CCT College Dublin

Assessment Cover Page

To be provided separately as a word doc for students to include with every submission

---

Module Title:	Programming for DA Statistics for Data Analysis Machine Learning for Data Analysis Data Preparation and Visualisation
Assessment Title:	Pedestrian Footfall Predictions For The Dublin City Area
Lecturer Name:	Sam Weiss Marina Iantorno Muhammad Iqbal David McQuaid
Student Full Name:	Faelan Redmond
Student Number:	Sba22190
Assessment Due Date:	11/11/2022
Date of Submission:	11/11/2022

Declaration

By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

Pedestrian Footfall Predictions For The Dublin City Area

Faelan Redmond
SBA22190
sba22190@student.cct.ie

November 11, 2022

Abstract

The overall goal of this project is to accurately predict the level of street usage of Dublin streets with the use of classification machine learning algorithms. Along the way we will take a look at traffic counts comparing the mean number and distribution of cyclists vs pedestrians entering the city through each of 33 count sites each morning, before applying the Poisson distribution to calculate the probability of seeing more than 120 at a given site each morning. We then move to the pedestrian footfall data where EDA is carried out in order to see how footfall patterns change with weather and if specific streets follow the same usage patterns. Finally we will apply a decision tree and random forest classifier to the dataset, and evaluate the performance of each model and identify the best solution to our goal.

Contents

1 Introduction

in the midst of the 4th industrial revolution, gathering and understanding data is becoming more important than ever before. As populations in major cities soar over recent years, collecting high quality data and carrying out the accompanying analysis is our most vital tool in understanding what resources are needed, and where they are needed the most. This data-driven decision making greatly reduces excess expenditure, and expenditure in areas that don't necessarily need it.

Aside from city planning applications, Data also has a positive business and social aspect. If prospective business owners have insight into where people are most likely to be during the days or nights, it makes it much easier to make informed decisions about where the optimal location would be for the sales of goods and services, thus helping increase the likelihood of success. Conversely this footfall and usage density data can also help people avoid the crowds. For neurodivergent or anxious individuals large crowds can be quite overwhelming and may restrict them from getting daily tasks completed, so if they have insight into where people are likely to be and when this would allow them to better plan visits to certain locations, taking one obstacle out of their day-to-day life. Our overall objective in this project is to produce a machine learning algorithm which will take a few simple inputs and provide a prediction regarding the usage level of streets in Dublin, along the way we will utilise multiple statistics, data preparation, EDA, and machine learning tools & techniques.

2 Data

2.1 Data

This project will follow two main stages. In the first we will look to carry out a statistical analysis on the numbers of people entering the city between the hours if 7am and 10am on foot, and on bicycle just as an interesting side quest. Following this, we will then look at the usage of streets in the Dublin city area & try to predict the usage levels of the streets at any hour of the day we choose. The following subsections will briefly discuss the main datasets used for each section mentioned above.

2.1.1 Traffic counts Data

To facilitate the first section of this project, which, as we mentioned, revolves around the analysis of people entering the city on foot and bicycle during morning traffic, we will use the "Traffic Counts: Cordon Count, Quays Count DCC" dataset found through the data.gov.ie data repository. This data comes in the form of 33 separate Excel workbooks that contain the counts of many vehicle types between the hours of 7am -7pm over a two-day period in November 2018, where each workbook represents the data recorded on a single counter. Since the data came pre-tabulated & formatted in excel some work was required to convert this to a pandas dataframe format, the data contained within however was simply discrete numeric data representing counts and totals, however, we were only concerned with the raw counts, int64 format, of the pedestrian('Ped') and cyclist('P/C') columns. The locations of all the counters can be seen below on this map visual provided by DCC.

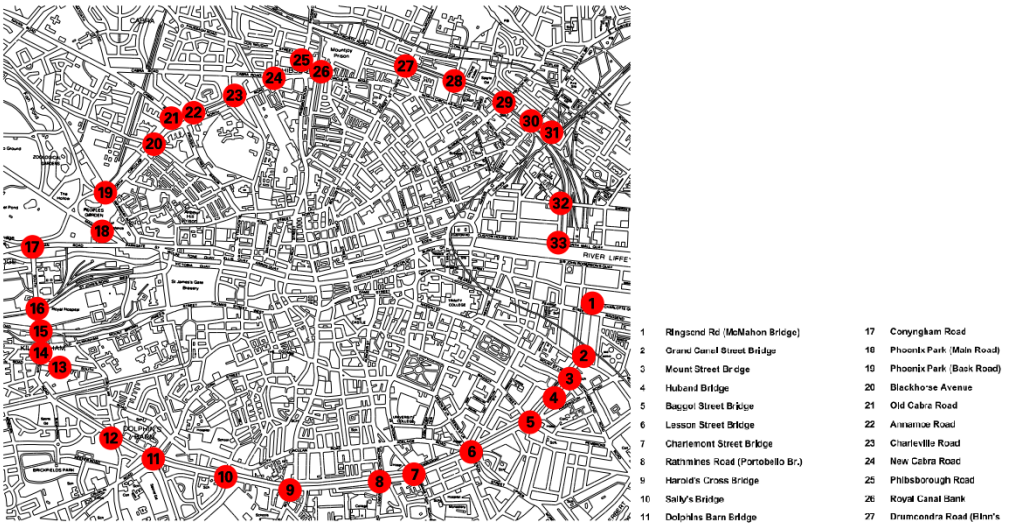


Figure 1: Locations of counters used to record traffic counts

2.1.2 Pedestrian Footfall Data

The second part of the project which revolves around using predictive analysis in order to calculate street usage levels. To facilitate this three datasets are used, the most important one will be the "Pedestrian Footfall DCC" dataset; however, it will be complimented by the "Public Lighting DCC" data as well as hourly weather data recorded at Dublin Airport. These datasets were obtained from the data.gov.ie repository and MET eireann, respectively.

In these datasets we are only concerned with specific attributes, while the rest can be ignored or discarded. From the pedestrian footfall data we will be using the whole dataset, these are all discrete values imported an int64 datatype in python, any blank values will be imported as nan and dealt with later. From the weather data we will focus only in the rain and temp attributes which are imported as float64 datatypes in pandas, and from the public lighting data we only require the latitude and longitude coordinates which will be imported as float64 also the rest of these datasets can be ignored or discarded.

3 Analytical Methods

A data scientist can only be considered as good as his/her toolkit is. Given any dataset, there will be some degree of preparation or analysts required in order to optimise the utility it can provide to the problem at hand. This section will give a very high level overview of the methods we used while working our way through this project including the description of algorithms and distributions used, as well as some simple data preparation steps. For any further information regarding steps taken or specifics of how these techniques or algorithms were implemented please refer to the python notebooks included with this document.

