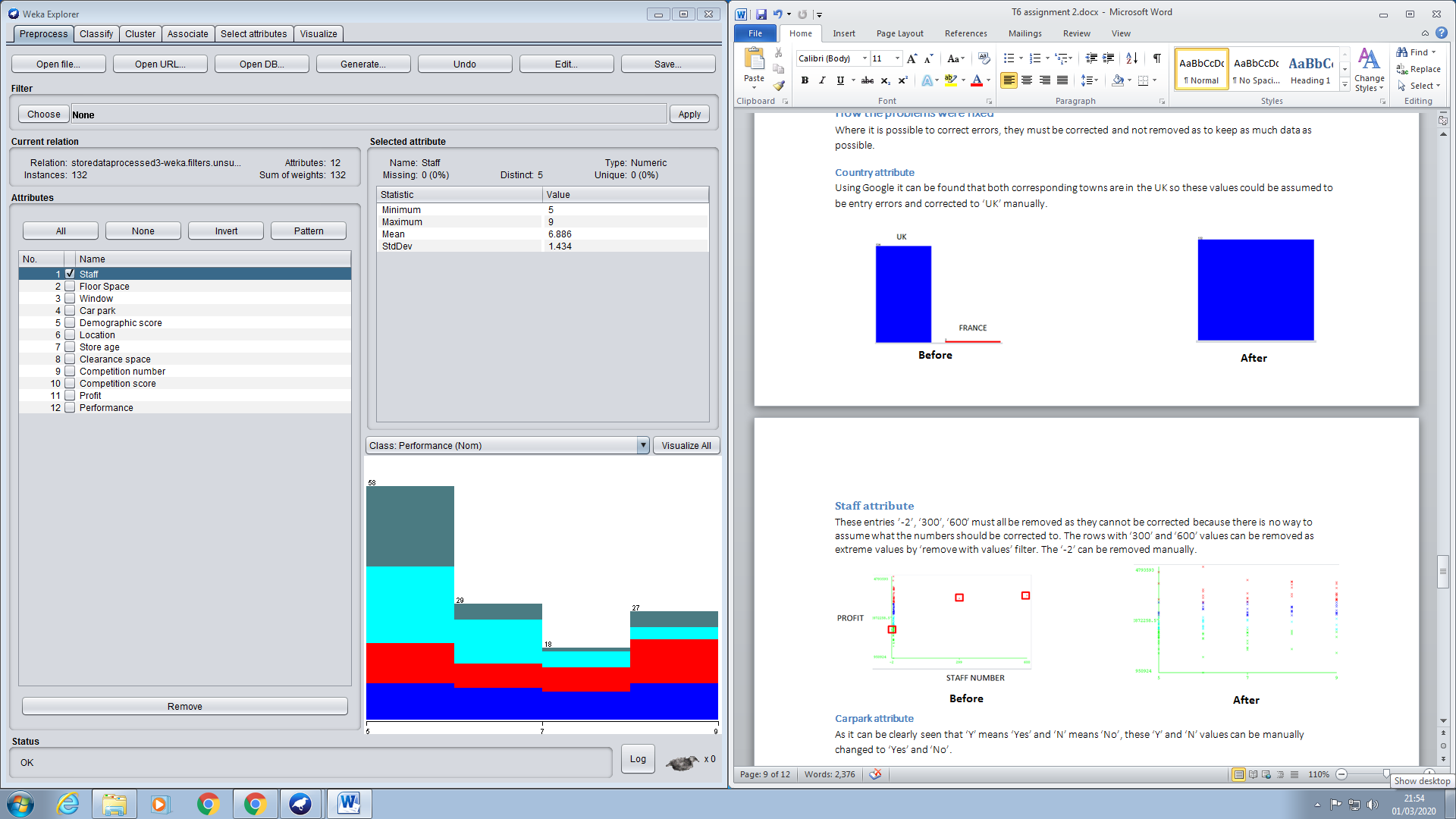
## What is pre-processing?

Data pre-processing is the practice of transforming inconsistent data that may have errors and missing values etc. into something more useable. So for example, when cleansing the data you would deal with any missing values (in this case there is none), deal with noisy data, identify and deal with any outliers and fix any inconsistencies. This must be done because machine learning models give the best results when they have been trained on well prepared data.

As there is a relatively small data set (136 records, rather than thousands), it is easy to manually go through the data and correct any potential errors there may be. It is also possible to find and remove outliers using filters in Weka, so I will be using filters where possible but also will be doing some pre-processing manually in excel.

Also due to the relatively small data set, any potential errors will be corrected where possible and not removed in order to preserve as much data as possible as this could negatively impact the performance of the model.

## Country



UK

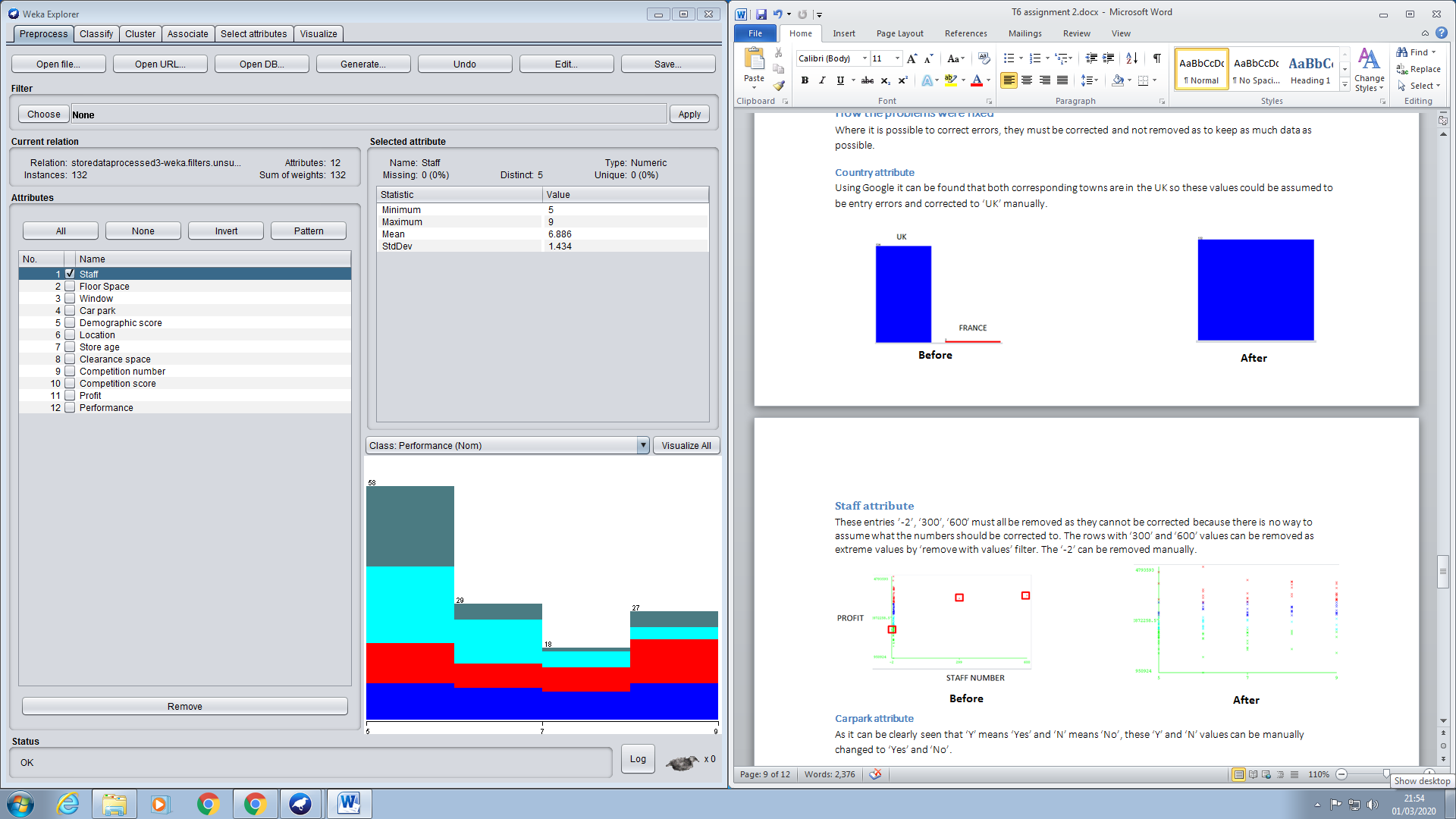
In the Country attribute there are 2 minority values, instances 40 and 96. These 2 values can be shown in the figure to the right. There are 2 instances of ‘France’ and 134 ‘UK’. These will be assumed as potential data entry errors and corrected to ‘UK’ as after searching the corresponding towns, it was discovered that both of these towns are in the UK.

