JUNE 10, 2020 Name: Nikhil H

DATA MODELLING AND PRESENTATION

Contents

An Abstract / Executive summary:

The study is related to prediction of location based on the RSSI signal strength values detected by the 13 ibeacons located in the library. The model was trained on two classifiers decision tree and KNN and was compared. The decision tree was better in predicting compared to KNN with highest accuracy of 30% as it is better at classifying the categorical data.

Introduction:

The objective of this project is to build classifiers to predict an indoor location based on RSSI readings from 13 iBeacons. The data was collected in Waldo Library, Western Michigan University using an iPhone 6S. The data sets were sourced from UCI Machine Learning Repository. RSSI measurements are negative values. Bigger RSSI values indicate closer proximity to a given ibeacon. The figure of the layout of library is also provided which gives more information about the placements of the ibeacon in the library.

Methodology:

Task 1- DATA PREPARATION:

Data Retrieving

Load the csv data using the pandas read_csv() function with specific parameters.

Use comma for separator parameter sep(,) **Shape of the Data:**

Check the dimensions of the dataset to verify the number of rows and columns using the df.shape function. The dataset is as expected 1420 rows and 15 attributes which is the same number of records and columns in csv file.

Drop the unrequired features:

We will drop the date column as it is not necessary for classification. Drop the date column using the df.drop( ) function.

Extra-whitespaces

A lambda function is used to strip the white spaces, based on the condition that the strip function will be applied if the data type of the attribute is 'object'(as it contains strings), else the strip function won't be applied.

UPPER CASE

A lambda function is used to cast the text data to upper case, based on the condition that the upper() function will be applied if the data type of the attribute is 'object'(as it contains strings), else the upper() function won't be applied.

Missing values and check datatype

Check the number of missing values in each of the columns by using the df.info() function. As displayed in the output there are no missing values as all the columns are displayed with non-null.

The datatype of target attribute location is object and the datatype of 13 descriptive features are integers.

Typos

Check the different location values using the df['location'].unique() function. From the output we can conclude that there are no misspellings or typos in the data.

Sanity checks

Check for the number of unique values in the location attribute using the df['location'].nunique() function. There are 105 levels of target feature but according to the map 18 * 21 = 375 possible levels

Check for the unexpected values using the unique() function. There are some target features (location) containing V and W. They are W15 and V15 which are not expected as they are not present in the map. So that means actually there are 18(ibeacons) * 23 = 414 possible target values

Data transformation

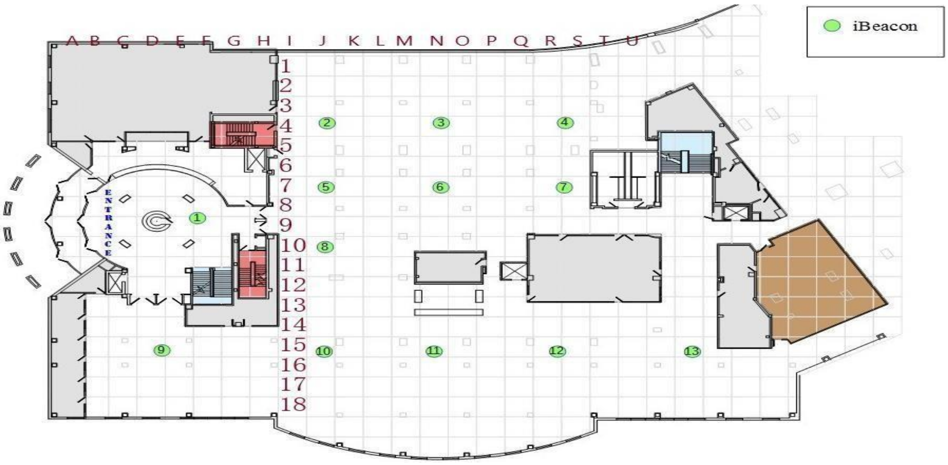
Transforming the negative signals to positive values by adding 200 to all the values, which will turn all the negative numbers to 0 and positive numbers. As all the signals are measured in the same range we can do this. EX: 200 - 78 = 122 which is the value in 1st row for b3006. So a value 0(which was originally -200) will mean no Signal was detected.