3.1 Traffic Counts Data

3.1.1 Statistics

Calculating probabilities is often the area in statistics that people struggle with most; this is because each probability distribution we see is only valid if a few specific conditions are

met. In this case we are trying to calculate the probability that a number of events will occur during a specific period of time. From this description, it should be obvious that the distribution we need to use is the Poisson distribution. The formula for this is seen below.

$$p(x;\lambda) = \frac{e^{-\lambda}\lambda^x}{x!} \text{ for } x = 0, 1, 2, \cdots$$

Here λ is the mean number of events per unit time t. The use of this distribution actually works out rather conveniently as it gives us a fantastic excuse to implement the normal distribution. When the value of λ increases to more than 20(UCLA 2022), the normal distribution becomes a reasonable approximation to the Poisson distribution allowing $\mu = \lambda$ and $\sigma = \sqrt{\lambda}$. This process involves us going from using a discrete distribution to a continuous distribution, due to this a continuity correction factor is required. This is simply the inclusion of $X \pm 0.5$ for more than or less than calculations respectively(Stephanie 2021). From here the Z scores can be calculated as usual and the probabilities read off tables,

$$Z = \frac{X - \mu}{\sqrt{\sigma}}$$

Then its just a matter of consulting the pre-calculated z-score tables to find the corresponding probability value.

3.2 Pedestrian Footfall Data

As the requirements of for statistics have been satisfied in the above section, from here on we will focus on the major steps taken to prepare the pedestrian footfall and complimentary datasets for machine learning. We will then go on to briefly discuss the algorithms used for the machine learning applications section, and some motivation as to why they were chosen.

3.2.1 Data Preparation

Prior to carrying out any EDA or machine learning we must first build a dataset from which we can do this. The following section will describe the high level description of how we did this. During the preparation of the pedestrian footfall data, one of the biggest challenges was that of dealing with missing data. We opted to focus on q3 2022, however within this period there were gaps of 20-300 hours where counts were completely missing. This could be due to an error in data recording or a fault in the detector during this period, if this is the case the data is said to be missing completely at random(S.Buuren 2022). There are many approaches to data imputation such as substituting in statistical values like the mean/mode/median or simply filling in the previous or next value. Due to the large gaps in this time series data none of the above were considered suitable as they would fail to capture the natural variation seen in timeseries data. Instead KNN imputation was used to impute the missing data. This will take into account the data in other columns and choose the missing values based off the mean from its nearest neighbours(Chowdhury 2020).

An important part of understanding our pedestrian footfall data is identifying any usage patterns that are common among different streets. In order to identify these, we used a k-means clustering algorithm which will assign streets to a cluster based on their apparent usage patterns. Choosing the optimal K is important, we determined this value using the "elbow method" which will show us how the squared error changes with different values of K. The

value after which this change becomes more gradual is our optimal parameter(Sampaio 2022). In order to prepare the data for this we first defined periods of use that fall under [morning rush, daytime, evening rush, night] during the weekday & weekend by using the datetime package in python and conditional statements. The dataframe was then manipulated using df.groupby() so that the street names are arranged as rows, with the time periods being the columns; see below for illustration.

Clustering Category	Weekday Daytime	Weekday Evening Rush-hour	Weekday Morning Rush-hour	Weekday Overnight	Weekend Daytime	Weekend Evening Rush-hour	Weekend Morning Rush-hour	Weekend Overnight
Aston Quay/Fitzgeralds	990169.0	753907.0	838441.0	930192.0	409176.0	329439.0	207839.0	467816.0
Aston Quay/Fitzgeralds IN	434261.0	304535.0	402595.0	462352.0	184519.0	141213.0	108161.0	230300.0
Aston Quay/Fitzgeralds OUT	555908.0	449372.0	435846.0	467840.0	224657.0	188226.0	99678.0	237516.0
Bachelors walk/Bachelors way	632736.8	559734.4	571256.8	670543.6	258457.6	210050.8	130086.0	315237.4
Bachelors walk/Bachelors way IN	334174.6	292390.8	316574.8	302709.8	144539.8	112007.2	69006.0	150018.4

Figure 2: format of final clustering dataframe

From here we start implementing the complimentary information, in a separate dataframe the clustering results and street names were included as columns as a starting point. Since the DCC counter locations data is out of date we passed the street names through the google api with the string "Dublin, Ireland" appended to the end of each to mitigate the chances of receiving incorrect coordinattes. A haversine function was then used with these coordinates and those of public lighting locations to determine the number of street lights within a 0.1km radius, the final dataframe looks like this,

	Cluster ID	Street	Street_geo	lat	lng	Lighting
0	2	Aston Quay/Fitzgeralds	Aston Quay/Fitzgeralds Dublin Ireland	53.346674	-6.259631	49
1	2	Bachelors walk/Bachelors way	Bachelors walk/Bachelors way Dublin Ireland	53.347354	-6.260028	60
2	0	Baggot st lower/Wilton tce inbound	Baggot st lower/Wilton tce inbound Dublin Ireland	53.334385	-6.245730	23
3	1	Baggot st upper/Mespil rd/Bank	Baggot st upper/Mespil rd/Bank Dublin Ireland	53.334034	-6.245192	26
4	2	Capel st/Mary street	Capel st/Mary street Dublin Ireland	53.348467	-6.268745	38

Figure 3: cluster$_r esults data frame with cluster, street, coordinate, and lighting data$

To finally bring everything together we melted the footfall dataframe so that all the street names, time, and counts appear across three columns. To this we merged the weather data for each day over the recorded period and finally the dataframe mentioned above such that all our data was in one place. After this some feature engineering was implemented to ensure that our data could be applied to machine learning. This included implementing a binary classification for if its wet and another for if its considered warm. Anything greater than 0mm rain results in a wet classification, 1, and anything 12 degrees or greater is considered a warm day,1. We also included the day, month, and hour value relevant to each entry. Last but not least we created the target class for our machine learning algorithm, street usage was calculated by dividing each count value by the max counts recorded on that street over the 3 month period. From these values we then created the class attribute 'usage level' which have the values below.

- Low = 0
- Medium-low = 1
- Medium-high = 2

- Medium-high = 3

	Time	Clustering Category	Street	Counts	date	rain	temp	Cluster ID	lat	lng	Lighting	max counts	usage	warm	wet	usage_level	Hour	Day	Month	Street ID
0	2022-07-01 00:00:00	Weekday Overnight	Aston Quay/Fitzgeralds	1614.0	2022-07-01 00:00:00	0.0	12.9	2	53.346674	-6.259631	49	4921.0	0.327982	1	0	1	0	4	7	1
1	2022-07-01 01:00:00	Weekday Overnight	Aston Quay/Fitzgeralds	1267.0	2022-07-01 01:00:00	0.0	12.6	2	53.346674	-6.259631	49	4921.0	0.257468	1	0	1	1	4	7	1
2	2022-07-01 02:00:00	Weekday Overnight	Aston Quay/Fitzgeralds	1169.0	2022-07-01 02:00:00	0.0	12.4	2	53.346674	-6.259631	49	4921.0	0.237553	1	0	0	2	4	7	1
3	2022-07-01 03:00:00	Weekday Overnight	Aston Quay/Fitzgeralds	575.0	2022-07-01 03:00:00	0.0	12.6	2	53.346674	-6.259631	49	4921.0	0.116846	1	0	0	3	4	7	1
4	2022-07-01 04:00:00	Weekday Overnight	Aston Quay/Fitzgeralds	438.0	2022-07-01 04:00:00	0.0	12.2	2	53.346674	-6.259631	49	4921.0	0.089006	1	0	0	4	4	7	1

Figure 4: Final pedestrian footfall dataframe containing all data required for analysis

3.2.2 Machine Learning

Finally to cap off the project we applied supervised machine learning algorithms to the data to try and predict the usage level of a street at a given time of the day. Supervised learning was used here as we already have the class labels of each row of data & intend to use these to train a model. Tree based models were used here as they were most likely to preform well regardless of the weak attribute/class relationships. The first algorithm used was a decision tree; these consist of a series of nodes stemming from the root splitting the data into increasingly purer nodes as it progresses in depth. The optimal attribute used to split in each node will be selected by computing values such as the entropy and gini index, which give measures of node purity. A common issue with decision tree models is overfitting where the model trains too rigidly to the training dataset; to mitigate this, we include a ccp-alpha(cost complexity pruning) parameter in our algorithm, which selectively prunes the model in order to increase its generality. This means that the model will better adapt to unseen test data.