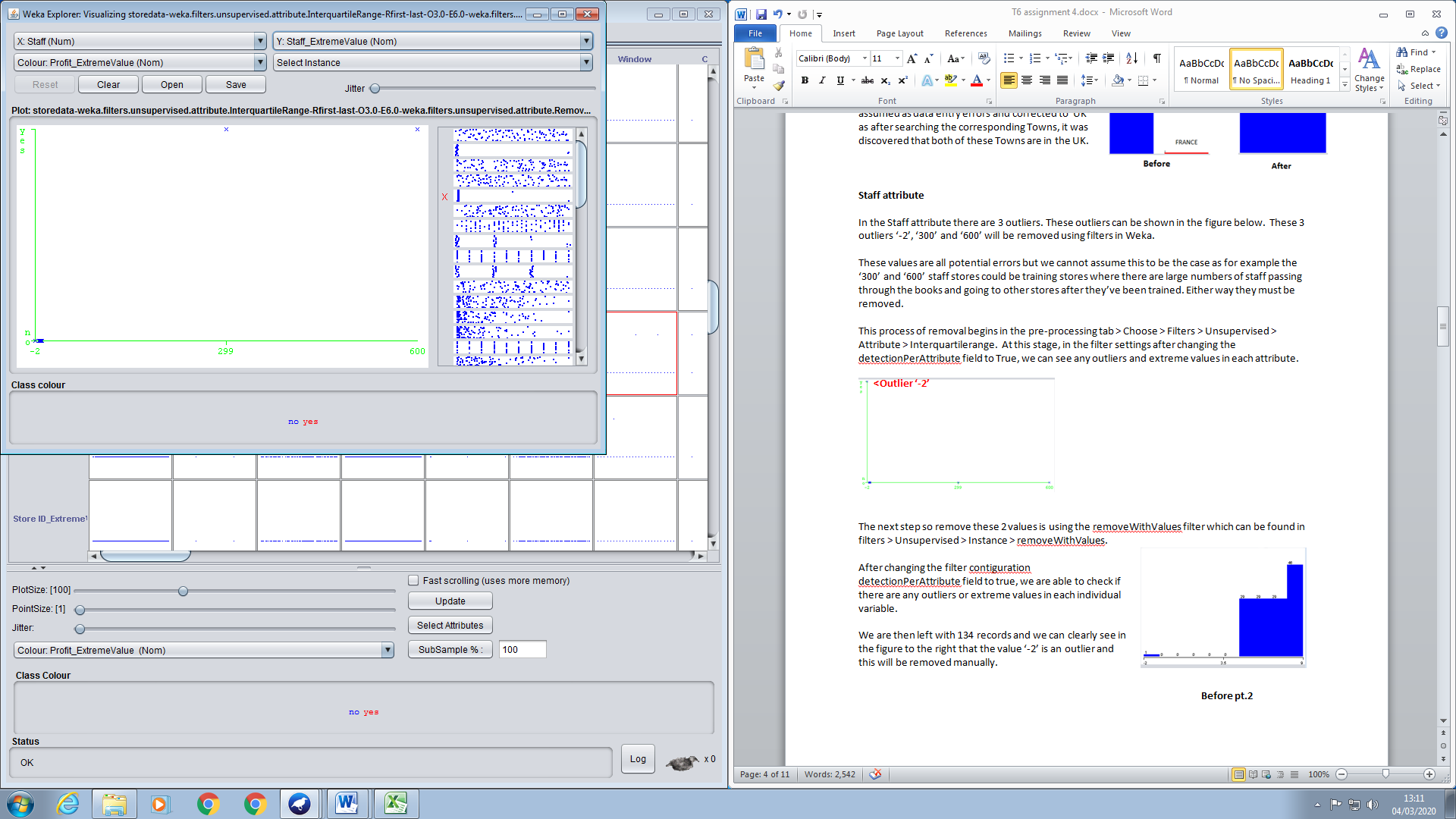
Figure 2 - Country cleansing

## Staff attribute

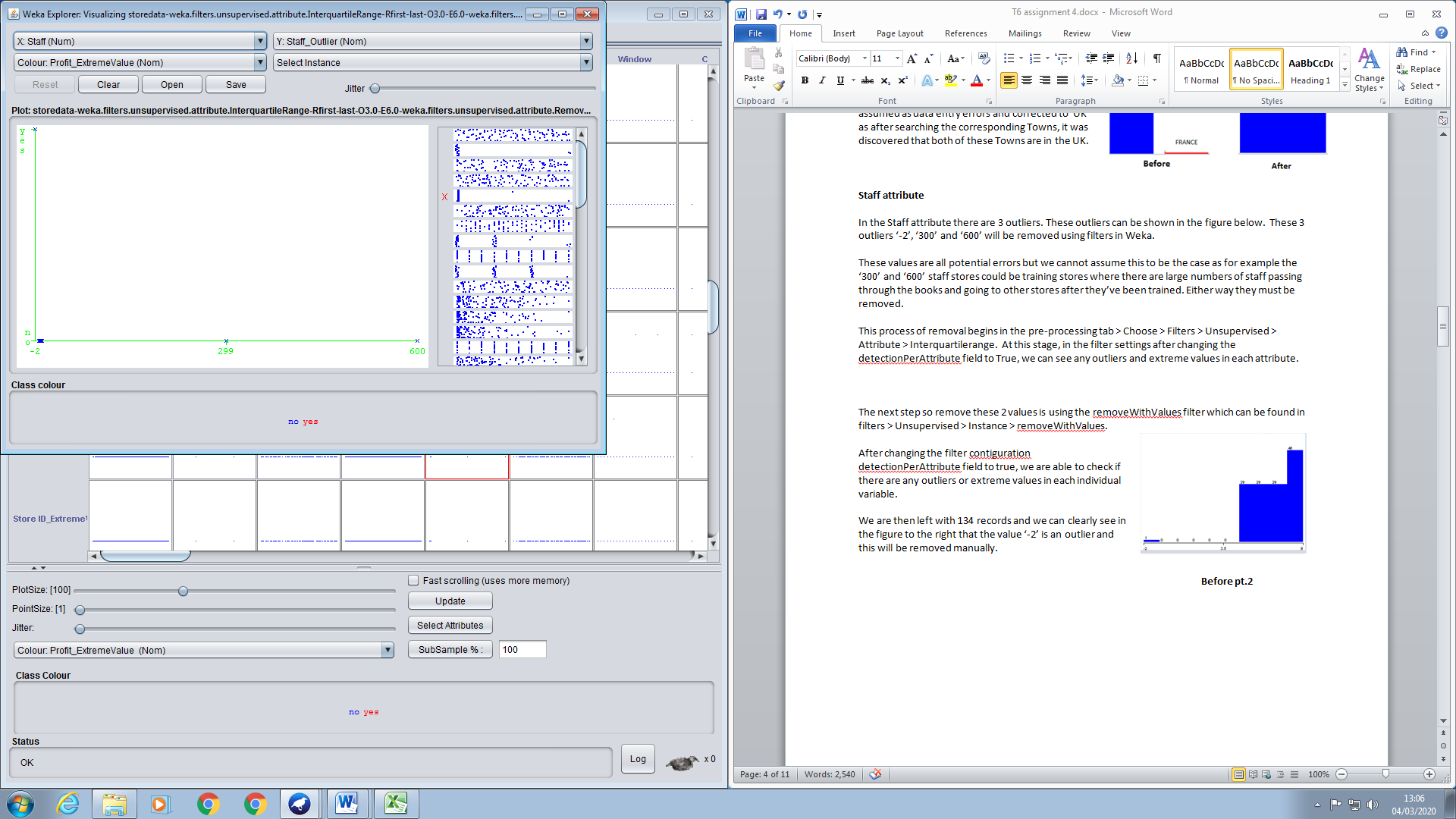
In the Staff attribute there are a potential 3 outliers, instances 3, 54 and 110. These can be shown in the figure below. These 3 values ‘-2’, ‘300’ and ‘600’ will be removed using filters in Weka.

These values are all potential errors but we cannot assume this to be the case as for example, the ‘300’ and ‘600’ staff stores could be training stores where there are large numbers of staff passing through the books and going to other stores after they’ve been trained. Either way they must be removed as they could impact the performance of the model.

This process of removal begins in the pre-processing tab > Choose > Filters > Unsupervised > Attribute > Interquartilerange. At this stage, in the filter settings after changing the detectionPerAttribute field to True, we can see any outliers and extreme values in each attribute.



**<Outlier ‘-2’**



**Extreme values ‘300’ ‘600’ ^**

Figure 3 - Discovering outliers in Staff

After identifying these values we can remove them using the SubsetByExpression filter and inputting the expression ‘(!(ATT5 <5)) and (!(ATT5 >9))’. This range is chosen as it removes all outliers from the Staff attribute.

