Predictive Analytics for retransmission of Wi-Fi data

Abstract

The idea of *spectrum* relates to radio frequencies that are allocated to the mobile industry and other areas that communicate over airwaves. Spectrum is a sovereign asset that the government manages and issues licenses for its use. Those in the mobile industry and repeatedly demonstrating its potential to generate economic value and social benefit. They are continuously encouraging national regulators of the spectrum to release enough, affordable spectrum in a timely manner for mobile so the industry can continue to generate more economic value and social benefit. The spectrums bands themselves are all different. Different bands of spectrum are appropriate for different purposes. For example, low frequency transmissions can usually travel longer distances and pass through dense objects more easily. However, it the amount of data that can be transmitted over these lower frequencies is limited. While, higher frequency waves can transmit more data but are not as good at passing through dense objects. The task of allocating and licensing appropriate spectrum to services and sectors while maximizing the value generated becomes a challenging task. [5] Any information the licensing bodies can obtain regarding the usage of allotted spectrum is useful for maximizing the value from this finite resource, is of value. The prediction of the retransmission was considered with the idea that the methodology for predicting the retransmission could be used for occupancy. The method of prediction was explored using the random forest algorithm. This was done using the randomForest package in R and the randomForest function within this package. [12, 16]

Introduction

What is retransmission? When one party sends something to another party a copy is retained until the recipient acknowledges receipt. The send can automatically retransmit (resend) the data using the retained copy. Reasons for retransmission may be:

- No acknowledgement of receipt of original transmission of message
- Sender discovers transmission was unsuccessful
- Receiver knows expected data has not arrived and notifies send
- Receiver has received the data but it is in a damaged condition and asks sender to resend. [17]

When does packet loss occur? Packet loss occurs when one or more packets of data fail to reach their destination. Most common loss is due to network congestion. This means packets are arriving for a sustained period of time to a given router or network segment at a rate greater than it is possible to send through then the only option is to drop packets. A bottleneck is another reason for dropped packets. This occurs when a single router or link is constraining the capacity of the complete travel path or of the network travel in general. Other factors for packet loss or corrupted packets during transmission are, too weak radio signals due to distance or multi path fading; faulty network hardware; faulty network drivers. Packets can be intentionally dropped; by normal routing routines. [13]

What is the difference between 2.4GHz and 5GHz? 2.4 GHz is becoming more and more crowded. It is used by most wireless devices such as laptops, phones and tablets. Lower end wireless spectrum is used other devices as well such as cordless phones, garage door openers, baby monitors and more. 5GHz however, is better suited for devices such as laptops, phones or tablets because it can transmit larger amounts of data and is less congested. The drawbacks of 5GHz include it is less able to penetrate through solid walls and objects. Due to the congestion on 2.4GHz there is more chance of dropped connections and slow data throughput. But it is better for transmitting data over longer rangers and through walls and large objects. 5 GHz is more suited for "indoor" use. That is, ideally suited for connections inside the house. Due to the lack of congestion, higher data transmission rates and smaller effective range. However, as you move away from the access point the efficiency may decrease. [23]

Why channels 1, 6 or 11? In the 2.4GHz band channels 1, 6 and 11 are the only non-overlapping channels. Three main causes of interference are co-channel, adjacent channels and non-Wi-Fi. Co-Channel is when every client and access point on the same channel compete for times to talk. Adjacent channels occur when every client and access point on overlapping channels talk over each other. Non-Wi-Fi, when non 802.11 devices compete for medium access. Co-channel interference is a problem when there are too many Wi-Fi devices on the same channel. Adjacent channel interference occurs when channels can overlap and channel selection can be critical. Each channel on the 2.4GHz spectrum is 20MHz wide. The centres of which are separated 5MHz and the entire spectrum is only 100MHz wide. So 11 channels have to squeeze into 100MHz and thus overlap. There are three channels that do not overlap, 1, 6 and 11. Co-channel interference means devices take turns talking so the more devices on one channel the longer it takes a device to talk since it has to wait for its turn. [24]

What is predictive analytics? Predictive analytics is an area of data mining dealing with extracting information from data and using it to predict trends and behavior patterns. Unknown event is usually in the future but not necessarily. An example of something to be predicted in the past is identifying suspects after a crime or credit card fraud after it has occurred. Predictive analytics uses a wide variety of statistical techniques such from predictive modelling, machine learning, data mining that analyze current and historical facts to make predictions about future or otherwise unknown events. For example in business predictive models are used to exploit patterns found in historical and transactional data to identify risks and opportunity. The defining element is that predictive element is that predictive analytics provides a score, probability, for each individual "thing", customers, employee etc., in order to determine, inform or influence those making decisions. Some industries where predictive analytics is used are actuarial science, marketing, financial services, insurance, telecommunications, retail, travel, healthcare, child protection, pharmaceuticals, capacity planning. Credit scoring is a well-known application of predictive analytics. Scoring models process a customer's credit history, loan application, customer data etc. and then rank the person's likelihood of making future credit payments on time. The essence of predictive is to capture relationships between explanatory variables and predicted variables from past occurrences and using this to predict the unknown outcome. [15]

What is machine learning? Machine learning is a type of artificial intelligence (AI) providing computers with the ability to learn without being explicitly programmed. Focuses on development of computer programs that can teach themselves to grow and change when exposed to new data. It is similar to data mining, searching through data to look for patterns. Machine learning uses the data to detect patterns in data and adjust program actions accordingly. Machine learning algorithms are either supervised or unsupervised. Supervised algorithms apply what has been learned in the past to new data. Unsupervised algorithms draw inference from datasets. [11, 18, 21]

What is a random forest? A random forest is a collection of decision trees. A decision tree is a hierarchical or tree like representation of decisions. The technique iteratively breaks input data into two or more samples. This is repeated recursively until specified conditions are met. [6] A random forest is a collection of decision trees where each tree now has a vote in deciding the final class of an object. This means a random forest is an ensemble process. What makes a random forest random? In a dataset or data frame there are observations and variables. When creating a decision tree samples of the dataset are taken with replacement and a sample of variables is also taken for each dataset sample. This sampling is done randomly hence, a random forest. The advantages of using a random forest are that the chances of over fitting may be reduced, and there is higher model performance or accuracy. Random forest can be used for classification or regression problems depending on the type of dependent variable. In table 1 below the type of dependent variable is given and the corresponding use of the random forest. [7, 18]

Dependent Variable	**Decision Tree Class**
Factor	Classification
Numeric/Integer	Regression
Not available	Unsupervised

Table 1 Decision tree class based on dependent variable type.