TASK 2.1: DATA EXPLORATION :

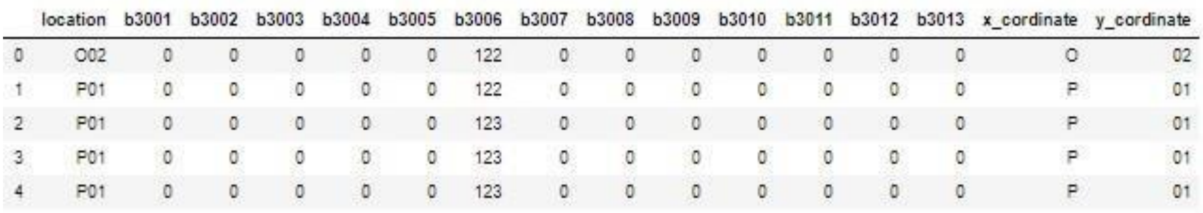
Use the df.describe() function to check the column wise summary for each of the ibeacon values. According to the summary RSSI values ranges from (145) highest proximity to (0) out of range and b3009 has the highest RSSI value of (145). The lowest mean of RSSI value is in b3001.

1. Exploration of the attribute location:

As there are 105 unique values in location it is difficult to map or get insight from the location attribute. So we will split the attribute into x_coordinate(A to U) and y_coordinate(1 to 18) as displayed in the below image:



We will name the column with letters as x_cordinate and column with numbers as y_cordinate and add it as a new column to the oringinal dataframe. So the dataframe will have totally 16 attributes as displayed in the below image:



	location	b3001	b3002	b3003	b3004	b3005	b3006	b3007	b3008	b3009	b3010	b3011	b3012	b3013	x_cordinate	y_cordinate
0	O02	0	0	0	0	0	122	0	0	0	0	0	0	0	O	02
1	P01	0	0	0	0	0	122	0	0	0	0	0	0	0	P	01
2	P01	0	0	0	0	0	123	0	0	0	0	0	0	0	P	01
3	P01	0	0	0	0	0	123	0	0	0	0	0	0	0	P	01
4	P01	0	0	0	0	0	123	0	0	0	0	0	0	0	P	01

Plot a bar chart and check the location which has most number of records in the x_cordinate and the y_cordinate.

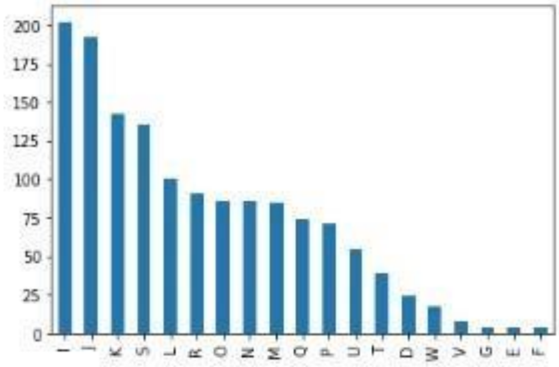


Figure 1

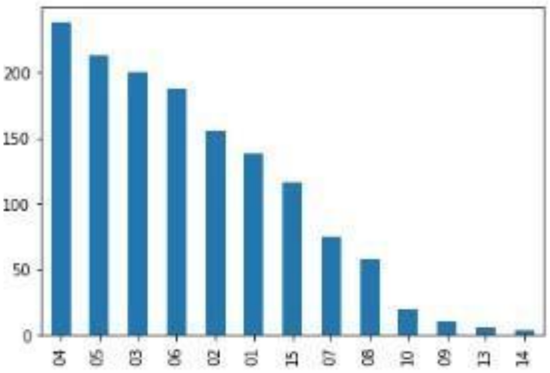


Figure 2

Analysis of figure 1: The more number of records are between the I and U locations in the decreasing order as displayed in figure 1.

Analysis of figure 2: The more number of records are between the 01 and 06 locations of y_cordinate in the decreasing order displayed in figure 2.