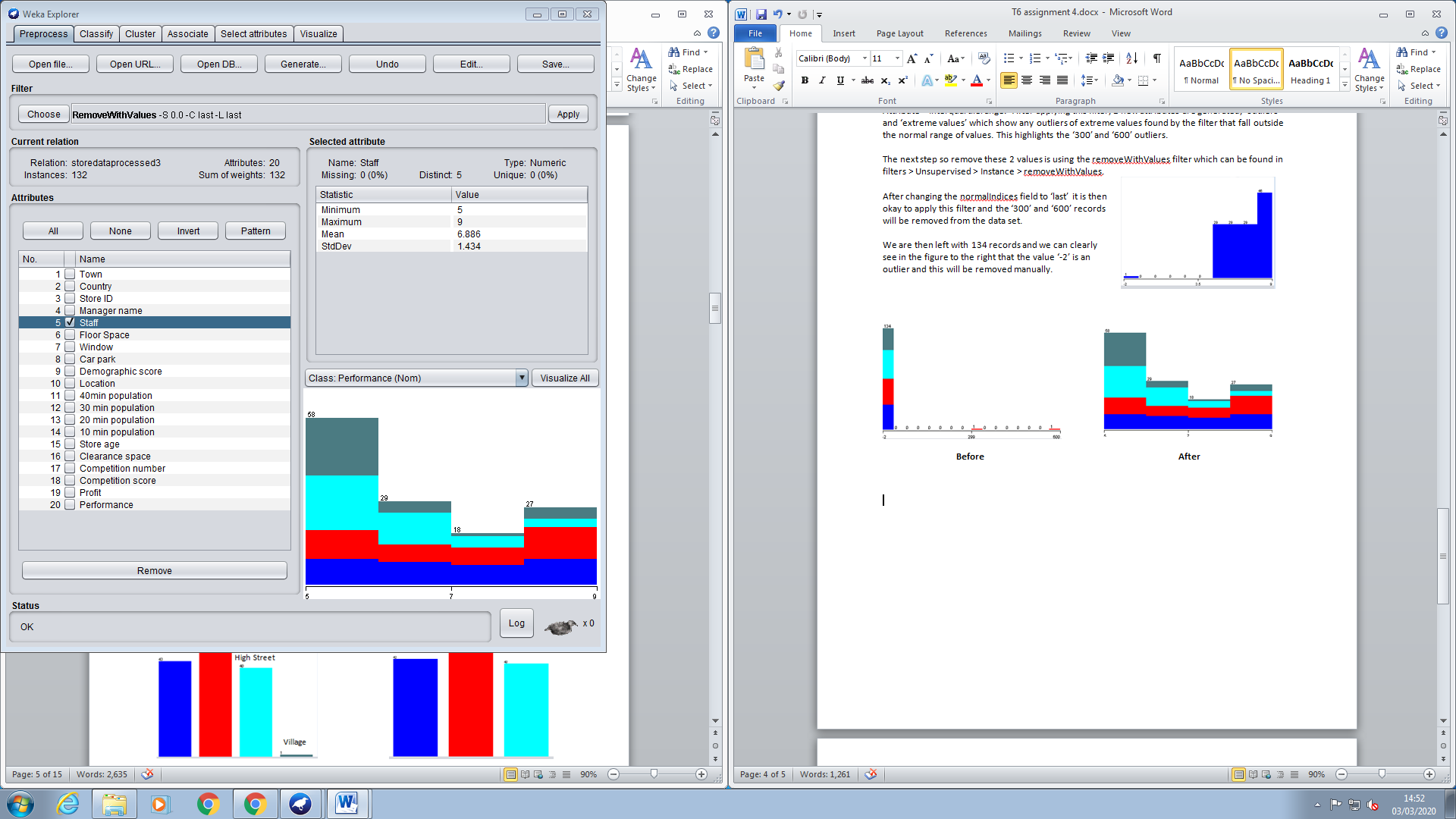
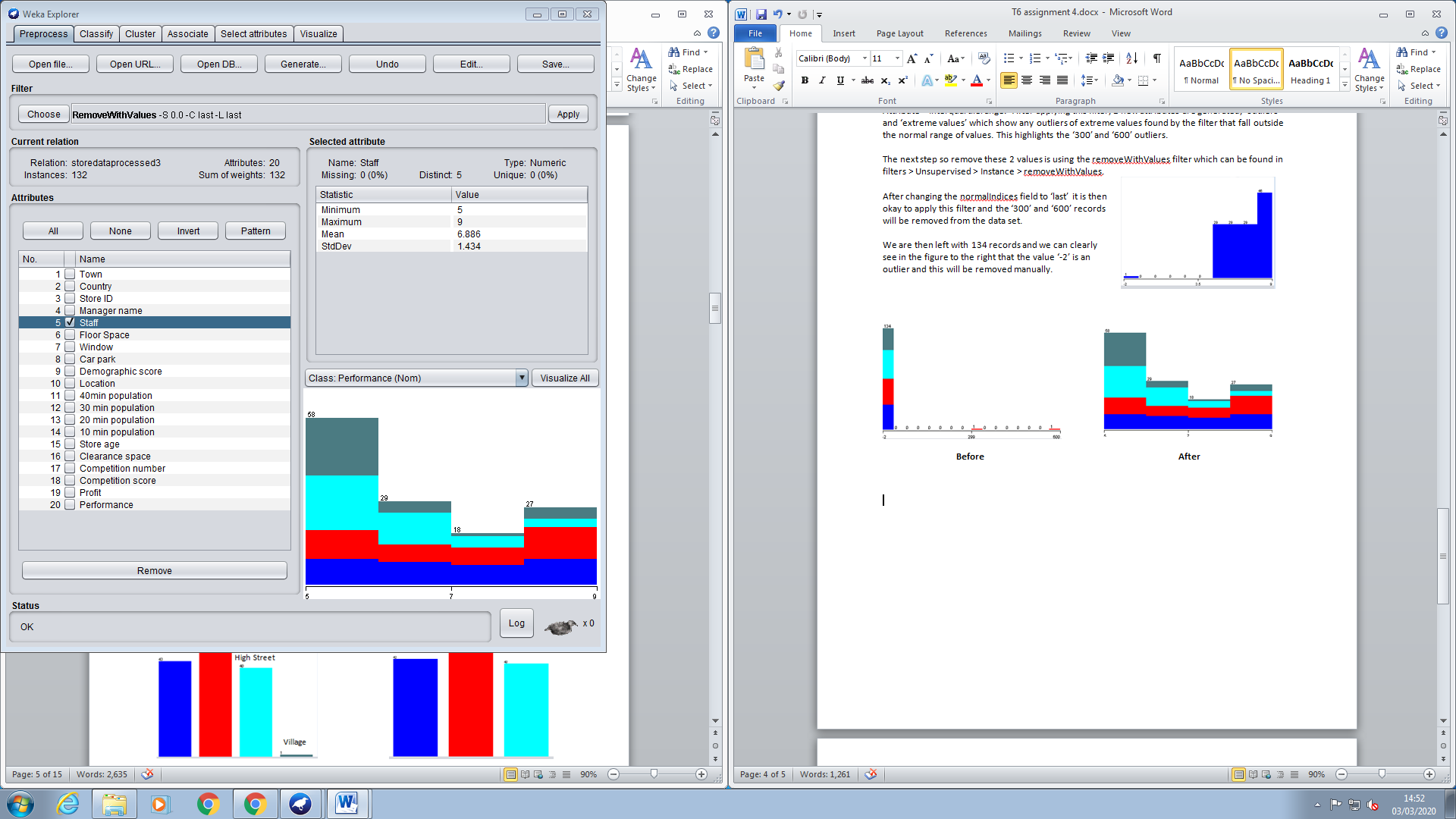


Figure 4 - removing outlier from Staff

## Carpark attribute

In the Carpark attribute there are 7 minority values/potential entry errors that can be shown in the figure below, instances 8, 11, 30, 62, 84, 116 and 122. The expected values for the Carpark attribute are ‘Yes’ or ‘No’. In the data we also have 4 values ‘Y’ and 3 values ‘N’. These will be assumed to mean ‘Y’ for ‘Yes’ and ‘N’ for ‘No’ so will be manually changed to represent this.