The second algorithm used is the random forest classifier. This is quite closely related to the decision tree algorithm, being tree based, however the difference here is that its an ensemble classifier. This means that it will create many decision trees based off of random samples of the training dataset known as a forest. The best preforming of these models will be selected and utilised in the final model, It is this approach that makes random forest so superior to a decision tree classifier as it reduces overfitting and bias by averaging the results of many trees. Correlation between trees in the forest will be constrained by making use of bagging, which is the act of randomly sampling from the training data with replacement(Brownlee 2016), and random feature selection, which means that each tree is constructed of a different subset of features. This is effective an evolution of the decision tree, so running them both and comparing the results makes for an interesting exercise.

In order to ensure that the models are as accurate as possible, an algorithm was used to split the data into folds and test various combinations of model parameters, returning the best performing combination. Instead of using gridsearchcv here, we opted to use an experimental tool from sklearn called HalvingGridSearchCV. This approach was chosen because of its superior speed. While gridsearch simply tests every model on the whole dataset, this algorithm instead starts by testing each potential model on only a small selection of data. Each time it does this, only the best performing models are carried forward to the next iteration. Each time it does this more and more data is included in the testing until the best model is returned.

The Framework used throughout this project closely follows the steps outlined in CRISP-DM. This was chosen as the steps best align with the natural flow of understanding, processing, and analysing data. This drives efficiency reducing wasted time on blindly analysing data while trying to answer either no question or the wrong question. This framework is the

standard for nearly every industry, for this reason becoming familiar with it now would be extremely beneficial and be a big help in applying what we are learning during these projects to real world business or research problems, and understanding data analysis processes in industry.

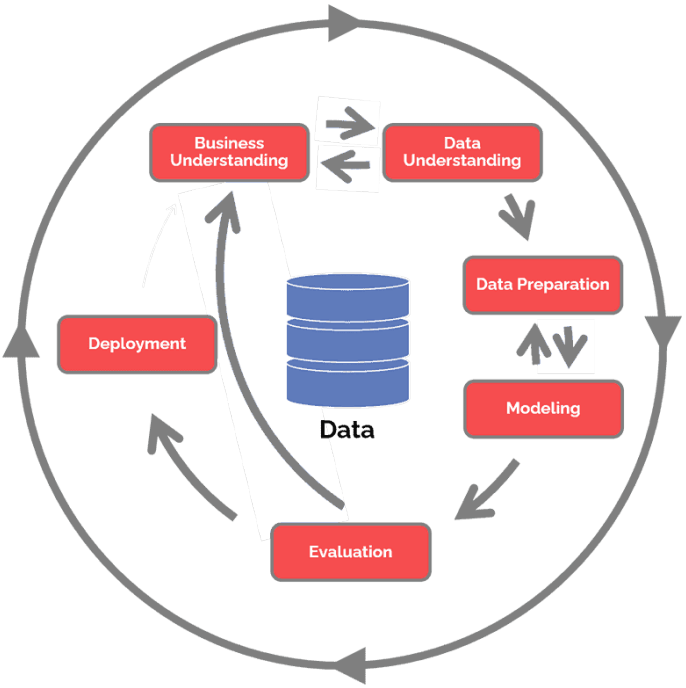


Figure 5: CRISP-DM Steps

4 Results & Discussion

4.1 Traffic Counts

4.1.1 Statistics

Prior to carrying out any predictive analytics regarding pedestrian footfall counts in Dublin city, we first thought it would be interesting to investigate the number of people entering the city by foot between the hours of 7am-10am. Initially we looked at the total inbound counts at each count site, figure 6, from this we found that 25.44% of the inbound counts were pedestrians & a further 13.04% of the counts were cyclists.

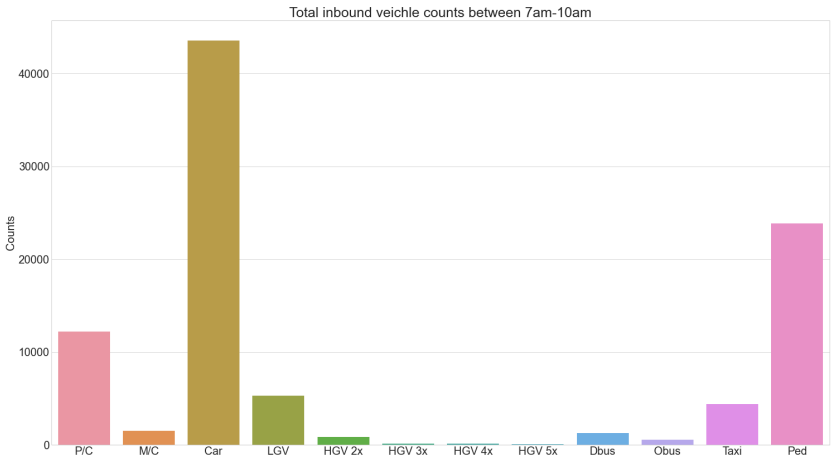


Figure 6: Total inbound counts of vehicle between 7am and 10am

We can further drill down into these figures, looking only at the percentage of travelers who are walking or cycling at the count site level.

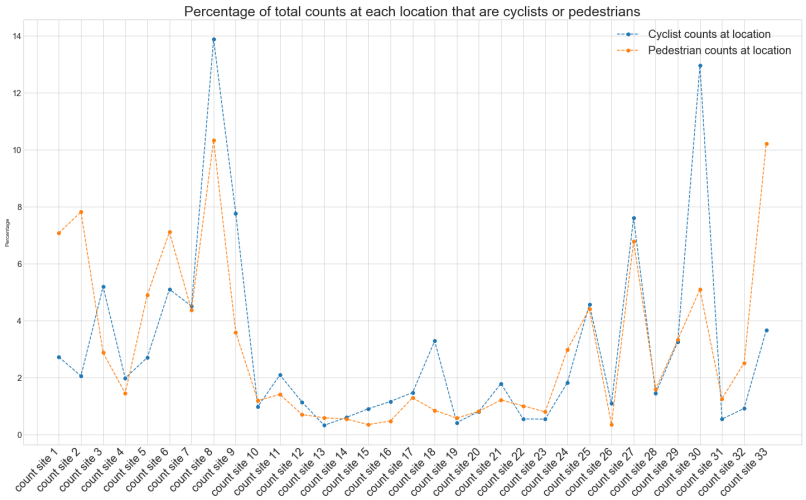


Figure 7: Percentages of pedestrians and cyclists between 7am and 10am at each counter site

The highest percentage of cyclists occurs at station 8 & 30, and the highest percentage of pedestrians occurs at sites 8 & 33. This could be an indication of the presence of safe cycling and walking infrastructure in these locations in the form of well-maintained footpaths or cycle lanes. Upon consulting google street view, using locations marked on figure 1, we can see that this is the case and these locations boast healthy pedestrian/cyclist infrastructure. Looking at sites 26 & 13 where some of the lowest cycle & pedestrian counts were recorded shows narrow footpaths and no cycle close by lanes, this could be a reason for for the low counts coupled with the possibility of a relative lack of employers in these areas.