Exploration of the individual variables:

1) **Exploration of ibeacon b3001:**

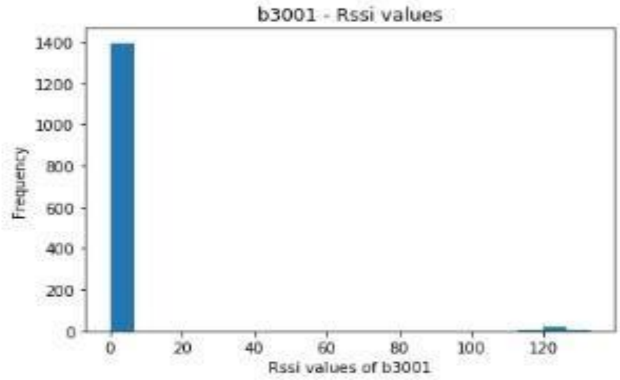


Figure 3

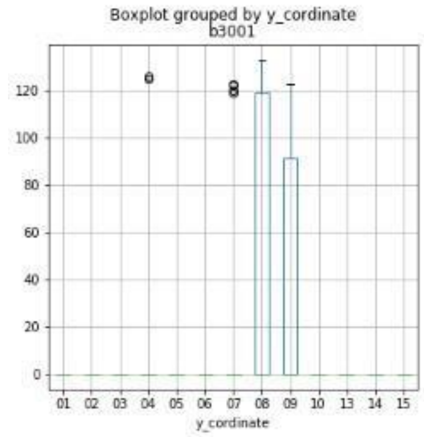


Figure 4

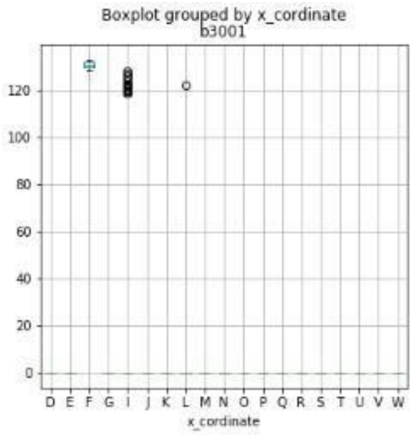


Figure 5

Analysis of ibeacon b3001: The histogram(Figure 3) shows very less number of instances where it is detected. As the ibeacon b3001 is placed near the entrance it makes sense that it will be less detected. But the strong signals from this beacon is detected at x-coordinates F,I,L (Figure 5) and ycoordinates 4 , 7 , 8 , 9 (Figure 4) as displayed in the boxplot.

2) Exploration of ibeacon b3002:

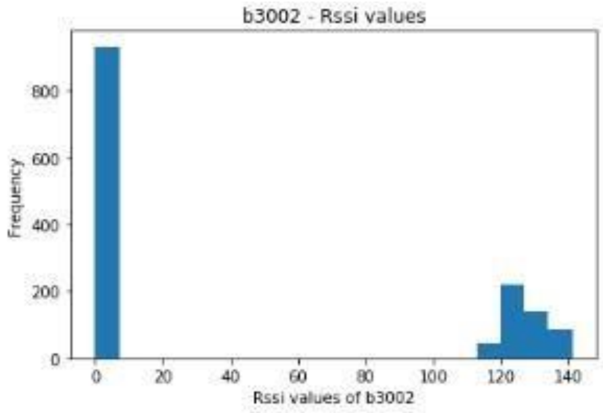


Figure 6

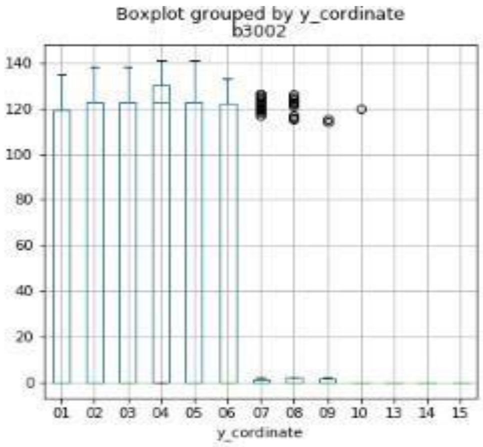


Figure 7

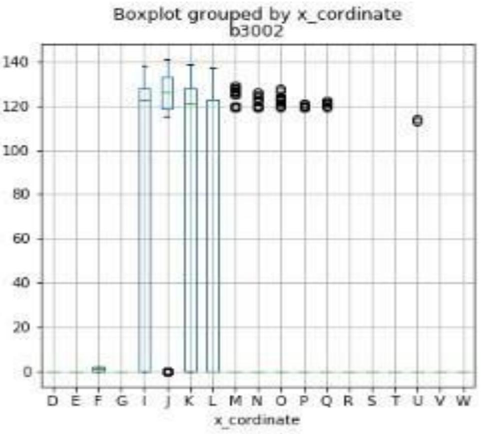


Figure 8

Analysis of b3002: It detects high number of RSSI signals at different locations as displayed in the histogram (Figure 6). It also has high signal strength detected in various coordinates of x (Figure 8) and y (Figure 7). There is also a signal strength detected at U which could be an outlier, because considering the distance between x-coordinate U and the ibeacon b3002 it is rarely possible to get an RSSI signal strength of 110.