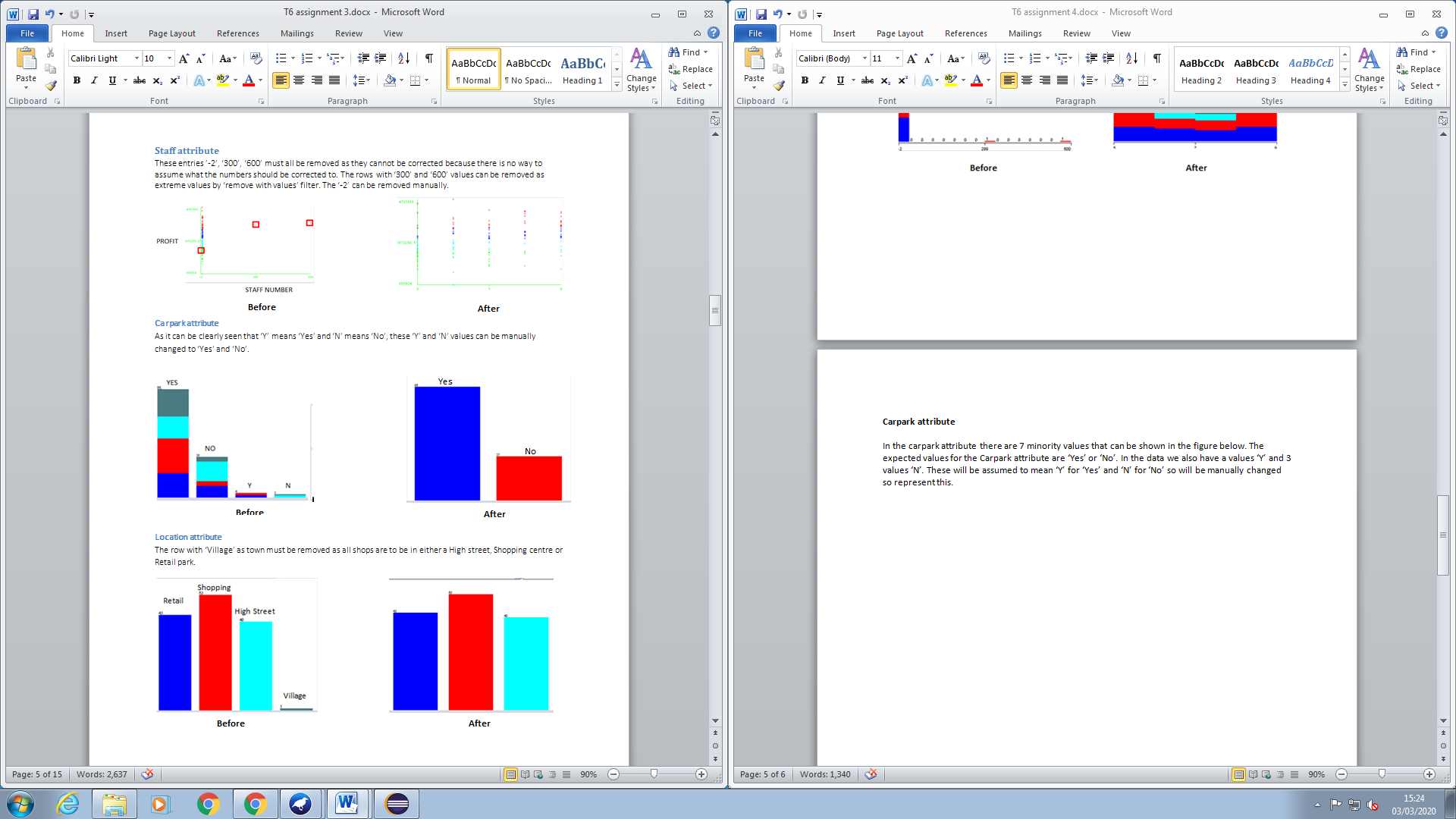
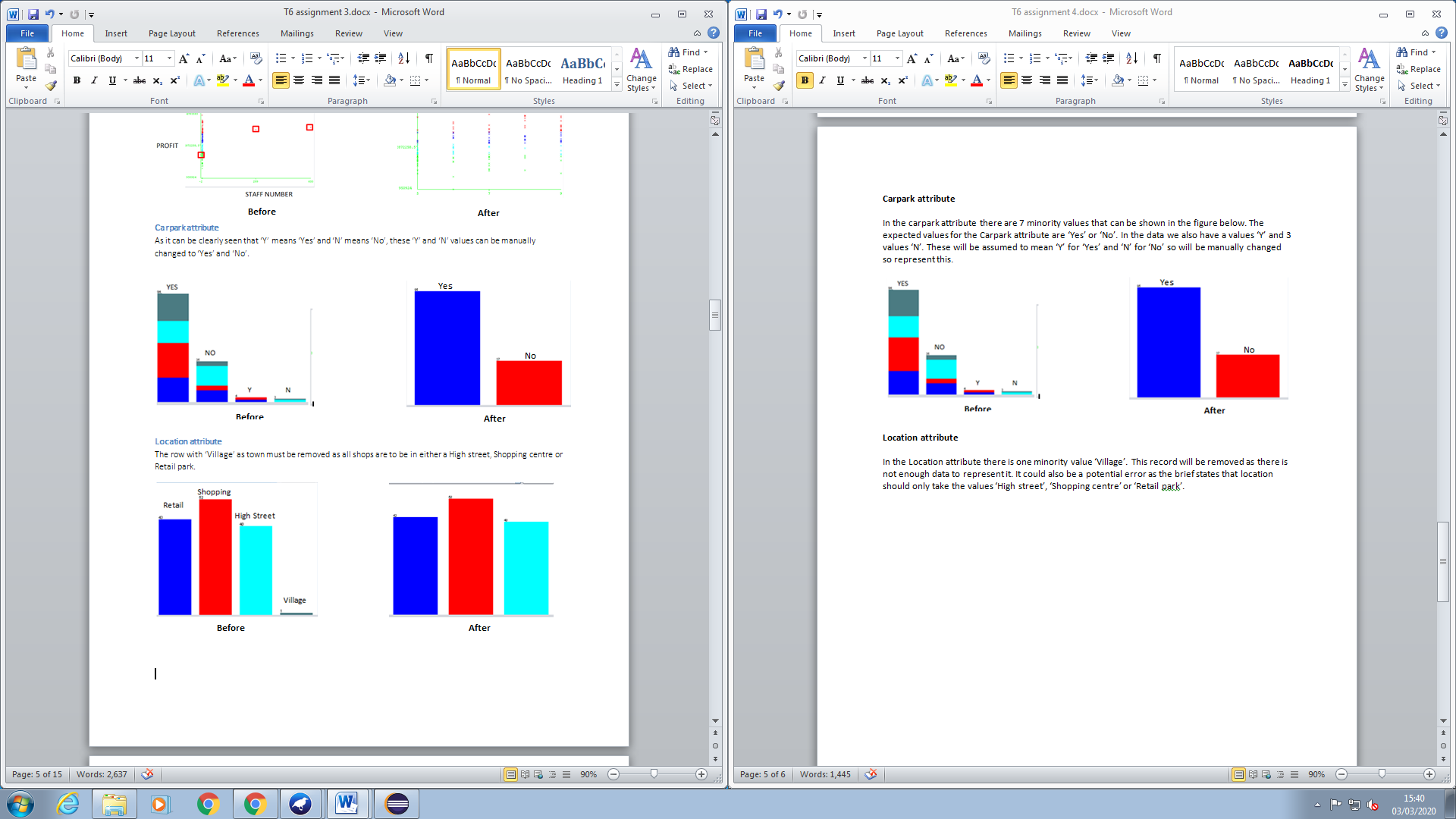


Figure 5 - Carpark cleansing

## Location attribute

In the Location attribute there is one minority value ‘Village’, instance 36. This instance will be removed as there is not enough data to represent it. It could also be a potential error as the brief states that location should only take the values ‘High street’, ‘Shopping centre’ or ‘Retail park’.

Shopping



High Street

Retail

Figure 6 - Location cleansing

## 10 min population

In the 10 min population attribute there are 3 outliers. These have been discovered using the Interquartile range filter in Weka. After doing some quick analysis, I have discovered it is unlikely that any population variables will be included in the model. Therefore, there is no need to remove these records.

## 

## Summary

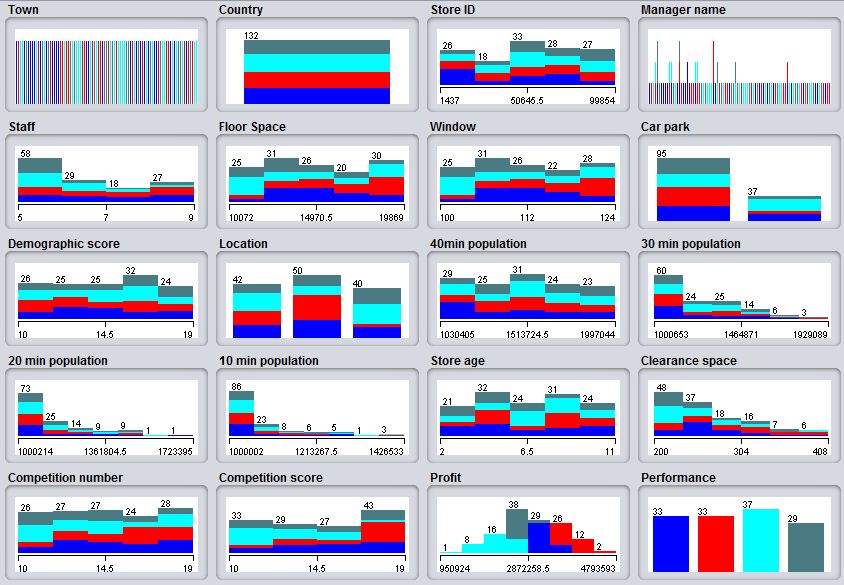


Figure 7 - Distribute of attribute values after pre-processing

# Attribute analysis

The shop owner has provided a number of attributes for which he has gathered data for. There are several attributes that appear to have little to no relation to predicting profit or classifying performance. But we do not immediately know which ones to keep and which to remove. So, we must carry out some analysis to find this out. This analysis has been carried out one the dataset where all potential data entry errors have been removed.

## Town

|  |  |
| --- | --- |
| Represents | The name of the town |
| Datatype | Nominal |
| Continuous/Discrete | Discrete |
| Uniqueness | 100% |

The Town attribute will not be investigated further as it stores no relevant information that could aid the prediction of Profit/classification of Performance. Therefore, it can be classified as noise and removed.

## Country

|  |  |
| --- | --- |
| Represents | Country the store is located |
| Datatype | Nominal |
| Continuous/Discrete | Discrete |
| Uniqueness | 0% |

The country attribute will not be investigated further as all stores are in the UK, it holds no information relevant to predicting the Profit/classification of Performance. As a result it will be classified as noise and removed.