To further gain insight into the number of people entering the city on foot or bicycle we will generate some descriptive statistics related to the total pedestrian or cyclist counts at each site. Due to the large variation in count numbers here, use of the mean and standard deviation would not be a good approach to take, as the resulting values would be heavily influenced by the extreme values in the data. Instead we will employ a 5 figure summary comprised of the maximum, minimum, lower quantile, upper quantile, and the median. The values calculated for the cycling data are visualised in the below boxplot,

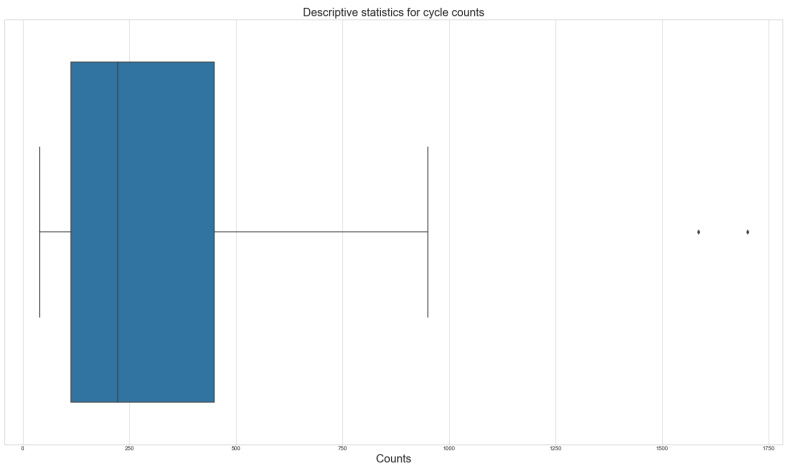


Figure 8: Boxplot of total cycle counts at each counter site

The notable values from this box plot are:

Statistic	Value
Min	40
25%	112.5
Median	223
75%	448.5
IQR	336
Max	1699.5
outlier	1699.5, 1585

Table 1: Descriptive statistics for total cycle counts at each station

And for the pedestrian data,

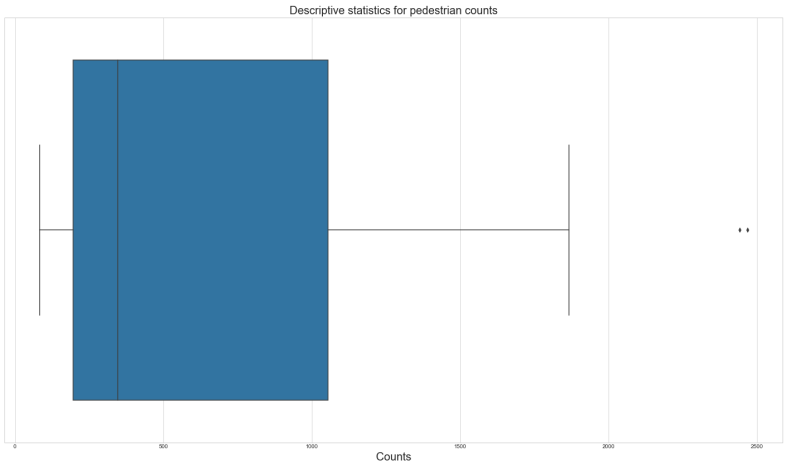


Figure 9: Boxplot of total pedestrian counts at each counter site

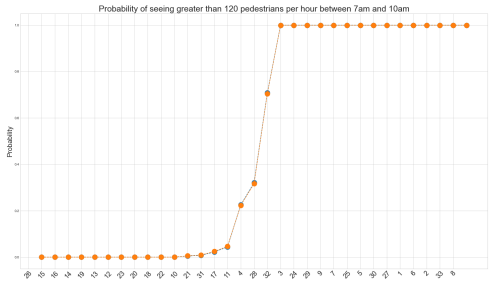
Statistic	Value
Min	84
25%	195.5
Median	346.5
75%	1054
Max	2466.5
IQR	858.5
outlier	2466.5, 2441

Table 2: Descriptive statistics for total pedestrian counts at each station

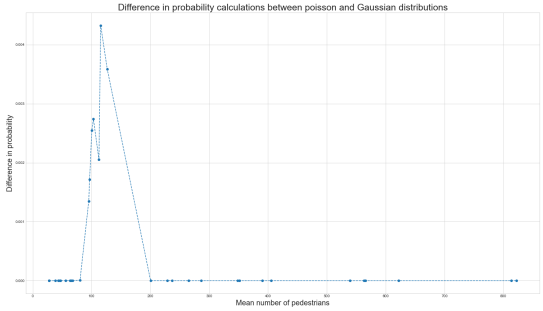
From these descriptive statistics we can see that the median number of pedestrians crossing the counters during the morning hours is much higher than that of cyclists. This may be because cycling can be quite a dangerous form of transport on public roads since there is still a lot of development to go regarding cycling lanes and infrastructure, and bicycle theft is prevalent in the city. We can see from the difference between the median and the 75% quantile that there is a high degree of dispersion in these high pedestrian counts; this is further reflected in the inter quantile range(IQR). The larger IQR seen in the pedestrian counts tells us that we are likely to see a much higher deviation from the median value at each count site than we would with the cycle counts data. In summary, while we are likely to see much more pedestrians than cyclists at each count site, we are more likely to see large variations in specific counts between sites.

Finally to button up the statistical requirements of this project we made use of a Poisson distribution, our objective with this was to calculate the probability of counting more than 120 pedestrians at each site over a 1 hour period during 7am-10am. Since the Gaussian distribution can be used to approximate the Poisson distribution for large values of λ, as mentioned previously, we have calculated this probability using this method as well. This approach will allow us to test the validity of this claim by plotting the difference between

the values.



(a) Gaussian and Poisson calculated probabilities



(b) Difference between Gaussian and Poisson probabilities

The points on these plots are sorted in ascending order λ. From plot (a) we can see that the probability of seeing more than 120 pedestrians increases with our value in lambda, which is a trivial observation, as it makes intuitive sense that we are likely to see 120 counts if the mean number of counts in that period is 200-300. We also see that for this case the Gaussian distribution gave a very good approximation of the probability. Very little difference is seen in plot(a), and in plot(b) we see a plot of the difference between the two distributions with noticable deviations only seen where the mean counts was 80-120, with the maximum deviation being 0.004325 which is a great result.