Below is a simple and hopefully relatable example of a random forest.

You cannot decide which movie to watch so you ask your friend W if you will like movie X. You provide friend W with a list of movies, X1, X2… Xn, that you've seen and whether you liked them or not. This is a *labelled training set*. Regarding movie X, you ask friend W if you will like it. Friend W now asks you '20 questions' about the movie. E.g. Is Johnny Depp in the movie?
These questions are like the variables describing the movie X. Friend W is now a *decision tree.* However, friend W is a human and does not generalize movie preferences well. Friend W *overfits*. So instead you ask a bunch of friends to get more than one opinion as to whether you should see movie X or not. E.g. You ask friends W1, W2, W3, W4…in addition to the original friend W, and each votes on whether you will like the movie X or not, and hence whether you should see it or not. This is an *ensemble classifier*, in this case a *forest*. Now, the information you give each of your friends about your movie preference is slightly different so you do not end up with the same result. For example, perhaps you told friend W that you loved Finding Nemo but maybe you really just happy and in a good mood that day. So you do not want to tell all your other friends this fact. Maybe you said you liked Toy Story 3 but you actually really, really loved Toy Story 3. So you would perhaps tell some of your other friends this fact. In the end, you give each of your friends' slightly different versions of the data you gave friend W, so slightly *perturbed versions* of the original data. You do not change your response, i.e. the love or hate for a movie, just the intensity, i.e. more love or more hate towards a movie. You provide a *boot strapped* version of your original training data. For example, say you told friend W you liked Zootopia and The Secret Life of Pets but disliked Finding Dory. But then you told friend W1 you liked Zootopia so much you watched it twice and disliked Finding Dory but do not even mention The Secret Life of Pets. From this ensemble of friends' opinion and recommendations, you hope the errors get cancelled out in the majority. So your friends form a *bagged – bootstrap aggregated – forest* of your movie preferences. An example of the opinions, recommendations or conclusions your friends have come to based upon the data provided them:

Friend W: You like vampire movies

Friend W1: You like Pixar movies
Friend W3: You hate everything

One problem with the data is, say you loved Titanic and Wolf of Wall Street, it was not because you like Leonardo DiCaprio, but for other reasons. You do not want your friends to base their recommendations on if Leonardo DiCaprio is in the movie. So a random subset of all possible questions is allowed. (When building a decision tree, at each node some randomness is used to select the attribute to split on, i.e. by randomly selecting an attribute from a random subset) So, in this situation, your friends cannot ask you if Leonardo DiCaprio is in the movie whenever they want. So, perturbing the movie preferences injected randomness at the data level; making friends ask different questions at different times adds randomness at the model level. Now your friends form a *random forest*. [10]

The original objective that was presented was to predict the occupancy and retransmit rate of a 2.4 GHz Wi-Fi signal at a time X in the future where X = 1 minute, 1 hour, a day. Packets may be retransmitted when the occupancy is higher causing collisions between packets. With the time provided, the prediction of the retransmission was first considered with the idea that the methodology for predicting the retransmission could be used for occupancy. The method of prediction was explored using the random forest algorithm. This was done using the randomForest package in R and the randomForest function within this package.

Methods

For this project Wi-Fi data was considered that was already processed. So we were not working the raw data for this particular project. The variables provided are given in table 2. These were the variables provided at the end of Jun 2016. Some variables changed or were added since but those are not reflected here.

Variable	**Description**
Start_datetime	the date and time of start of the analysis period
Band_scan	values 2.4 GHz or 5 GHz
Time_resolution_hrs	Time interval / analysis period for the following quantities. 1 hour was used
Num_ap	the number of Aps observed in the time interval
Ssid	text label for the AP defined
Mac_source_address	MAC address for the AP
Total_clients	total number of client devices served
Channel_id	integer identifies for the Wi-Fi frequency channel: these are predefined across all 802.11 standards
Channel_centre_freq_mhz	inserted from channel list for convenience
Channel_bw_mhz	inserted from channel list for convenience
Channel_occupancy_data	percentage of time_resolution_hrs where traffic was observed; based on all packets
total_packets	total number of packets observed
total_packets_nonbeacon	total number of non-beacon packets observed with the ‘retry’ bit enabled