## Store ID

|  |  |
| --- | --- |
| Represents | The stores unique ID |
| Datatype | Numeric |
| Continuous/Discrete | Discrete |
| Uniqueness | 100% |

The Store ID attribute will not be investigated further as the stores unique identifier provides no information that is relevant to predicting Profit/classifying Performance. So due to this it will not be included in any further analysis.

## Managers name

|  |  |
| --- | --- |
| Represents | Store managers name |
| Datatype | Nominal |
| Continuous/Discrete | Discrete |
| Uniqueness | 80% |

The Managers name attribute will not be investigated further and can be classified as noise as the store manager’s name, in this case does not provide any information that would aid a prediction of Profit/classification of Performance. If there was some additional information on the performance of each manager then this could be useful but in this case there is no such information so it will not be included.

## Staff numbers

|  |  |
| --- | --- |
| Represents | No. of staff working in the store |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 0% |

The Staff attribute will be included in further attribute selection analysis. This is because it could have some correlation to Profit as potentially having more staff results in a better customer experience which could result in more profit. But, on the other hand there is some information missing that would help make a decision, like is each store open the same number of hours? As a store that is open 24/7 would probably employ more staff than a shop that is open 9-5.

I will make the assumption that the Staff attribute represents the average number of staff working in the store at any given time and all stores are open the same number of hours at the same time.

## Floor space

|  |  |
| --- | --- |
| Represents | Floor space of the store |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 98% |

The Floor space attribute will be investigated further as it can tell us the size of a particular store and therefore how much stock it can hold. This could potentially have a relation to profit.

## Window space

|  |  |
| --- | --- |
| Represents | Window space of the store |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 0% |

The Window space attribute will be investigated further as it potentially, the more window space a store has, the better it can advertise its stock.

I will be assuming that the window space is the space behind the window where products can be placed to advertise them, rather than the surface area of the windows.

## Carpark

|  |  |
| --- | --- |
| Represents | If the store has a carpark |
| Datatype | Nominal |
| Continuous/Discrete | Discrete |
| Uniqueness | 0% |

The Carpark attribute will be included in further attribute analysis as it could be argued that if a store has a carpark then more customers are able access it and will be willing to travel a further distance to get to the store.

I will assume that the carpark will be an appropriate size for the store as for example a carpark could have 5 spaces or 200.

## Demographic score

Figure 8 - Demographic score scatter

|  |  |
| --- | --- |
| Represents | How a store fits is target audience |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 0% |

The Demographic score attribute will be included in further attribute analysis as it makes sense that the better a store fits its target audience, the more customers will be attracted to the store and the more sales it will make and the more profit will be generated.

It can be shown in the graph that this is not the case. Bizarrely, the opposite of what is expected happens as the better the demographic score, the less profit a store generates.

So as a result of this we will need to analyse this attribute further to discover if it should be included in the model.

## Location

|  |  |
| --- | --- |
| Represents | Location of the store |
| Datatype | Nominal |
| Continuous/Discrete | Discrete |
| Uniqueness | 0% |

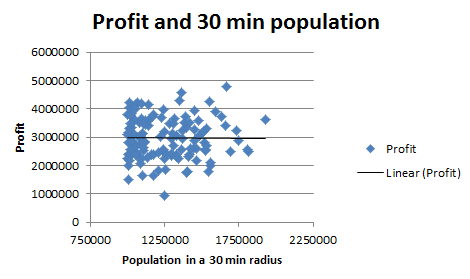
The Location attribute will be investigated further as for example, a shopping centre could be busier than a high street which would result in more customers and more profit in stores located in shopping centres.

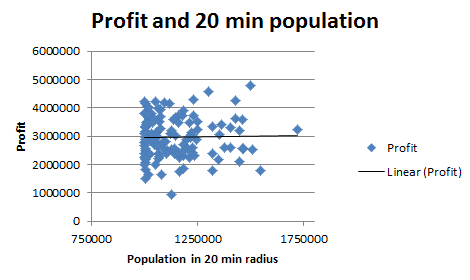
It can be seen in the figure to the right that generally shopping centres perform the best, retail parks performance is spread and high streets perform the worst.

## 

## 40, 30, 20, 10 min population

|  |  |
| --- | --- |
| Represents | Population size in radial travel time (minutes) |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 100% |



All population variables will most likely be removed as it appears there is some ambiguity around what is represented by them. Is this by walking, driving or public transport? If they were all travelling using the same method then this wouldn’t be an issue but there is no way of knowing.

There is also no mention if there are multiple stores within overlapping radiuses. A person could be captured in many stores radiuses but they only shop at one of the stores and this could impact any correlation between the attribute and profit.

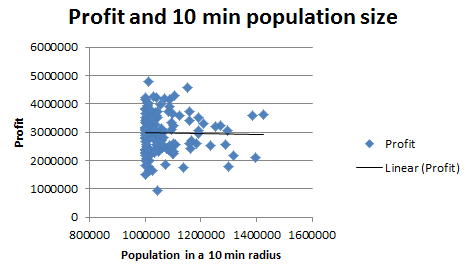


Figure 9 - Profit and population size

By looking at these graphs it can be seen with the best fit line that there is no strong correlation between any population attributes and profit and therefore will provide no information gain in the model. It will be investigated further just to confirm this is the case.

## Store age

|  |  |
| --- | --- |
| Represents | Age of the store |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 0% |

The Store age attribute will be investigated further as perhaps an older shop could have acquired more loyal customers over the years and this could result in more sales and profit.

## Clearance space

|  |  |
| --- | --- |
| Represents | Floor space used for clearance items |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 45% |

The Clearance space attribute will be investigated further as the cheaper the items are the more likely they are to be bought. And the more cheap items there are, the more sales and profit will be generated.

## Competition number

|  |  |
| --- | --- |
| Represents | Number of competing stores in the area |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 0% |

The competition number attribute will be included in further investigation as the number of competing stores could have an impact on the amount of profit a store generated as customers have more options to go to other competition stores.

## Competition score

|  |  |
| --- | --- |
| Represents | Performance of surrounding competition |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 0% |

The Competition score attribute will also be included in further investigation as the better the competing stores are in the surrounding area, the less customers will end up shopping at the World of Bargain store which will result in less sales and less profit.

But, this could also be argued that the better the surrounding stores are, the more customers are attracted to the surrounding area and as a result more customers will end up going to World of Bargains.

## Profit (Target variable)

|  |  |
| --- | --- |
| Represents | The stores profit from the previous year. |
| Datatype | Numeric |
| Continuous/Discrete | Continuous |
| Uniqueness | 100% |

The profit is the target variable so it must be included only in the prediction model. This is what will be used to assess the accuracy of the model by comparing the predicted profit with the actual profit from the last year.

## Performance (Target variable)

|  |  |
| --- | --- |
| Represents | A stores performance |
| Datatype | Nominal |
| Continuous/Discrete | Discrete |
| Uniqueness | 0% |

The Performance attribute is the target variable for classification. It must also be included but only in the classification model. This will be used to assess the classification accuracy of the decision tree model by comparing the model classification with the actual classification of performance.

# Modelling

## Prediction

### Multilayer perceptron

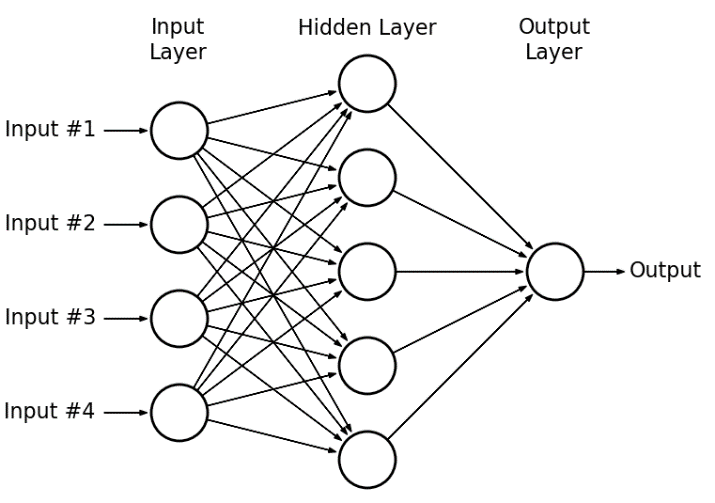
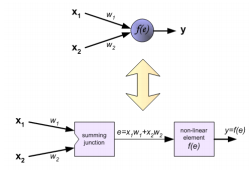
A **multilayer perceptron** is a deep, artificial neural network that was designed roughly based on the brain. It consists of an **input layer** that takes in the signal and an **output layer** that makes a prediction based on the input (this represents the task well as we are required to give a prediction for profit). In between these two layers there are a number of **hidden layers** which are not directly observable from the system inputs and outputs. A hidden layer is where there are **artificial neurons** that take in a set of weighted inputs and produce an output through an activation function. These neurons provide the computational power in the network.

Figure 10 - Neural network structure

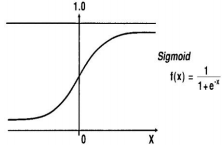
Neural networks are very good at finding meaning in complicated and imprecise data and finding complex patterns that humans and other computing techniques cannot find. Neural networks use the data to modify the weighted connections between all of its functions until it is able to predict the data accurately/with a minimized output error.