4.2 Pedestrian Footfall

4.2.1 Exploratory Data Analysis

After the data preparation and feature engineering had been completed for the footfall count data in DCC, it was important to gain some insight into the data before applying machine learning. Although descriptive statistics Are important, they wont be discussed in any great detail here just due to the abundance of attributes in the dataset, these can instead be found in the jupyter notebook. To get an idea of how busy each street is, a bar graph was created in order to visualize the mean number of counts recorded on each street over the three-month period.

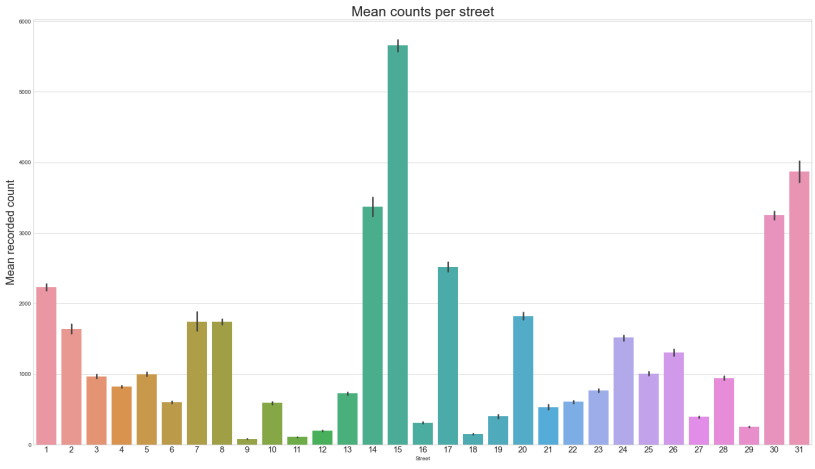
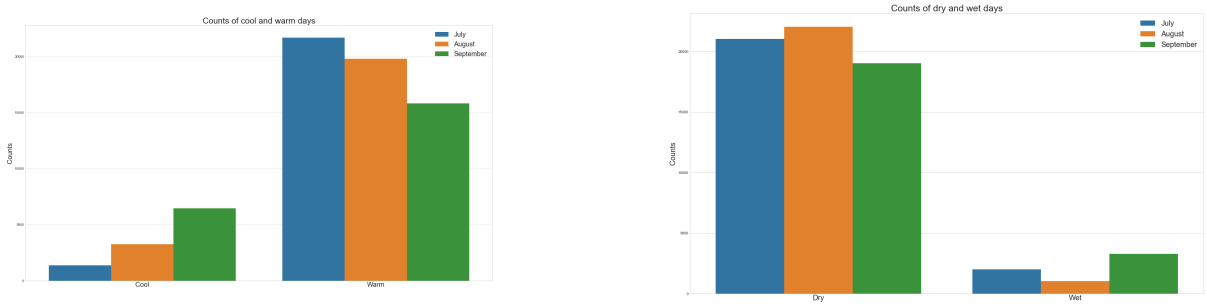


Figure 11: Mean counts recorded on each street

From this plot we see that Grand Canal Street upper and Westmorland Street see the greatest footfall, and D'oiler Street and Dawson street see the lowest footfall. This outcome aligns with expectations, The area surrounding grand canal street is populated by large tech and finance companies that generate a lot of employment, and Westmorland Street is the main connecting street between North and South Dublin city. D'oiler and Dawson street are likely neglected due to the proximity of more popular shopping and tourist locations nearby, with Dawson being overshadowed by Grafton Street and D'oiler street being more of a backstreet and located parallel to Westmorland street.

As we are dealing with weather data in this project, it would be interesting to know the frequency with which wet/dry or cool/warm days occur. It would also be nice to see if there is any connection between the month of the year and the recorded weather conditions. To analyze this, we produced a count plot showing the frequency of weather conditions grouped by the month in which they were recorded.



(a) Breakdown of cool and warm days

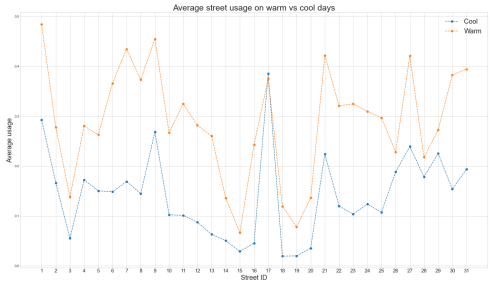
(b) Breakdown of wet and dry days

Upon reviewing the breakdown of weather conditions, we see almost exactly what we would expect. The occurrence of warm days decreases steadily over the months, with us seeing the highest count in July and the lowest counts in September, and conversely the occurrences of cool recordings increase steadily over the months, also with us seeing the greatest number in September and the lowest count in July. Similarly, the occurrences of wet recordings are the lowest in the summer months and higher in September, and the opposite being true for the dry recordings. This all makes sense as the summer months are typically warmer and drier, with September being the start of the autumn season, where the days get shorter and rain or cool weather starts to become more common.

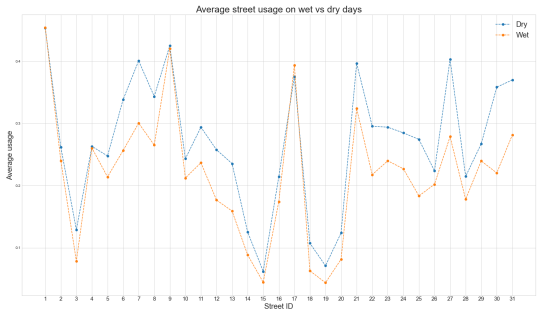
One may wonder if these weather conditions have a noticeable effect on the usage recorded at each counter location. To investigate this, we simply produced a plot of the average usage recorded at each site when it was wet or dry and cool or warm weather. These plots should serve to give us an idea how the use changes due to weather, and are found in figure 13.

While the results are along the lines of what we expected, they are still slightly surprising. Plot(a) shows the average usage on cool vs warm days. Although it remains the same at street 13, Grafton street, this is not so surprising, as Grafton street is known to be a popular tourist & shopping street and is in close proximity to offices. The largest difference is seen at street 7, which is surprising as it is in such close proximity to high traffic locations such as Trinity College and Grafton street. Plot(b) shows the same data, but now for wet and dry conditions. While there is still a definite difference in average counts on wet and dry days, its not as large as in the previous plot. The average usage is the same on many more sites, with only relatively small differences seen in others. The main conclusion that we can draw from these plots is that Irish people are more adverse to cool weather than they are wet weather

when it comes to making use of the streets for walking.