3) Exploration of ibeacon b3003:

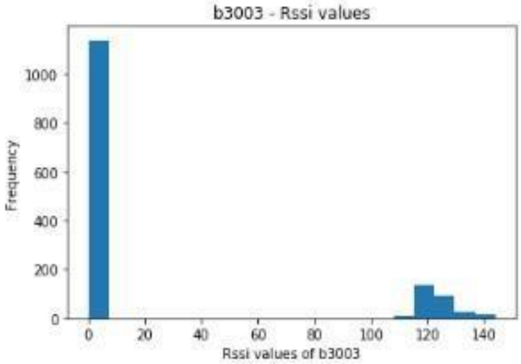
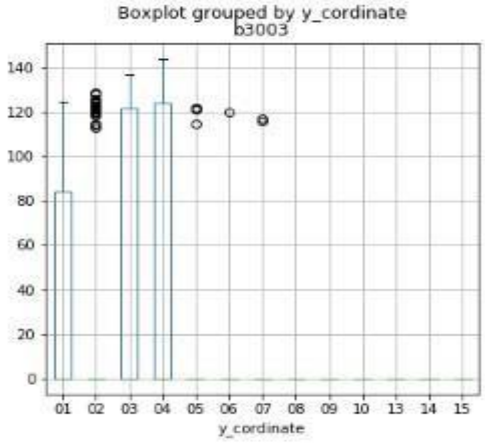


Figure 9



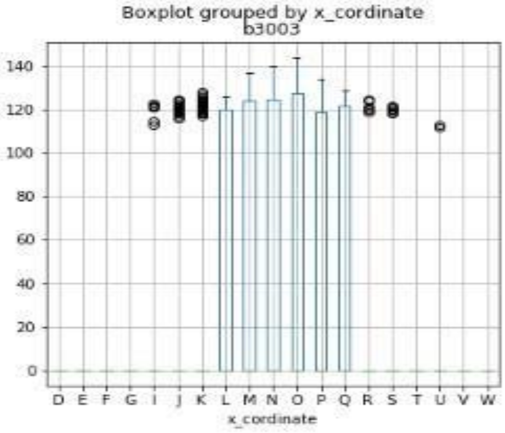


Figure 10

Figure 11

Analysis: RSSI Signals for b3003 is detected at various locations as displayed in the histogram (Figure 9). For the y-coordinates the RSSI signals are detected at 1 to 7(Figure 10) and in x-coordinates they are detected from I to U (Figure 11). The highest variance of signal strength is at M to O in xcoordinate and 3 to 4 in y-coordinate.

Exploration of ibeacon b3004:

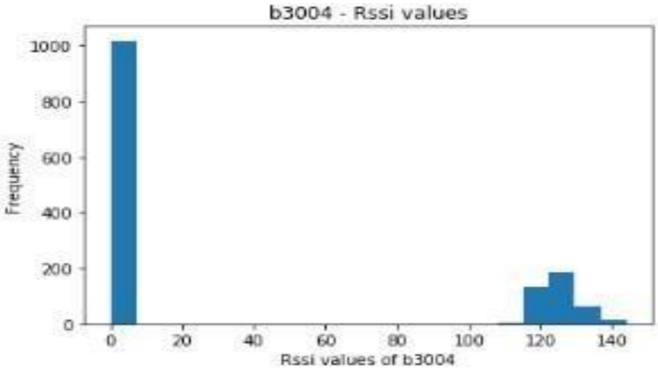


Figure 12

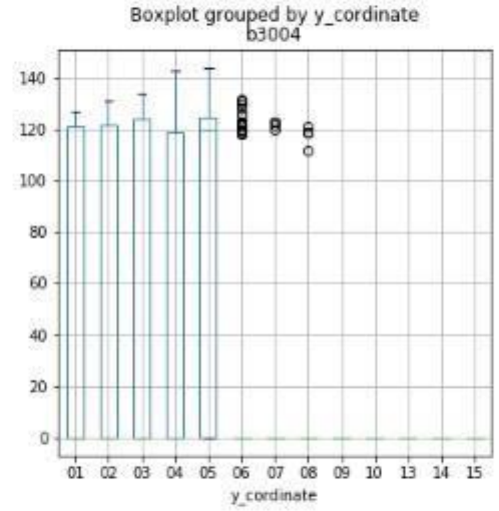


Figure 13

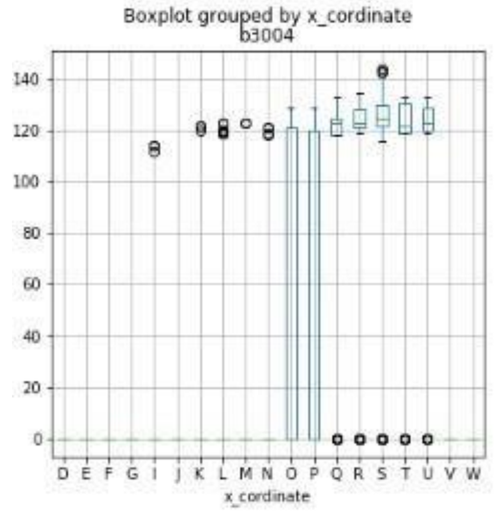


Figure 14

ANALYSIS: The RSSI values of ibeacon 4 are detected at similar locations to ibeacon 3 as displayed in the histogram (Figure 9). The highest variety of RSSI signals is detected at S in x-coordinate (Figure 10) and 3,4,5 in y-coordinate (Figure 11) .

Exploration of ibeacon b3005:

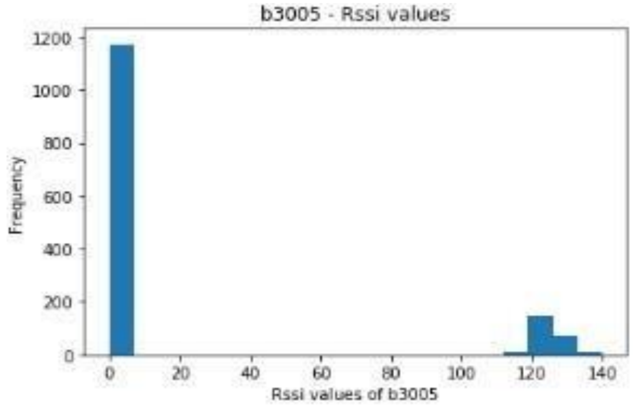


Figure 15

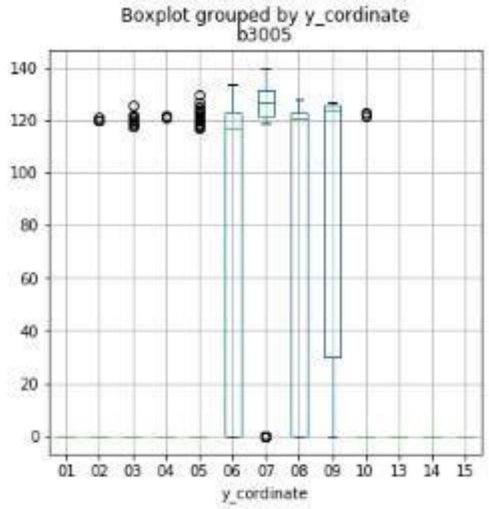


Figure 16

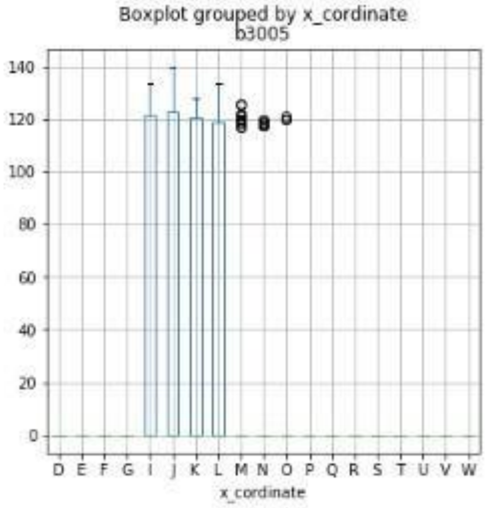


Figure 17

ANALYSIS: The RSSI signals are detected at various locations as displayed in the histogram (Figure 12).There are high variety of signals detected between I and O in x-coordinate (Figure 14) and 7 in ycoordinate (Figure 13) as displayed in the boxplot. The y coordinate 7 has high RSSI values because the ibeacon b3005 is located near the y-coordinate location.