**Training (How it learns)**

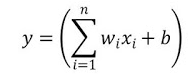
MLP uses a supervised learning technique called **Backpropagation** for training. The goal of this is to propagate the error signal ( ) back to all the neurons. So when the error signal for each neuron is computed, the weights of each input node are modified. It makes a prediction when the error does not change a significant amount and receive the output values.



Each neuron consists of the **weighted (w) sums** (Begins using random weights) **of its inputs(x)** (equation shown in diagram to the right) and an **activation function**. This adds a **bias (b)** to the weighted sum of the input and produces the output(y) of the neuron.



The activation function could be a sigmoid function which equation and graph can be seen to the right.



The activation function is applied in the following way, adding the bias.

Figure 11 - MLP maths

### Attribute Selection Analysis

It is possible to use Weka’s Select attribute feature to aid our analysis to discover what attributes should be included and what ones should be removed.

By selecting the ‘Select attributes’ tab, then the Attribute Evaluator ‘CfsSubsetEval’ and with the search method ‘GreedyStepwise’ this ‘evaluates the worth of a subset of attributes by considering the individual ability of each feature along with the degree of redundancy between them’ and ‘Performs a greedy search through the space of attribute subsets’(Definitions taken from Weka about section).

Applying this to all the available attributes after cleansing gives us the following results:

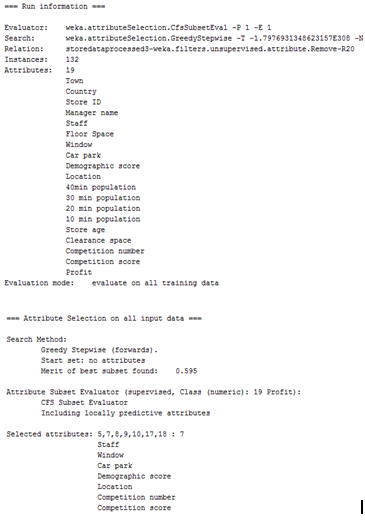


Figure 12- Prediction attribute selection

The results show the subset of attribute that the attribute selection algorithm has deemed more relevant to predicting profit. Staff, Window, Carpark, Demographic score, Location, Competition number and Competition score will be used in the model to predict profit. This also confirmed that the attributes that were suspected to be irrelevant, should be removed (Town, Country, Manager name, Store ID, 40 30 20 and 10 min population).

### The model

The **correlation coefficient** is the value that shows how well the predicted value compares to the actual value. 0 being no correlation and 1 being a high correlation

The **Root mean squared error** is the average error made between the predicted value and the actual value.

The Correlation coefficient (CC) and the Root mean squared error (RMSE) are what will be used to **assess the accuracy** of the prediction model.

The number of **hidden layers** can be increased in order to represent more complex relationships. I will be trialling different values for hidden layers to see if this has an impact on the predication accuracy.

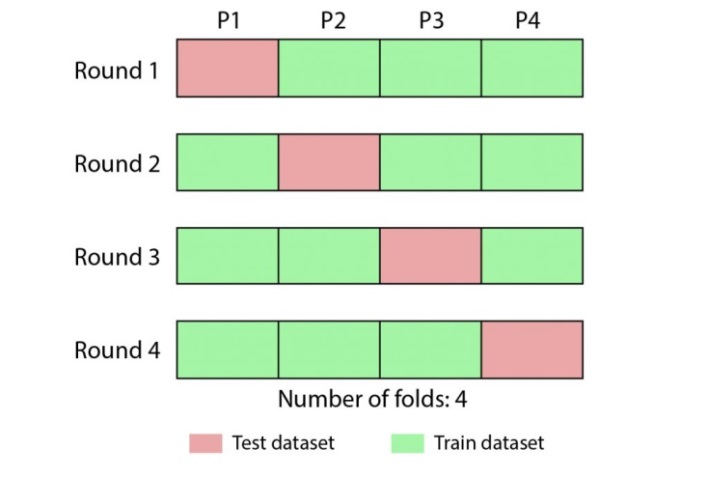
As well as this, the **learning rate** will be altered to discover if this makes any improvement on the performance of the model. The learning rate is ‘The amount the weights are updated’. Learning rate is one of the most important parameters when configuring a neural network. A learning rate that is too small will hinder the models ability to train whereas a learning rate too high will results in unstable training.

Figure 13 - Cross validation

All combinations will at first be using **10 folds of cross validation.** What this means is, the dataset is split into 10 subsets/folds. In each iteration, 9 folds will be used for training and then the final fold will be tested on. This is repeated 10 times and every fold is used for testing and training. As all data is trained and tested on, the results produced are more trustworthy when compared to other methods like a 66% split where 66% of the data is used for training and the remainder it used for testing.

All models will be using the default **epoch** value 500. An epoch is the number of times all training vectors are used once to update the weights. As the dataset is small 500 epochs is adequate.

For comparison, I will be running the default Parameters for Hidden layers and Learning rate. (The default Hidden layer value ‘a’ equals (attribs + classes) / 2).

**Results for 10 folds of cross-validation**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Model ID** | **Parameters** | | **Training result all attributes** | | **Training results subset of attributes** | |
| **Hidden layers** | **Learning rate** | **Correlation coefficient** | **RMSE** | **Correlation coefficient** | **RMSE** |
| **Default** | a | 0.3 | 0.5742 | 600421.4591 | 0.5594 | 703449.1704 |
| **1** | 1 | 0.1 | 0.6736 | 565572.7404 | 0.7539 | 467746.2074 |
| **2** | 1 | 0.2 | 0.6564 | 579496.632 | 0.7422 | 485997.5673 |
| **3** | 1 | 0.3 | 0.4926 | 740966.7469 | 0.7232 | 516852.6549 |
| **4** | 2 | 0.1 | 0.7038 | 526865.7284 | 0.7034 | 521101.3371 |
| **5** | 2 | 0.2 | 0.6829 | 531380.6389 | 0.6975 | 525727.8901 |
| **6** | 2 | 0.3 | 0.634 | 603574.5277 | 0.6962 | 532920.5393 |
| **7** | 3 | 0.1 | 0.6954 | 522697.7701 | 0.6353 | 574497.2526 |
| **8** | 3 | 0.2 | 0.7322 | 484544.8413 | 0.5977 | 614011.5824 |
| **9** | 3 | 0.3 | 0.6891 | 526663.2103 | 0.6101 | 627289.1781 |

### Best MLP model using 10 folds

The best MLP prediction model (Model 1) using the reduced subset of attributes was achieved using 1 Hidden layer and a learning rate of 0.1. The output for this Model was a correlation coefficient of 0.7539 and a RMSE of £467746.2074, which is below the owner’s requested boundary. The RMSE value means that the error between the actual value and the output from the prediction model is equal to £467746.21(2dp as currency) on average. The Correlation coefficient value of 0.7539 is classified as being a ‘strong positive correlation’. So, the attributes selected have a strong correlation with Profit.

When compared to the full set of attributes, the subset of attributes when run using the same parameters in the MLP gives an improvement of nearly £100,000 and an improvement of the correlation coefficient value of roughly 0.08.

Also worth noting is that the best model produced by the reduced subset of attributes is better than the model using all attributes. This is likely due to the removal of irrelevant attributes that can negatively impact the models performance.

### Trialling different fold numbers

To check if there can be any improvements made on this model I will trail a range of different values for the cross-validation folds. Using the same hidden layer and learning rate values and attributes.

|  |  |  |
| --- | --- | --- |
| **Number of folds** | **Correlation coefficient** | **RMSE** |
| **7** | 0.7365 | 480221.9788 |
| **8** | 0.7648 | 456947.5592 |
| **9** | 0.755 | 466726.1247 |
| **10** | 0.7539 | 467746.2074 |
| **11** | 0.7596 | 472397.4946 |
| **12** | 0.7429 | 475422.5311 |

As we can see from the results table, the best RMSE and CC are achieved using 8 folds for cross-validation. The RMSE improve by roughly £11000 and the CC by roughly 0.01.

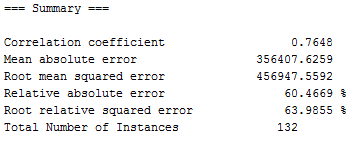
### Best MLP result

Epochs: 500

Hidden layers: 1

Learning rate: 0.1

Folds of cross-validation: 8



## 

## Classification

### Decision tree (J48)

**J48** is a **decision tree** algorithm that can be used for **classification**. A decision tree is a flowchart like tree construction that produces a set of branching decisions that end in a classification. It works best on nominal attributes and when using with numeric attributes, they need to be split into bins.

Typically a decision tree is made up of a **root node**, at least one **decision node**, **branches** and **leaf nodes**. Each node represents a single variable and each branch represents a value that a variable can take. For example when classifying data, the data instance starts at the top of the tree, then follows the branch that corresponds to the values that variable takes. Then this process keeps going until a leaf has been reached and at this point the variable is classified.