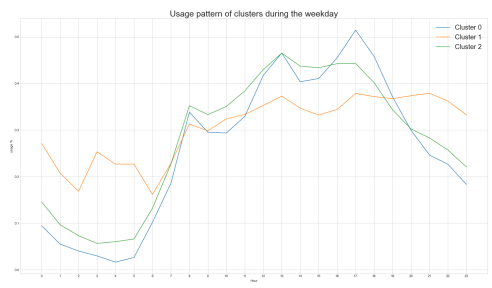
(a) Average street usage on warm vs cool days



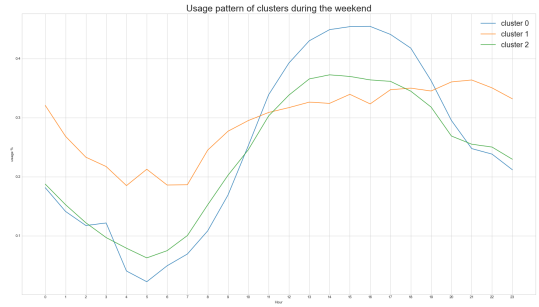
(b) Average street usage on wet vs dry days

Figure 13

One interesting area of investigation revolves around the usage patterns of streets involved in this study. To determine whether streets can be split into clusters related to their usage patterns, k-means clustering was used with the number of clusters, k, set to three as determined by the elbow method. The data was grouped by hour and cluster, with the corresponding average usage being calculated. Plotting this should reveal any prominent patterns that are present, these can be seen below for both weekdays and weekend counts.



(a) Weekday usage patterns for clusters



(b) Weekend usage patterns for clusters

Unfortunately this clustering analysis fails to show a significant difference between usage patterns of these streets. During weekdays we see peaks in each trend line at roughly 8am, 1pm, and 5pm. These peaks can very likely be accounted for by workers walking to work

at 8am, leaving for lunch at 1am, and walking home at 5pm. Cluster 1 shows the smallest peaks at these times, indicating that this may be an area where there are fewer office buildings or places of employment; however, notice that the street usage remains more consistent throughout the day.

The weekend cluster data do not show similar peaks at 8am, 1pm, and 5pm, further indicating that these were due to office workers' activities. The usage patterns of clusters 0 & 2 are quite similar, with a steady increase in usage during the day until it peaks at 3pm, likely due to casual shopping activity. After this usage begins to decrease, with a steep drop off after 6pm showing a minimum at 5am. Meanwhile the streets in cluster 1 see more steady activity up until 9pm before it begins to drop off, this could be an indication of a more vibrant night scene being present here. To provide a visual for the distribution and locations of these counters in Dublin city a map has been produced using folium, showing the counter locations and cluster groups.

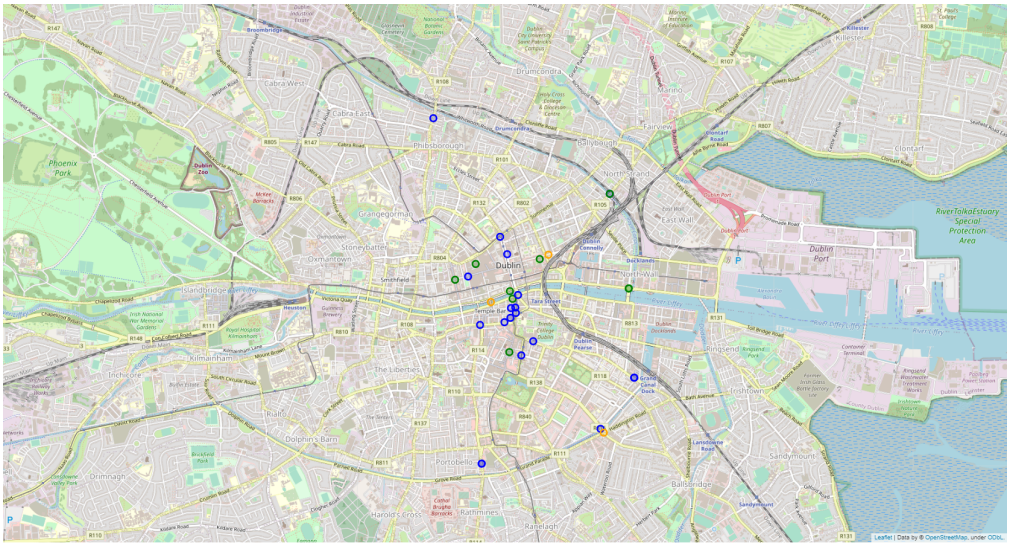


Figure 15: Map of counter locations in clusters

The last port of call before advancing to applying machine learning to this data is checking the data for correlation between attributes. This step can help to give us an idea of which attributes grow linearly with each other and which will have the greatest or lowest feature importance scores, enabling us to select the attributes that will return the greatest model accuracy scores.

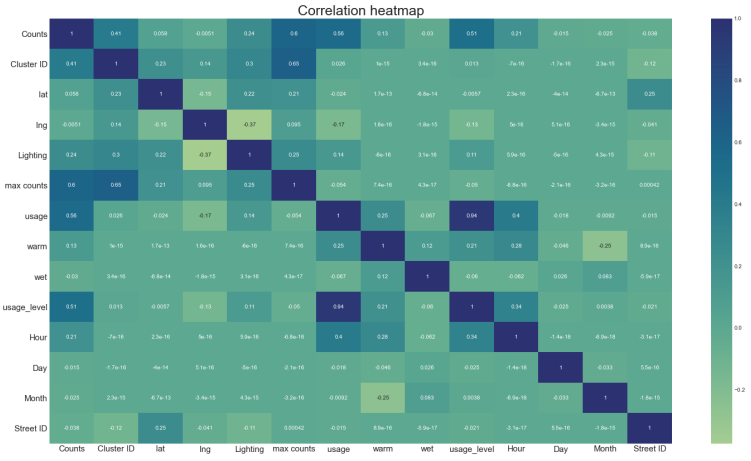


Figure 16: correlation between attributes

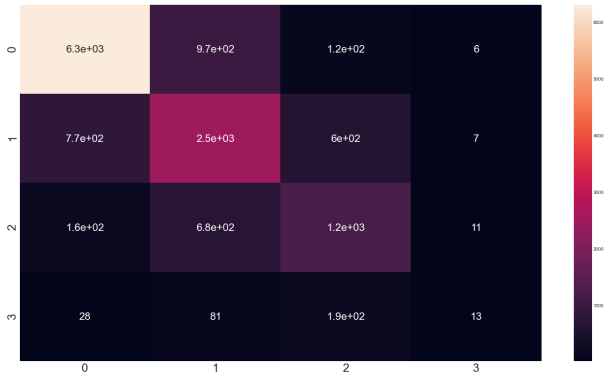
Across the board, we see very low correlations. The only times we see high correlation are when comparing an attribute to itself or another related through calculations, usage & usage level for example. When one looks exclusively at attributes that are independent of these relationships with the class attributes, the correlation is very low. There is a maximum correlation of 0.34 between usage level and hour of the day, and nothing else worth mentioning. Due to this, we will select all independent attributes as an input for our model & run it twice, removing the lowest importance features on the second run & comparing model outputs.