Conclusion on Analysis of each ibeacon:

It is clear from the analysis of RSSI signals of ibeacon that there is a relationship between the placement of an ibeacon and the signals detected. There are many instance where a signal from ibeacon is not detected (RSSI value=0) out of range. Most of the signal strengths seem to be distributed around 125 and the more instances where signal is detected the distribution looks like a normal distribution as displayed in some of the above histograms. There are some unusual Signal strengths detected like the b3002 is detected at the x-coordinate value U.

TASK 2.2 : Explore the relationship between pairs of attributes :

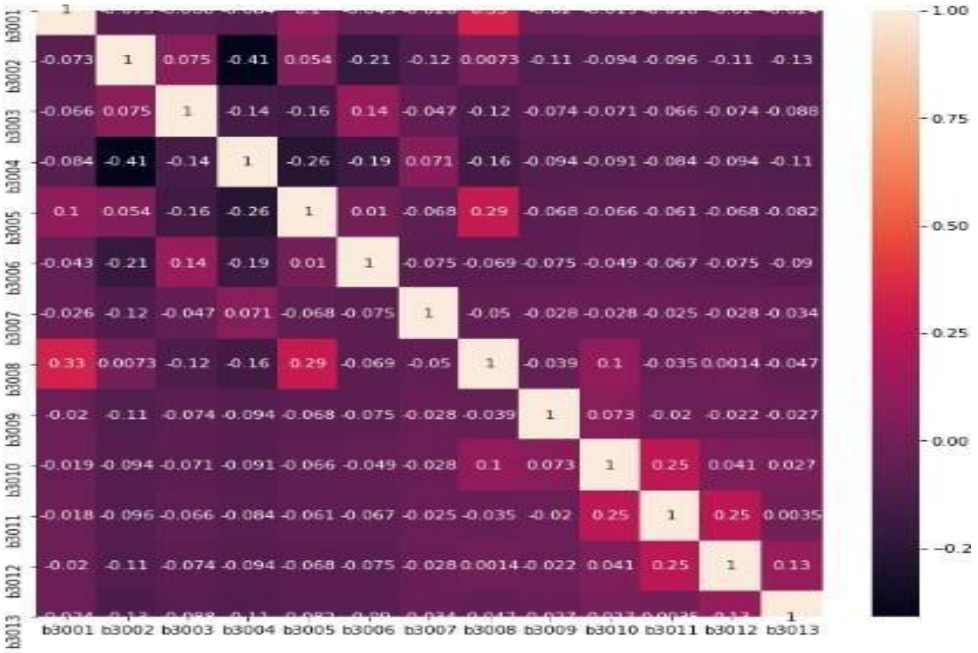


Figure 15

ANALYSIS: The heatmap displays the correlation values ranging from -1 to +1. There exists some correlation between some of the attributes. The ibeacons b3004 and b3002 have the highest negative correlation(they are inversely proportional). The b3008 and b3001 have a positive correlation so as the RSSI value of b3008 increases the value of b3001 also increases.

Task 3: DATA MODELLING

1) Decision Tree Classifier for Model Building:

Step 1) Feature Engineering and model selection:

We need all the attributes from b3001 to b3013 as we want to predict the location based on the RSSI signal strength value detected by each ibeacon. We will use decision tree for classification.

Step 2) Training the model:

Split the dataset into training and testing data which consists of the RSSI values for the 13 ibeacons and should contain the labels for target variables which is the Location. So the model should be trained to predict the location based on the RSSI values detected by the ibeacons. The Decision tree classifier will be used to train the model. The training data will consist of 80% and the test data will consist of 20%.

Step 3) Model Validation and Selection:

The advantage of using K fold cross validation is it uses the entire Train data. We need to select a model which has a good predictive power and generalizes well to the unseen data. To achieve this the classification error rate (percentage of mislabelled observations in the test data set) should be less. The best way to find optimal values of k is to try all possible values of k(while cross validation) and choose the value of k with highest accuracy. We will consider the k-value = 1 which has an accuracy of 44.01%. So the training data is predicting well.