There is no set structure a tree can take, but it is not a random structure. It matters where each variable goes as it needs to be optimised to get the maximum number of correct classifications and the classification process should be as fast as possible.



Figure 14 – Decision tree structure

J48 is based on the ID3 algorithm and it uses **entropy** and **information** **gain** in order to select attributes that will give the best split and removes the most uncertainty. The Entropy is the weighted average information across all possible values of a variable, also known as the uncertainty at any point in the tree. When choosing attributes the one that gives the most information and reduces the entropy will be selected.

**Information gain:**



This equation is used to calculate the information gains of a single event.

**Entropy:**



The entropy is calculated using the sum of probability multiplied by the information gain.

### Attribute Selection Analysis

By selecting the ‘Select attributes’ tab, then the Attribute Evaluator ‘CorrelationAttributeEval’ and using the Search Method ‘Ranker’ this evaluates the worth of an attribute by measuring the correlation (Pearson’s) between it and the class (Performance) and the Ranker ranks attributes by their individual evaluations.

In simple terms, by selecting the attributes with high correlation to the selected class (Performance) it allows for: Greater accuracy as the model can be trained on more relevant data, can help avoid overfitting by removing noisy data and reduce the amount of data that should be collected as this will be beneficial to the owner because the data is costly to collect.

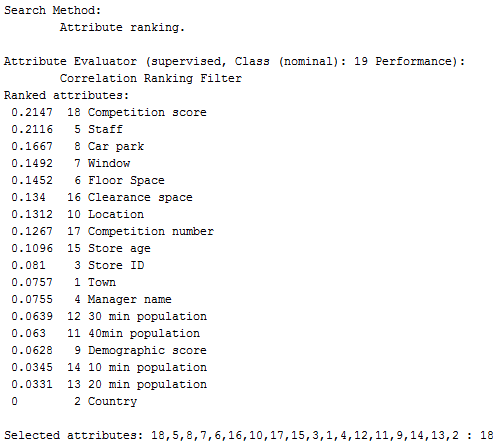
****

Figure 15 - Classification attribute selection

The figure above shows each attributes correlation to Performance in the full set of attributes. The attributes with a higher correlation number have a higher correlation to Performance.

This confirms the original thoughts from the earlier analysis that Town, Country, Manager name, Store ID, 40 30 20 10 min population, should all be removed as they have a low correlation to Performance. Using a cut off of 0.12 this output also shows that the Demographic score attribute should be removed as it has a low correlation to Performance.

### Specific Model Analysis Terminology

**Pruning** is the process of removing sections of the tree that provide little classification power.

The **confidence factor** is the amount of pruning carried out on a decision tree. A low confidence factor will result in a higher amount of pruning and high confidence factor will result in a lower amount of pruning. This is why the values in the results table stop changing after a certain point as at this point there is no more pruning.

The **minNumObj** is the minimum number of instances per leaf.

I will be trialling different parameter values in order to discover the best model configuration. This will include the default settings for comparison.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model ID | Parameters | | Results using all attributes | Results using subset of attributes |
| minNumObj | confidenceFactor | Classification accuracy % | Classification accuracy % |
| Default | 2 | 0.25 | 28.0303 % | 48.4848 % |
| 1 | 1 | 0.1 | 28.0303 % | 45.4545 % |
| 2 | 1 | 0.2 | 28.0303 % | 42.4242 % |
| 3 | 1 | 0.3 | 28.0303 % | 42.4242 % |
| 4 | 2 | 0.1 | 28.0303 % | 47.7273 % |
| 5 | 2 | 0.2 | 28.0303 % | 48.4848 % |
| 6 | 2 | 0.3 | 24.2424 % | 48.4848 % |
| 7 | 3 | 0.1 | 30.303 % | 46.2121 % |
| 8 | 3 | 0.2 | 30.303 % | 46.2121 % |
| 9 | 3 | 0.3 | 27.2727 % | 46.2121 % |

### Best J48 model using 10 fold

The best J48 decision tree model (model Default) achieved using 10 folds, default settings for confidence factor and minNumObj, and the reduced subset of attributes gave a classification accuracy of 48.4848%. This means that out of 132 instances, there was 64 classified correctly and 68 incorrectly.

When compared to the full set of attributes, the subset of attributes (when run using the same parameters) resulted in an improvement of roughly 20% for classification accuracy. This is due to the removal of irrelevant attributes.

### Trialling different fold numbers

|  |  |
| --- | --- |
| **Number of folds** | **Classification accuracy %** |
| **7** | 42.4242 % |
| **8** | 42.4242 % |
| **9** | 50 % |
| **10** | 48.4848 % |
| **11** | 37.1212 % |
| **12** | 43.1818 % |

From the table we can see that using 9 folds for cross-validation gives the best classification accuracy with 50% of instances being correctly classified. When compared with 10 folds, 9 folds gives an improvement of just over 1.5% to the classification accuracy. 50% is a relatively poor classification accuracy.

### Best decision tree result

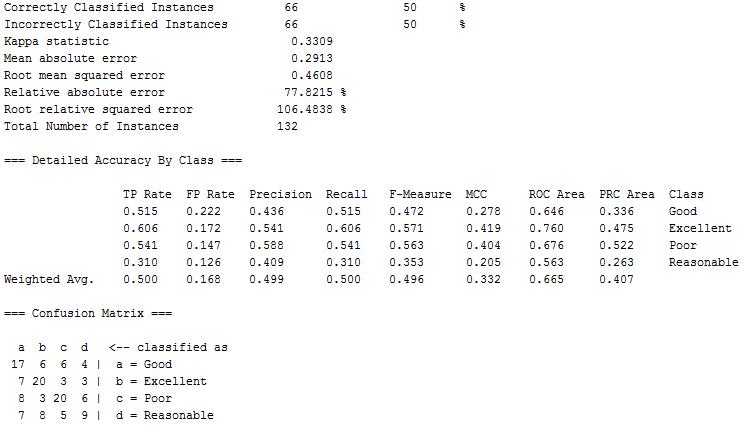


Figure 16 - Best decision tree output

The **confusion matrix** shows the correctly classified instances down the diagonal and the misclassified instances out with this. Take the number 4, top right, this means that 4 instances should have been classified as good, but was classified as reasonable instead.

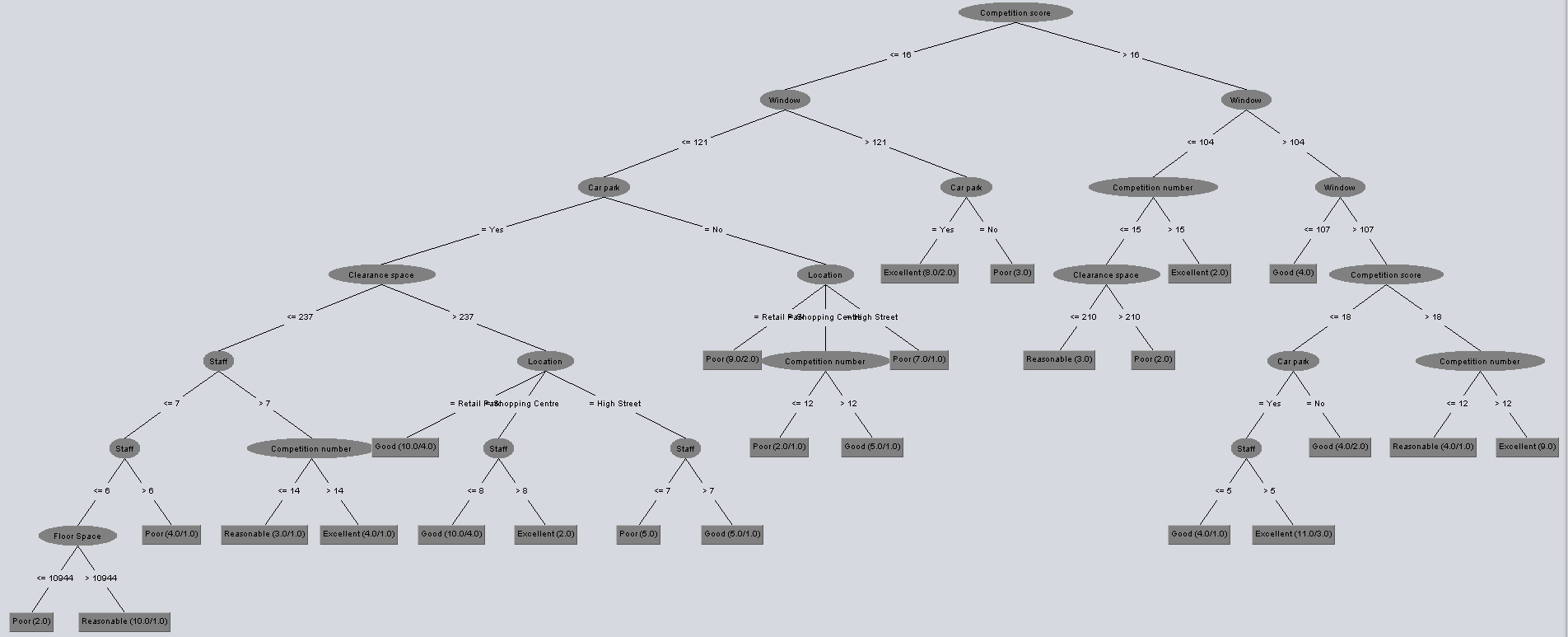


Figure 17 - Best decision tree diagram

### Decision tree ROC curves: Best model vs All attributes, Default settings model

These are the Receiver Operator Characteristic (ROC) curves of the different classifications for Performance. This is a comparison between the best model achieved (see left) using a decision tree vs the model using all the original attributes (see right) just to demonstrate the improvement achieved. The x axis is the false positive rate and the y it the true positive rate. We ideally want to maximize the area below the line and it also have a steep incline at the beginning.