<table>
<tr><td>protocol_versions</td><td>supported 802.11 protocol modes as reported by the AP within</td></tr>
<tr><td>time_resolution_hrs</td><td>This is stored as a bit-wise record where, bit4=’ac’; bit3=’n’; bit2=’g’; bit1=’b’; bit0=’a’</td></tr>
<tr><td></td><td>Each protocol version advertised by the AP results in a ‘1’ being XORed with the corresponding bit in the record. For example:</td></tr>
<tr><td></td><td><table><tr><th>Observed</th><th>bit pattern</th><th>integer</th></tr><tr><td>‘a’</td><td>OOOO1</td><td>1</td></tr><tr><td>‘g’</td><td>OO1OO</td><td>4</td></tr><tr><td>‘ac’, ‘b’</td><td>1OO1O</td><td>18</td></tr></table></td></tr>
<tr><td>Total_data_packets_80211a</td><td>total number of data packets (packetType=2) observed using 802.11a</td></tr>
<tr><td>Total_data_packets_80211b</td><td>total number of data packets (packetType=2) observed using 802.11b</td></tr>
<tr><td>Total_data_packets_80211g</td><td>total number of data packets (packetType=2) observed using 802.11g</td></tr>
<tr><td>Total_data_packets_80211n</td><td>total number of data packets (packetType=2) observed using 802.11n</td></tr>
<tr><td>Total_data_packets_80211ac</td><td>total number of data packets (PacketType=2) observed using 802.11ac</td></tr>
<tr><td>Max_data_rate_mbps</td><td>maximum of the observed information data rates</td></tr>
<tr><td>Min_data_rate_mbps</td><td>minimum of the observed information data rates</td></tr>
<tr><td>Average_data_rate_mpbs</td><td>average of the observed information data rates</td></tr>
<tr><td>Std_data_rate_mpbs</td><td>standard deviation of the observed information data rates</td></tr>
<tr><td>Med_data_rat_mbps</td><td>median of the observed information data rates</td></tr>
<tr><td>Average_latitude_dd</td><td>average latitude across all observed packets in decimal degrees</td></tr>
<tr><td>Average_longitude_dd</td><td>average longitude across all observed packets in decimal degrees</td></tr>
<tr><td>Average_altitude_m</td><td>average altitude across all observed packets in metres</td></tr>
</table>

Table 2 Variables for given wifi data.

Data Processing and Results

As mentioned in the introduction, channels 1, 6 and 11 do not overlap. But for this particular setting all the channels between 1 and 11 were considered. That is, channel_id’s between 1 and 11 were considered and a band_scan of 2.4GHz only was considered.

At first glance the following variables were only considered,

Start_datetime
Mac_source_address
Total_retry_packets_nonbeacon
Total_data_packets_80211g
Total_data_packets_80211n
Average_latitude_dd
Average_longitude_dd,

where start_datetime was broken up into its components. The start_datetime was given in the following format:

e.g. 2016-05-02 00:00:03.7366040

This was divided up into year, month, day, hour, minutes and seconds.
To get an idea of whether there was any obvious relationship between the variables the following scatter plots were considered:

total_retry_packets_nonbeacon vs total_data_packets_80211g
total_retry_packets_nonbeacon vs total_data_packets_80211n
total_retry_packets_nonbeacon vs average_latitude_dd, average_longitude_dd,

These can be found in figures 1-4 below. There is no clear way to describe the relationship between the total_retry_packets_nonbeacon and total_data_packets_80211g and total_data_packets_80211n. The values seem to be concentrated around 0. For the graphs of total_retry_packets_nonbeaon vs average_latitude_dd and total_retry_packets_nonbeacon vs average_longitude_dd while it seems there are some latitude and longitude values that seem to have more retry packets, they are still within a narrow region, about 0.4 for latitude and 0.6 for longitude.