4.2.2 Machine Learning

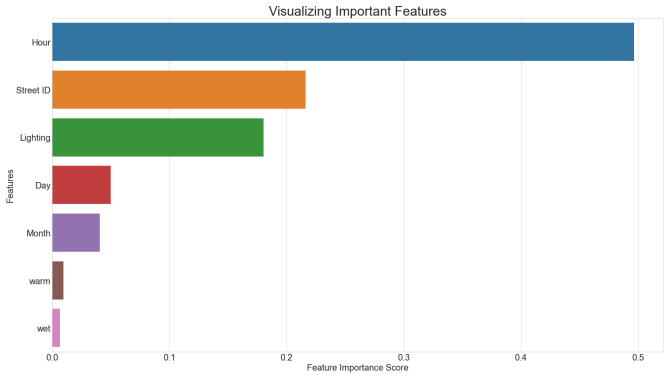
Following the completion of the EDA on this dataset, predictive analysis was carried out to try to predict the level of usage of any street at a given time of day and under varying weather conditions. Binning the usage level into categories and making predictions for this will provide enough information so that informed decisions can still be made regarding which street will be particularly busy or quiet at a given time. Since this is now a classification problem, we have a broader "error range" on predictions meaning that our accuracy score is likely to be much higher than that of a regression approach.

4.2.2.1 Decision tree

The first algorithm applied to the dataset is a decision tree classifier. In order to ensure that this model was as accurate as possible, the halving gridsearch function from sklearn was used. This tested the data on 1260 models & output the best performing parameters. The results of applying a training split to this data can be seen below.



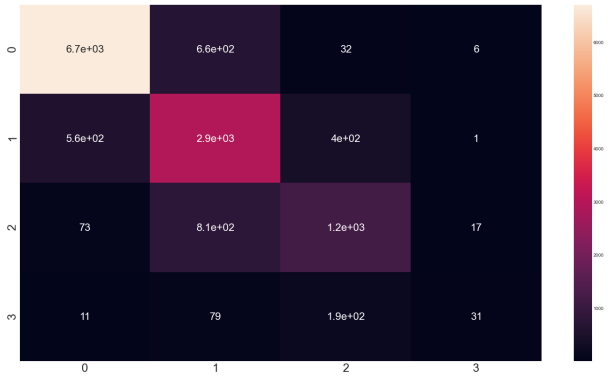
(a) Decision tree confusion matrix



(b) Decision tree feature importance

	precision	recall	f1-score	support
0	0.87	0.85	0.86	7393
1	0.59	0.64	0.62	3892
2	0.58	0.59	0.58	2093
3	0.35	0.04	0.07	312
accuracy			0.73	13690
macro avg	0.60	0.53	0.53	13690
weighted avg	0.73	0.73	0.73	13690

This model had a respectable accuracy score of 73.46% . The classification report shows that the model does struggle however to correctly classify data into group 2, and specifically group 3. As shown by the precision & recall scores in the classification report above. The precision gives a measure of how many of the identified positives are actually positive, meanwhile recall indicates how many positive cases were correctly predicted based off all positive cases in the data. By looking at the confusion matrix, we gain better insight into the classification errors that pull this model down. We can see that there are a large number of errors in each class label, but label 3 is the hardest for the model to identify with most predictions placed under class label 2. This error could be due to the over-representation of data in the dataset that falls under the 0 & 1 category. According to the feature importance plot, the weather data has very little impact on the outcome of the model. Due to this, the 'wet' and 'warm' will be dropped from our model.



(a) Decision tree confusion matrix

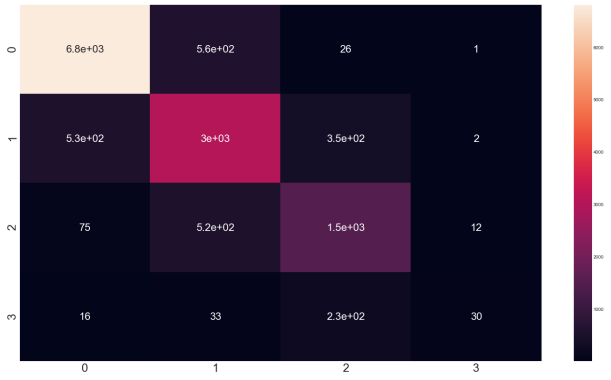
	precision	recall	f1-score	support
0	0.91	0.91	0.91	7393
1	0.65	0.75	0.70	3892
2	0.66	0.57	0.61	2093
3	0.56	0.10	0.17	312
accuracy			0.79	13690
macro avg	0.70	0.58	0.60	13690
weighted avg	0.79	0.79	0.79	13690

(b) Decision tree feature importance

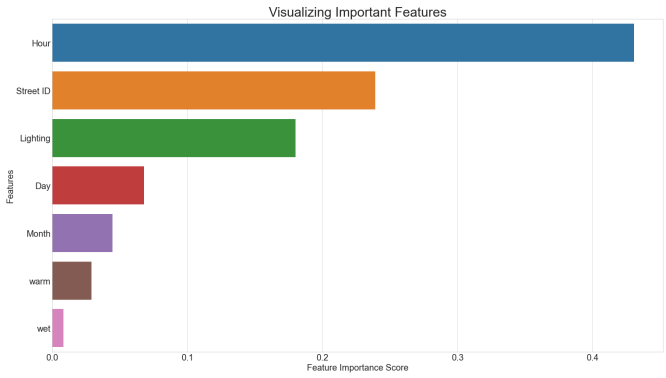
Above are the results of this feature-corrected model. The most noticeable difference is the increase in model accuracy, which now stands at 79.31%. Unfortunately the model still struggles to correctly predict class labels 2 & 3, although both precision and accuracy have both increased for nearly every class label, these still remain much higher for classes 1 and 2.

4.2.2.2 Random Forest

The second model that we implemented was the random forest classifier. This was selected as it is still a tree based algorithm, however we have the opportunity to see if an ensemble of decision trees results in better predictions for the class label, thus increasing overall prediction accuracy. The optimal parameters were once again found using the the halving gridsearch cross-validation algorithm. Like above we initially found that while the module preformed well when including all attributes, with an accuracy score of 82.80%, the feature importance once again displayed low reliance on weather conditions. However, removing these attributes had a negative impact on the model reducing the accuracy to 82.72%, so we will not go into any detail on that run here and instead you can see the results in the notebook.



(a) Random forest confusion matrix



(b) Random forest feature importance

	precision	recall	f1-score	support
0	0.92	0.92	0.92	7393
1	0.73	0.77	0.75	3892
2	0.71	0.71	0.71	2093
3	0.67	0.10	0.17	312
accuracy			0.83	13690
macro avg	0.76	0.63	0.64	13690
weighted avg	0.83	0.83	0.82	13690

Although this model does boast an overall better performance than the decision tree, it still suffers from similar issues as there is a very large classification error rate regarding the class label 3 predictions. while in every area it boasts higher precision and recall scores, we see that only 10% of the of the total positive cases in the data are correctly predicted, and only 67% of the label 3 predictions are actually correct. Again we see in the confusion matrix that many of them are still being incorrectly classified as label 2. Although the random forest classifier has been able to overcome some of the bias toward lower class values in the dataset, there is obviously still a way to go.