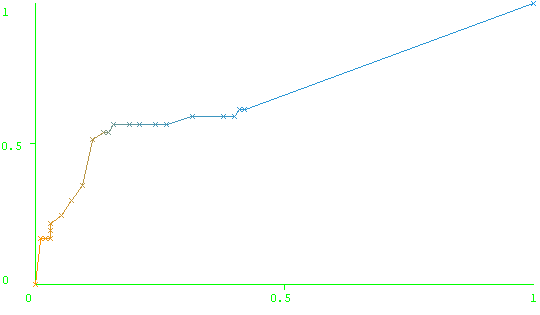
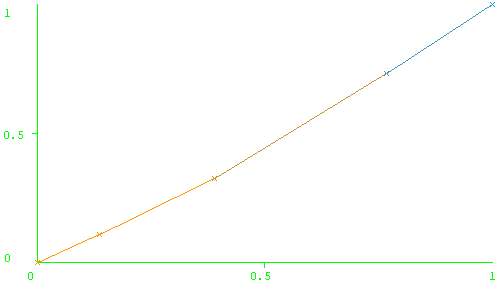
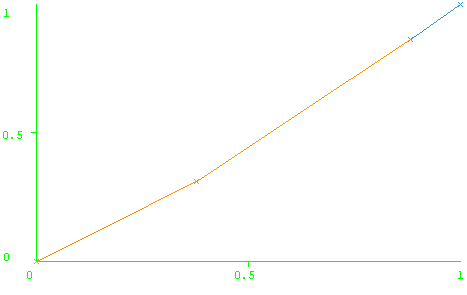


Figure 18 - Poor ROC curve best model vs worst



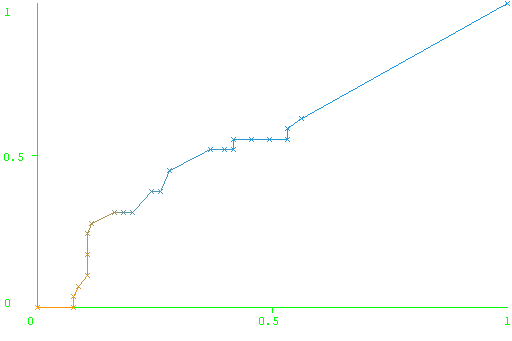
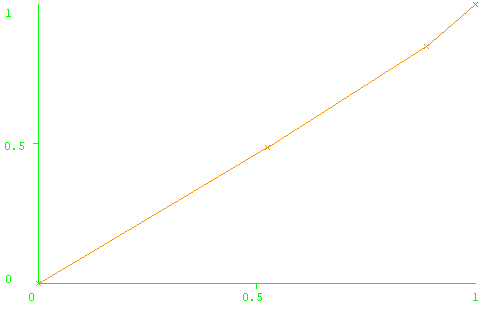
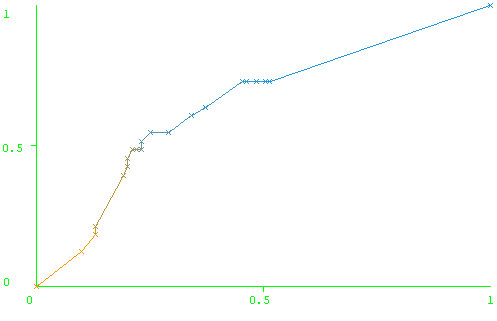


Figure 19 - Reasonable ROC curve best model vs worst

Figure 20 - Good ROC curve best model vs worst

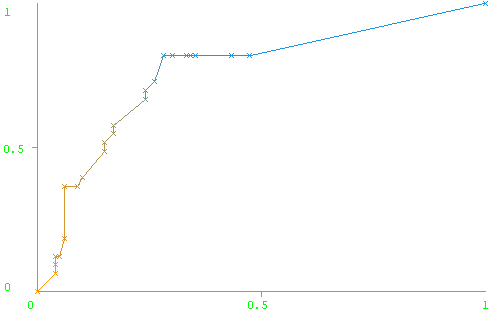
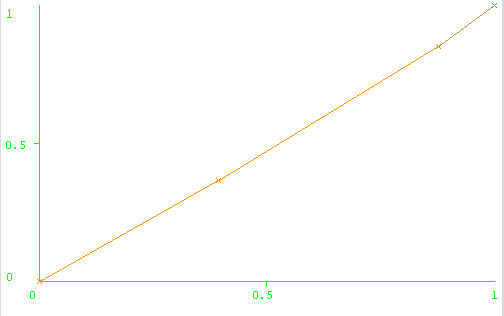


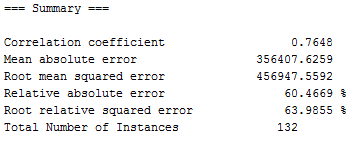
Figure 21 - Excellent ROC curve best model vs worst

# 

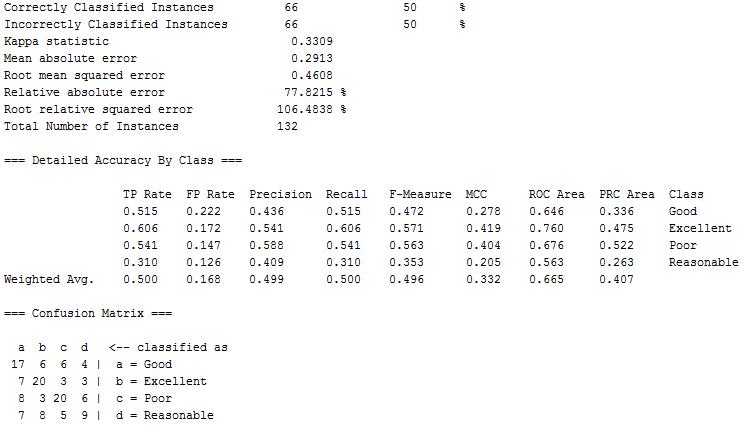
# Recommendations

## Conclusion of results

### Best MLP result (Prediction)



### Best decision tree result (Classification)



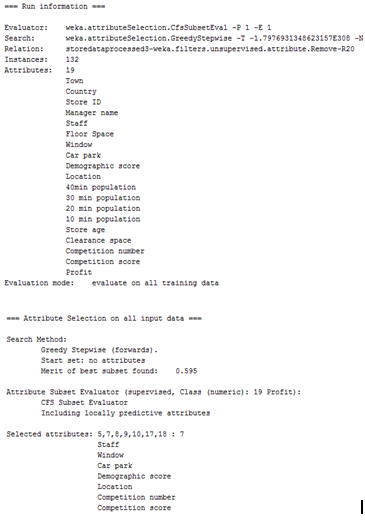
## Recommended technique

I would recommend the owner use the MLP model as it is able to provide an actual value for the prediction of Profit of a store with a reasonable level of accuracy() below the requested £500,000.

I would recommend against using the decision tree as it is only able to correctly classify instances 50% of the time and this level of accuracy is likely too unstable for use the business. It is also not very useful as if does not provide a specific value, just a classification of its performance.

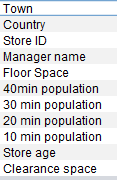
## Recommended attributes

I would recommend the owner continues to gather the following attributes for prediction:



This combination was recommended by the Weka attribute selection feature. These attributes were deemed to give the most accurate prediction.

Rest of variables that will not be used are:



These attributes are not useful as they lack any strong relation with profit.

My recommendation for the population data is to stop gathering it as any slight benefit in prediction it may provide, is negated by the cost of collection.

# References

Data prep lecture slides

Prediction lecture slides

Classification lecture slides

<https://pathmind.com/wiki/multilayer-perceptron>

<http://www.cs.stir.ac.uk/courses/ITNP4B/lectures/kms/4-MLP.pdf>

<https://ieeexplore.ieee.org/document/7808002>

<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>

<http://www.inf.ed.ac.uk/teaching/courses/inf2b/learnSlides/inf2b11-learnlec13-nup.pdf>

<https://docs.oracle.com/cd/B28359_01/datamine.111/b28129/process.htm#CHDFGCIJ>

<https://www.researchgate.net/figure/A-hypothetical-example-of-Multilayer-Perceptron-Network_fig4_303875065>

<https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm>

<https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/>

<https://blog.quantinsti.com/cross-validation-machine-learning-trading-models/>

<https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d>

<https://www.researchgate.net/post/how_to_assess_accuracy_of_prediction_model>

<https://www.researchgate.net/post/What_could_be_the_strategies_to_avoid_overfitting_in_deep_neural_Networks>

<https://www.techopedia.com/definition/33264/hidden-layer-neural-networks>