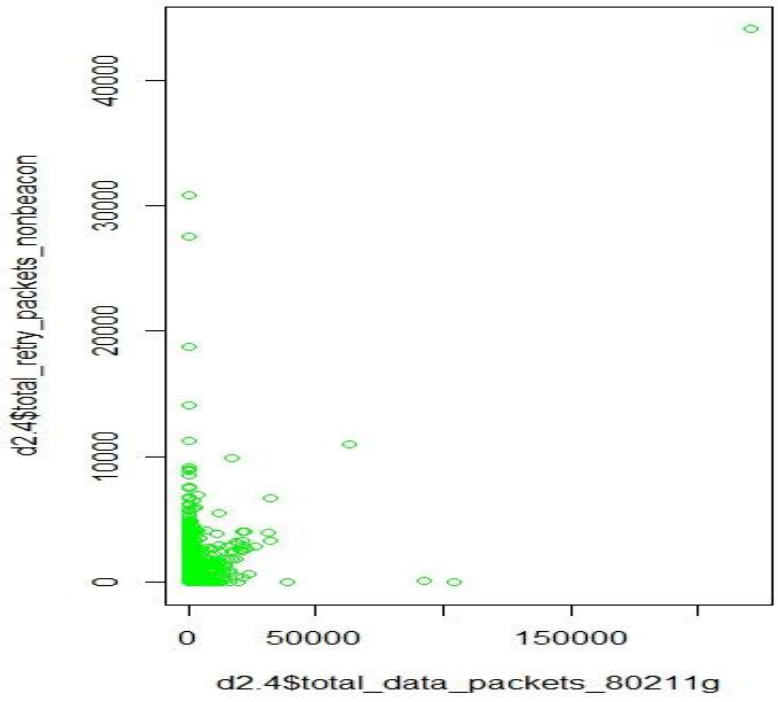

Figure 1 Plot of retry packets non-beacon vs total data packets 80211g

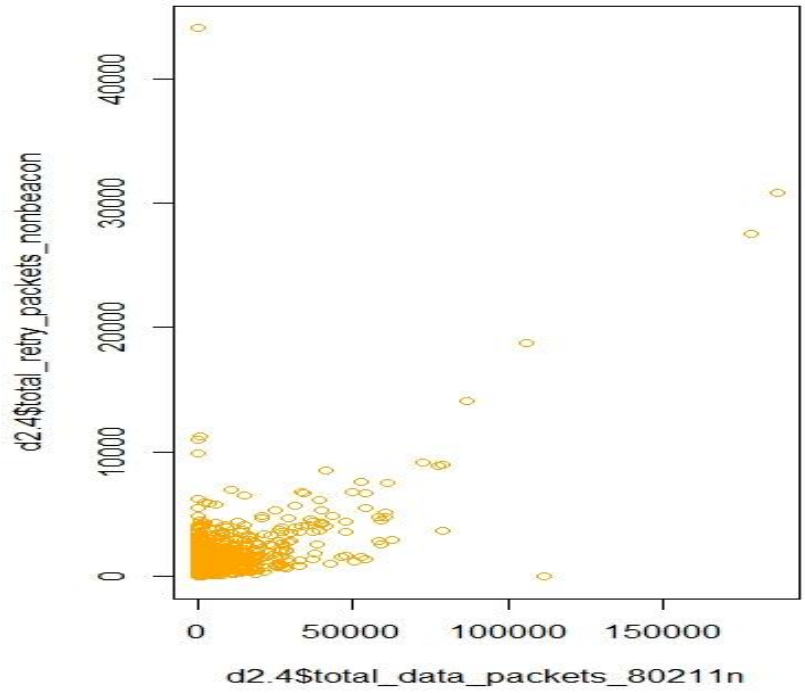

Figure 2 Plot of total retry packets non-beacon vs total data packets 80211n

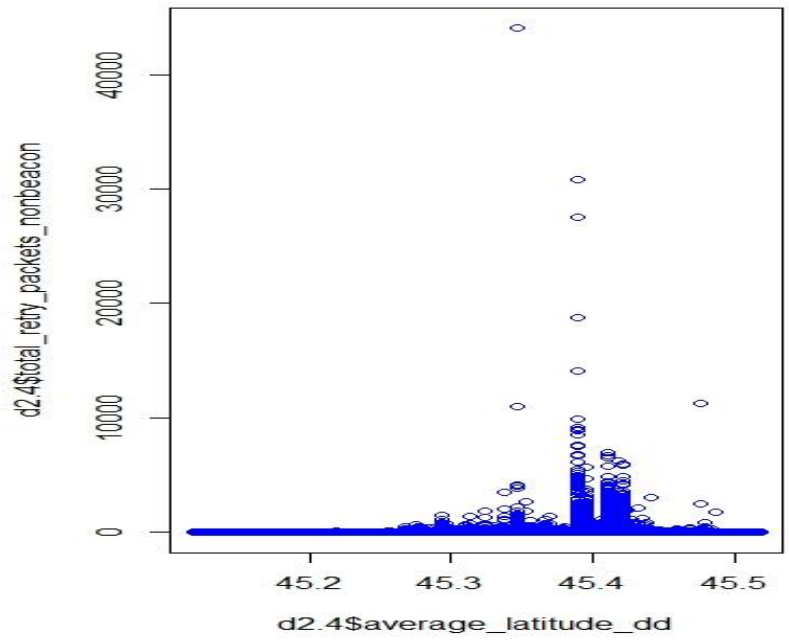

Figure 3 Plot of total retry packets non-beacon vs average latitude

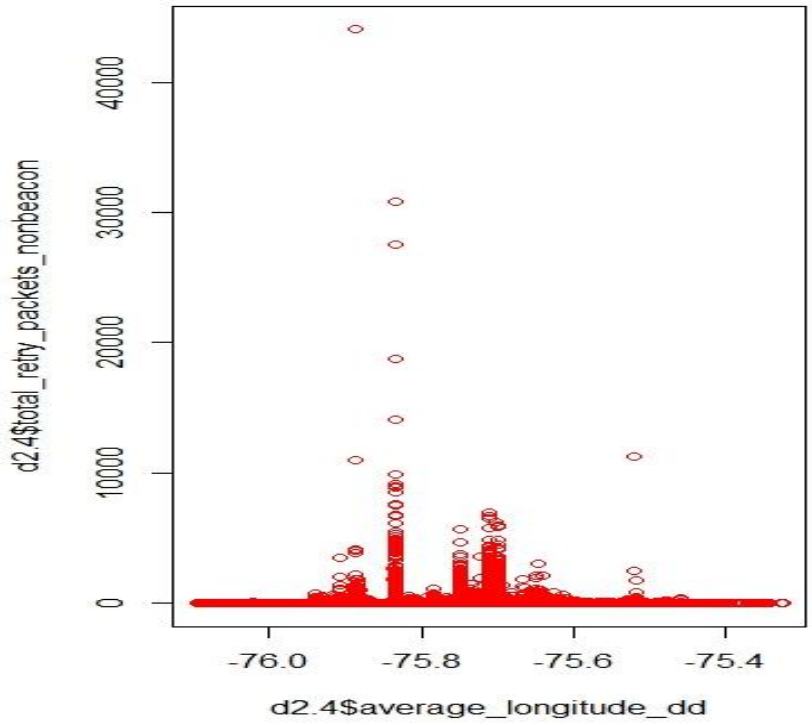

Figure 4 Plot of total retry packets non-beacon vs average longitude

The random forest algorithm was applied to the variables listed above, where total_retry_packets_nonbeacon was the dependent variable the remaining, start_datetime, mac_source_address, total_data_pacekts_80211g, total_data_packets_80211n, average_latitude_dd, average_longitude_dd were independent variables. (I don't have the code for this anymore because I changed it after I got the error. ☹) However, after running the algorithm on the following error occurred:

Error in randomForest.default(m, y, ...) : Can not handle categorical predictors with more than 53 categories.

After exploring the variables and attempts at changing the various parameters for the random forest algorithm the variable mac_source_address was the culprit. Mac_source_address is a categorical variable with 395155 distinct values or categories. Even after considering only non-null values of mac_source_address there were still 395117 distinct value or categories. This was too much for R and the random forest algorithm to handle. The mac_source_address variable was explored a little further in particular to determine if there were any "principal players" among the mac_source_address.

Plots of different variables against Mac_source_address were made to see if there were "principal players" among the mac_source_address (es). For example, a plot of the frequency of the mac_source_address was plotted to see if there was one particular mac_source_address that was particularly busy; total_retry_packets_nonbeacon vs mac_source_address; total_data_packets_80211g vs mac_source_address; total_data_packets_80211n vs mac_source_address; average_longitude_dd vs mac_source_address; average_latitutde_dd_ vs mac_source_address. There is no one mac_source_address that seemed to be a "principal player". These plots can be found in figures 5-9 below.