4.2.2.3 Final Model Selection

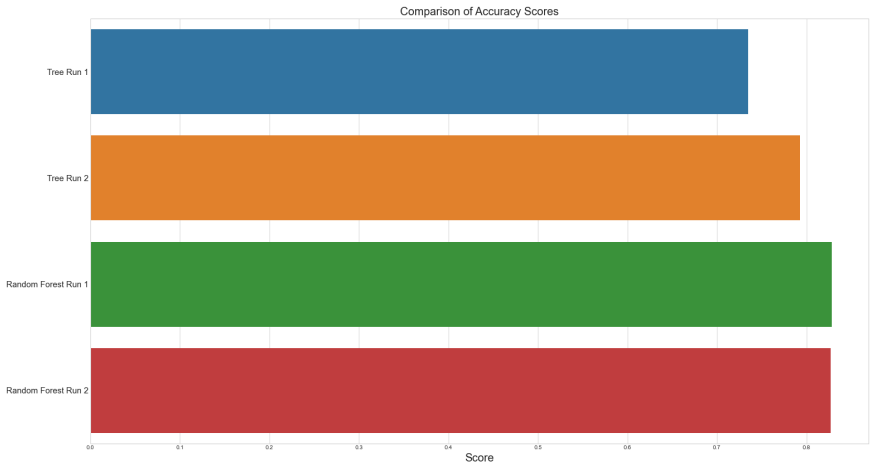


Figure 20: Accuracy score comparison for model runs

Looking at the results of both of these models, it is apparent that both fall down in the same areas, specifically in incorrectly classifying data as label three. Although both models were weak in this area, both models ended up classifying label 3 correctly 10% of the time, however random forest is more accurate as 67% of all label 3 predictions were actually correct, meanwhile its only 56% for decision tree. The overall score of the random forest classifier was also higher at 82.8%, so this is objectively the best model, of those tested, to predict the result based on the input data.

5 Conclusion

During this project, we focused on analyzing the number and patterns of pedestrians coming into dublin city in the mornings & how they utilize the streets during the day, along the way we also looked at the cyclists that come to the city. We saw that 25.44% of the total traffic entering the city between 7 and 10am is accounted for by pedestrians, and 13.04% by cyclists. Count sites 8 & 30 boast the highest percentage of clclists and count sites 8 & 33 the highest percentage of pedestrians. Descriptive statistics were obtained for the numbers of cyclists and pedestrians entering the city which showed that while a much higher number of pedestrians would be expected to enter the city due to the median value obtained, the number of cyclists is far more predictable thanks to the narrower IQR. Finally, a Poisson distribution was used to calculate the probability of seeing 120 pedestrians go past each counter in a 1 hour period between 7am and 10am; this probability approached 1 as the mean increased. Since the normal distribution can be used to approximate Poisson for large λ, we also calculated the probability using this and saw very very small deviations in values when λ became greater than 120.

Next, we took a look at the pedestrian counts in Dublin city centre. We found that the highest mean counts are recorded on Grand canal street & Westmorland street, with the lowest on D'oiler & Dawson street. Weather was taken into account and shown to have an effect on street usage, although surprisingly being cool or warm outside was more of a deciding

factor on the street usage than wet or dry days. This implies that pedestrians are more likely to stay off streets if its cold outside than if its raining, which was an unexpected outcome. In order to try to find patterns in street utilization, a clustering algorithm was applied to the data. The data was split into three clusters, where we unfortunately saw little difference in the usage patterns. Clusters 0 and 2 were very similar differing only in the number of counts, meanwhile cluster 2 indicated a similar behavior, however, the minimum counts were much higher and the maximum much lower. Peaks are seen at 8am, 1pm, & 5pm during weekdays, whereas usage over the weekends seems to be much more uniform. Due to this, it is likely that these peaks are as a result of the activities of office workers. The correlation map for the dataset revealed very low correlation between independent attributes, meaning attributes that aren't directly used to calculate the usage level or other attributes. These weak relationships made it difficult to carry out feature selection, so instead all independent features were included in the initial models and tuned after an initial run.

The two models selected to predict the usage level of Dublin streets at a given hour of the day are the decision tree classifier, and random forest. These were initially run with all attributes where they saw an accuracy score of 73.46% and 82.72% respectively. The weather related attributes had the lowest feature importance, so the models were then run without these attributes, and the accuracy scores changed to 79.30% and 82.8%. Overall, the random forest gave the best overall score only classifying roughly $\frac{1}{5}$ of data entries incorrectly, with it struggling most to classify high usage levels possibly due to the abundance of low and medium low data entries.

6 Appendix

6.1 Footfall Counter ID Key

Street	Street ID
Aston Quay/Fitzgeralds	1
Bachelors walk/Bachelors way	2
Baggot st lower/Wilton tce inbound	3
Baggot st upper/Mespil rd/Bank	4
Capel st/Mary street	5
College Green/Bank Of Ireland	6
College Green/Church Lane	7
College st/Westmoreland st	8
D'olier st/Burgh Quay	9
Dame Street/Londis	10
Dawson Street/Molesworth	11
Grafton Street / Nassau Street / Suffolk Street	12
Grafton Street/CompuB	13
Grand Canal st upp/Clanwilliam place	14
Grand Canal st upp/Clanwilliam place/Google	15
Henry Street/Coles Lane/Dunnes	16
Liffey st/Halfpenny Bridge	17
Mary st/Jervis st	18
Newcomen Bridge/Charleville mall inbound	19
Newcomen Bridge/Charleville mall outbound	20
North Wall Quay/Samuel Beckett bridge East	21
North Wall Quay/Samuel Beckett bridge West	22
O'Connell St/Parnell St/AIB	23
O'Connell st/Princes st North	24
Phibsborough Rd/Enniskerry Road	25
Richmond st south/Portabello Harbour inbound	26
Richmond st south/Portabello Harbour outbound	27
Talbot st/Guineys	28
Talbot st/Murrays Pharmacy	29
Westmoreland Street East/Fleet street	30
Westmoreland Street West/Carrolls	31

6.2 Class Label Key

Class label	Level	Criteria
0	Low	$>= 0, < 0.25$
1	Medium-low	$>= 0.25, < 0.5$
2	Medium-high	$>= 0.5, < 0.75$
3	High	$>= 0.75, <= 1$

References

Brownlee, Jason (Apr. 2016). *Bagging and Random Forest Ensemble Algorithms for Machine Learning*. URL: https://machinelearningmastery.com/bagging-and-random-forest-ensemble-algorithms-for-machine-learning/.

Chowdhury, Kaushik Roy (July 2020). *KNNImputer — Way To Impute Missing Values*. URL: https://www.analyticsvidhya.com/blog/2020/07/knnimputer-a-robust-way-to-impute-missing-values-using-scikit-learn/#:~:text=The%5C%20idea%5C%20in%5C%20kNN%5C%20methods,neighbors%5C%20found%5C%20in%5C%20the%5C%20dataset..

S.Buuren (2022). URL: https://stefvanbuuren.name/fimd/sec-MCAR.html.

Sampaio, C. (July 2022). *K-Means Clustering with the Elbow method*. URL: https://stackabuse.com/k-means-clustering-with-the-elbow-method/.

Stephanie (Apr. 2021). *Continuity Correction Factor: What is it?* URL: https://www.statisticshowto.com/what-is-the-continuity-correction-factor/.

UCLA (2022). URL: http://www.socr.ucla.edu/Applets.dir/NormalApprox2PoissonApplet.html#:~:text=Poisson(100)%5C%20distribution%5C%20can%5C%20be,1*100%5C%20%5C%3D%5C%20100).