Step 4) Applying the trained model to unseen future data:

The remaining unseen 20% testing data will be considered for prediction. The fit.predict() function will be used to apply the trained model on the testing data. A confusion matrix will be created and the observed and the expected values will be compared. The result is the accuracy of the model prediction which is 30%. So the model is able to predict 30% of the locations accurately based on the RSSI values of the ibeacons.

2) KNN Classifier for Model Building :

Step 1) Feature Engineering and model selection:

We need all the attributes from b3001 to b3013 as we want to predict the location based on the RSSI signal strength value detected by each ibeacon. We will use KNN for classification.

Step 2) Training the model:

The data is split into 80% training data and 20% testing data. The training data which consists of the RSSI values of the 13 ibeacon will be the features and the target will be location. An initial value of 5 will be given to the KNN classifier. The model will be tuned to get a higher prediction later.

Step 3) Model Validation and Selection:

As the dataset is small we can use the K fold cross validation. The advantage of using K fold cross validation is it uses the entire Train data. We need to select a model which has a good predictive power and generalizes well to the unseen data. To achieve this the classification error rate (percentage of mislabelled observations in the test data set) should be less. The best way to find optimal values of k is to try all possible values of k(while cross validation) and choose the value of k with highest accuracy. We will consider the k-value = 1 which has an accuracy of 31.69%. So the training data is predicting well.

Step 4) Applying the trained model to unseen future data:

The remaining unseen 20% testing data will be considered for prediction. The fit.predict() function will be used to apply the trained model on the testing data. A confusion matrix will be created and the observed and the expected values will be compared. The result is the accuracy of the model prediction which is 19%. So the model is able to predict 19% of the locations accurately based on the RSSI values of the ibeacons.

Step 5) Tune the parameters to increase the accuracy:

We will keep n_neighbors = 5 and set weights = distance and p = 1. After tuning the parameters and training the model again the model accuracy increased to 26%. So the model is able to predict 26% of the locations accurately based on the RSSI values of the ibeacons.

3) Decision tree on the selected location which have high instance of getting detected

Now that we have found the classifier with 30% accuracy. We can increase the accuracy more by selecting the specific locations where the highest RSSI signals were detected by the ibeacons. Based on the x-cordinate analysis we can see that the number of instance of values detected is higher in R,S,I,J,K,L. Based on the y-cordinate we can group them on the basis of location mentioned as group 1 in exploration(2,3,4,5,6,7)

Analysis: So we can conclude that as the number of instances of these locations were compared to the other locations so the model accuracy was high. The model with location y_corinate(02,03,04,05,06,07) and x_coordinate(S,J,K,L,I) will have an accuracy of 40%

RESULTS:

Classifier	Model accuracy	K-fold accuracy
Decision Tree	30%	44.01%
KNN	26%	31.69%
Decision tree classifier for Selected features	40 %	42 %

DISCUSSION:

From individual or univariate visualisation of the ibeacons, we found that there is a clear tendency towards the placement of ibeacons and where the signals are detected. In addition, we are suspecting that there are some outlier values in the dataset as seen in the graphs of individual iBeacons, but since these values appear for more than half the iBeacons this could also be a natural tendency of the iBeacon signals. From multivariate visualization, we found that there is no or less linear correlation between the iBeacons. Moreover, we found that there are a lot of zero values in the data set which could be an important characteristic.

CONCLUSION:

From the comparison of the two classifiers decision tree and KNN we can conclude that decision tree has a higher accuracy in predicting location than compared to KNN. As decision tree works better with categorical data. The accuracy of the model with decision tree classifier is 30%. We can still increase the accuracy of the model if the decision tree is applied on the specific locations with high instances. Then the model accuracy would increase by 40%.

REFERENCES:

[1] dataframe, a. and Walkins, k. (2015)* adding one to all the values in a dataframe, Stack Overflow. *Available at: https://stackoverflow.com/questions/30794525/adding-one-to-allthevalues-in-a-dataframe

[2]* How to Create a Correlation Matrix using Pandas - Data to Fish *(2020). Available at: https://datatofish.com/correlation-matrix-pandas/

[3]Predicting location from RSSI signals, Rstudio-pubs-static.s3.amazonaws.com. *Available at: https://rstudio-pubs-static.s3.amazonaws.com/399185_dea8330*