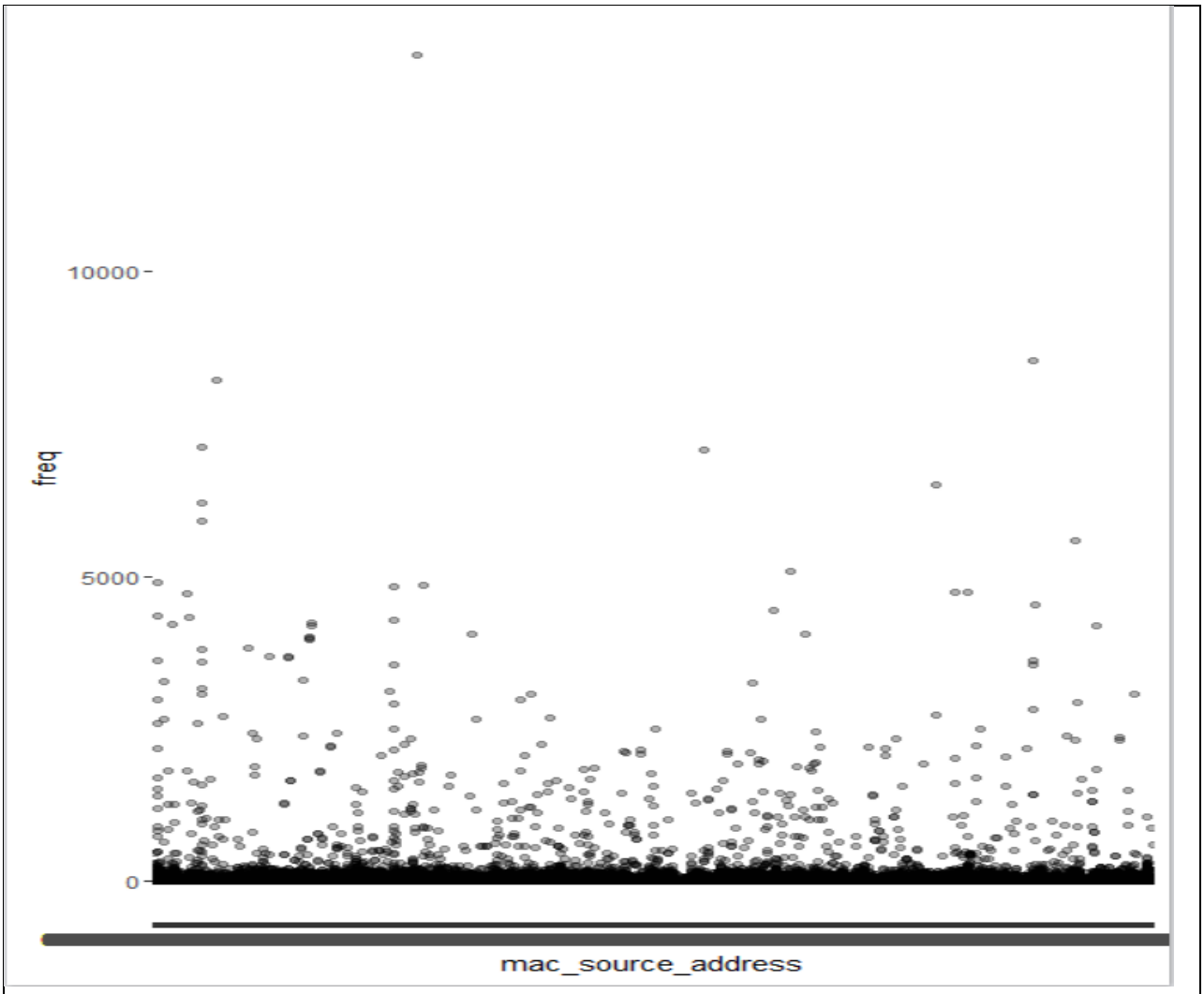

Figure 5 Frequency plot of mac source address

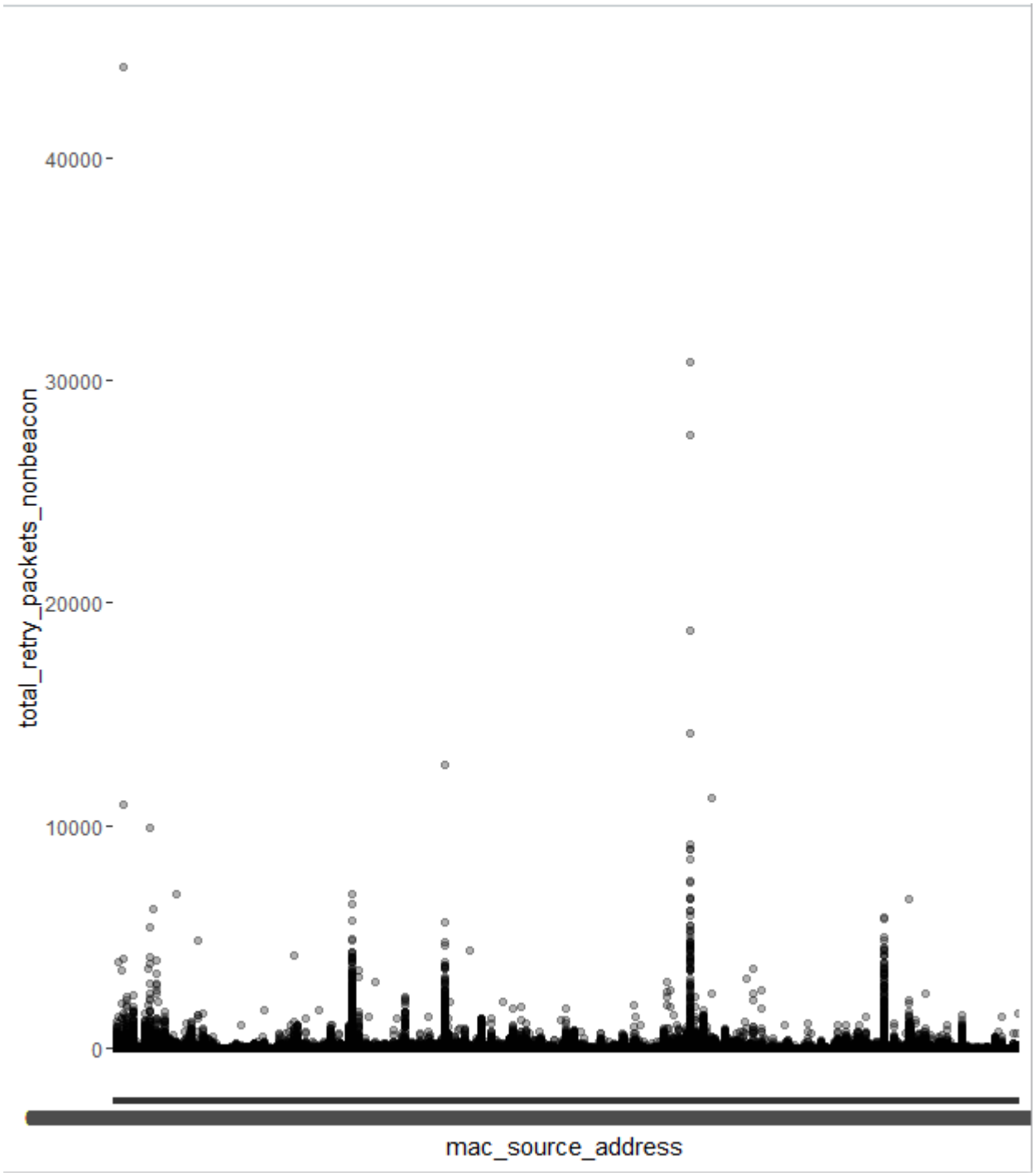

Figure 6 Plot of total retry packets non-beacon vs mac source address

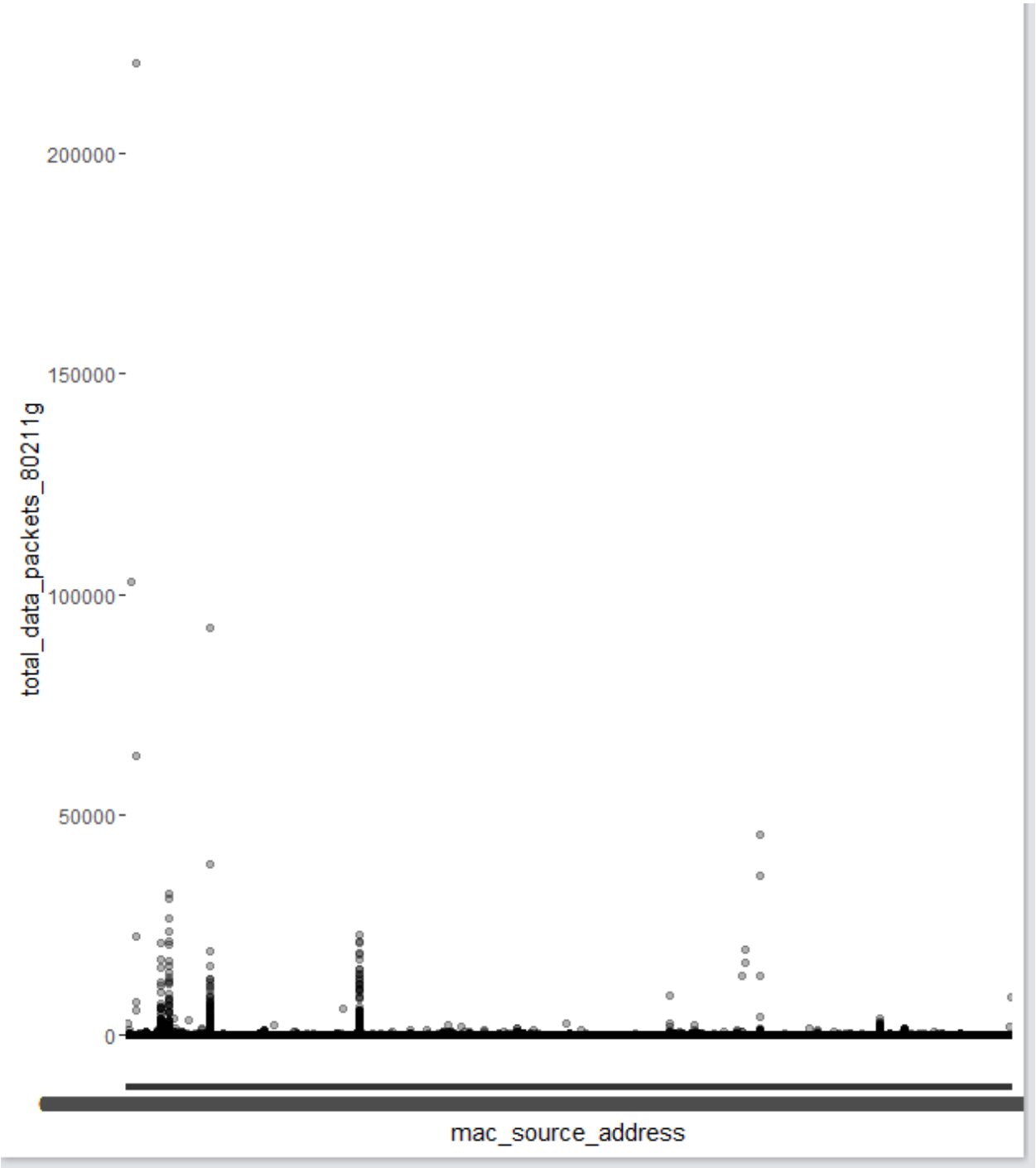

Figure 7 Plot of total data packets 80211g vs mac source address

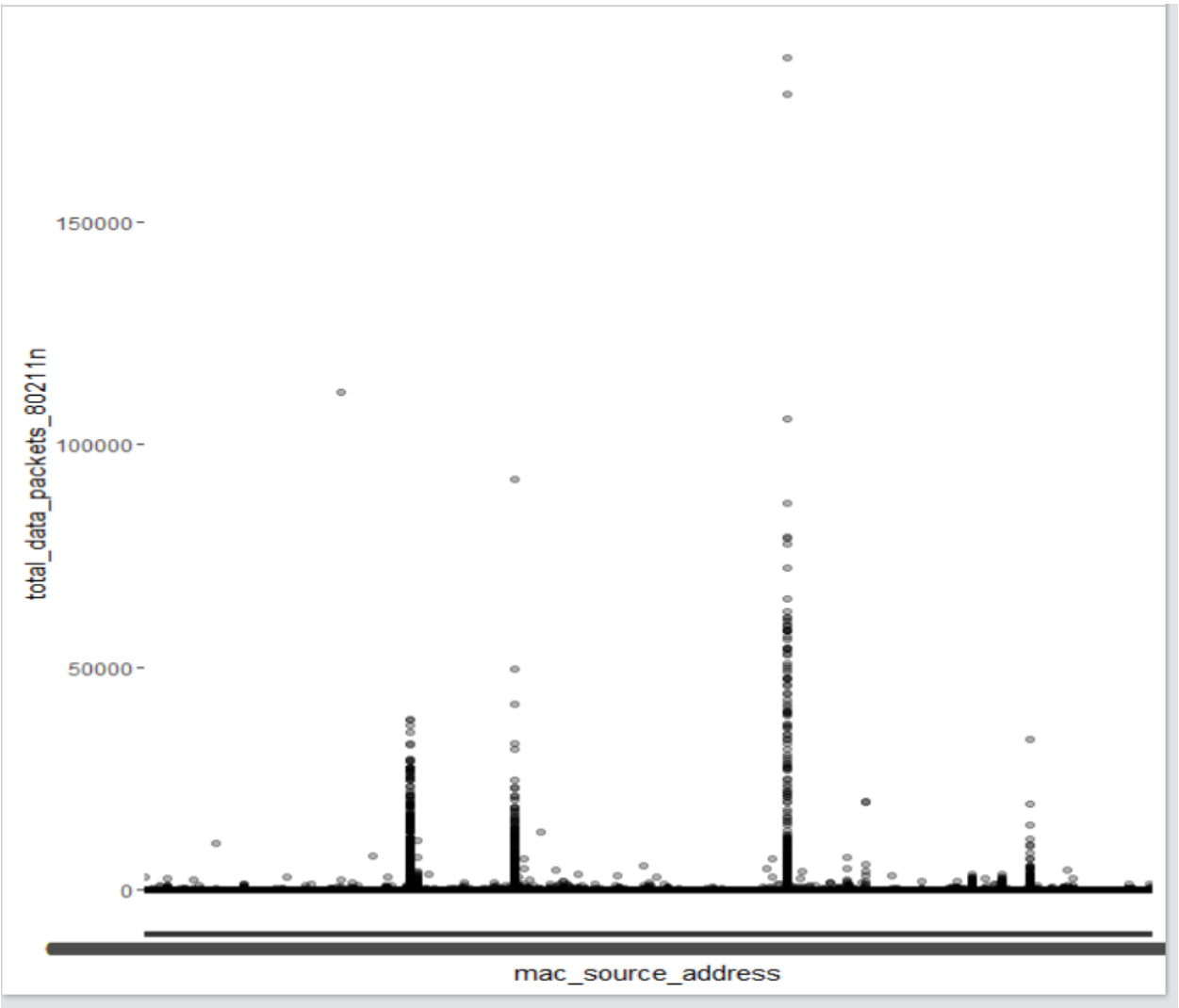

Figure 8 Plot of total data packets 80211n vs mac source address

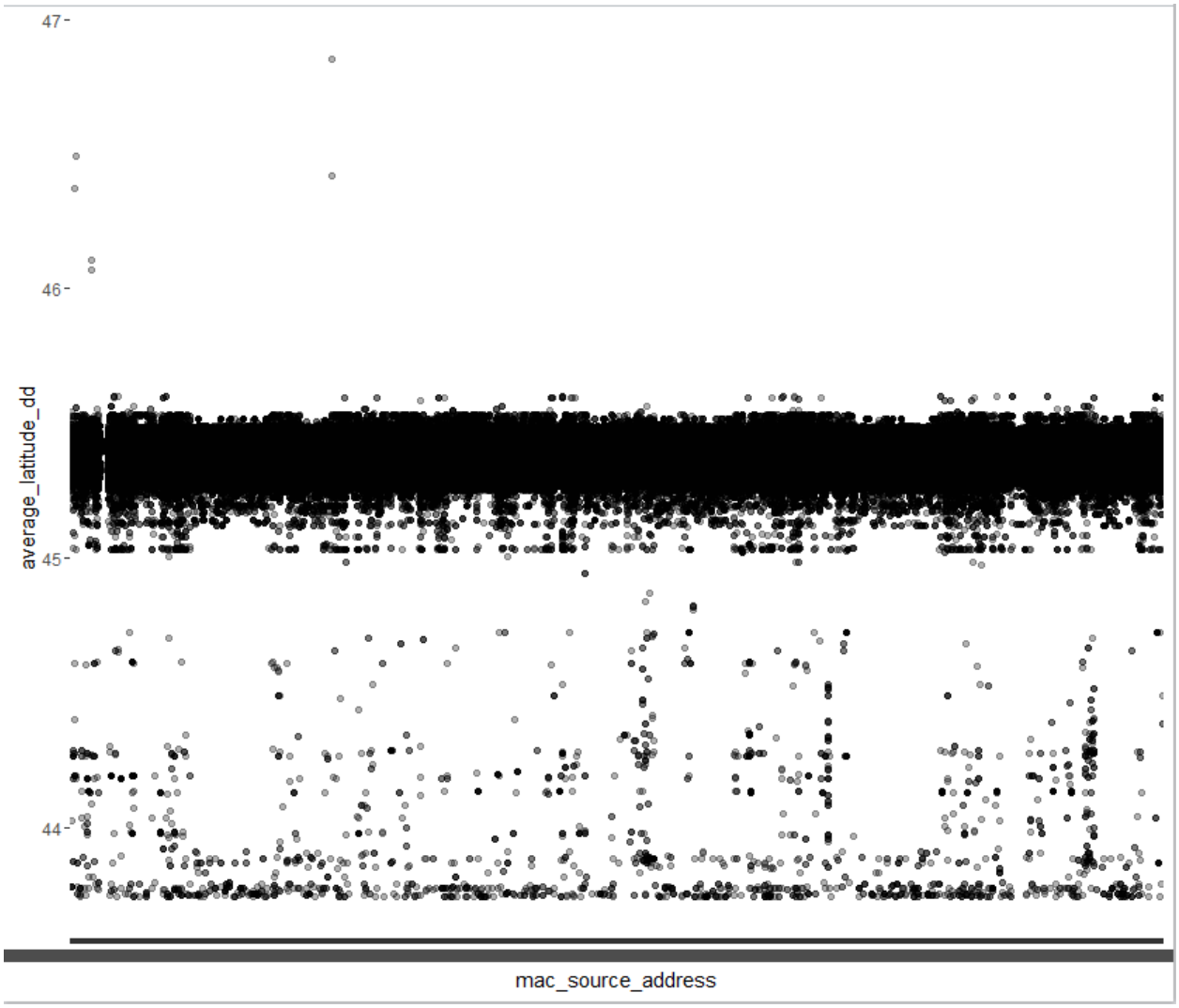

Figure 9 Plot of average latitude vs mac source address

There was no obvious sign that any one particular mac_source_address was a principal player. While in figures 6-8 it may seem as though there is not the case because there are so many distinct mac_source_addresses, over 395 000, that it is still a large number of mac_source_address that create those spikes in figures 6-8. Since there was no obvious sing that any one particular mac_source_address was a principal player, the mac_source_address was removed from the model.

Since the interest was in predicting retransmission, just knowing whether retransmission had occurred was determined to be sufficient information rather than an actual numeric value quantifying the retransmission. So the variable total_retry_nonbeacon, was converted into binary categorical variables, nonbeacon_cat which took on a value of 1 if there is a retransmission value; a value of -1 for NA; 0 otherwise. A frequency table for the new categorical variable nonbeacon_cat can be found in table 2. Most of the observations have a total_retry_nonbeacon value of 0 that is the packet does not require retransmission.

nonbeacon_cat	**-1**	**0**	**1**	**total**

frequency	7424	4002984	932615	4943023
percentage	0.15%	80.98%	18.87%	

Table 3 Frequency table for the derived categorical variable nonbeacon_cat. Nonbeacon_cat has a value of -1 when total_retry_packets_nonbeacon is null; nonbeacon_cat is 0 when total_retry_packets_nonbeacon is 0; nonbeacon_cat has a value of 1 otherwise.

Now the random forest algorithm was use for the prediction. The following model was the first considered:

nonbeacon_cat ~ average_latitude_dd + average_longitude_dd + total_data_packets_80211g + total_data_packets_80211n + hour1 + month1 + day1,

using the following code:

```
wifi.rf <- randomForest(as.factor(nonbeacon_cat) ~ average_latitude_dd + average_longitude_dd + total_data_packets_80211g + total_data_packets_80211n + hour1 + month1 + day1, data=trainData, ntree=5, keep.forest=FALSE, proximity=TRUE, importance=TRUE)
```

Note the use of proximity=TRUE in the above code. Among the parameters used in the random forest function in R, proximity is one of them. Proximity calculates the proximity measure, or closeness of rows, or observations, or a dataset. In the random forest setting if two observations occupy the same terminal node in a tree then the proximity of those two cases is increased by one.
The following error resulted from running the above random forest,

```
Error: cannot allocate vector of size 585.2 Gb
In addition: Warning messages:
1: In matrix(0, n, n) :
  Reached total allocation of 8125Mb: see help(memory.size)
2: In matrix(0, n, n) :
  Reached total allocation of 8125Mb: see help(memory.size)
3: In matrix(0, n, n) :
  Reached total allocation of 8125Mb: see help(memory.size)
4: In matrix(0, n, n) :
  Reached total allocation of 8125Mb: see help(memory.size)
```

Slight variations of this model was tried but still the above memory size error. Table 1 gives representative models considered including dependent and independent variables. Models 1-3 found in Table 1 gave the above memory size error. It was discovered that the use of proximity=TRUE caused the algorithm to create a matrix that in this case was very large, too large it seems for R to handle. Removing this matrix or changing proximity=FALSE and running the random forest algorithm was tried. This proved some success, i.e. no warnings. The "fourth model" below was considered. In this model and subsequent models, the *day* variable was changed to a *dayofweek* variable which is a categorical variable indicating which day it of the week it is. i.e. Monday, Tuesday, Wednesday, Thursday, Friday, Saturday and Sunday, rather than a numeric variable representing the date. Also, the month, second and minute variables were removed. There didn't seem to be much need for them at that time.

The **fourth model** considered was:
nonbeacon_cat ~ average_latitude_dd + average_longitude_dd + total_data_packets_80211g + total_data_packets_80211n + dayofweek

using the following code:

```
wifi.rf <- randomForest(as.factor(nonbeacon_cat) ~ average_latitude_dd + average_longitude_dd + total_data_packets_80211g + total_data_packets_80211n + dayofweek, data=trainData, ntree=100, proximity= FALSE, keep.forest=TRUE, importance=TRUE)
```

Note the use of proximity=FALSE in the above code. Below are the results from applying the random forest to the training set. Below is an example of the result of the random forest obtained from running the randomForest function in R on the data. Further details on the confusion matrix and importance table will discussed in the next model, model 5.

Variable	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
average_latitude_dd	34.2975	57.28672	61.28736	17150.345
average_longitude_dd	17.67292	68.8665	100.31747	21926.456
total_data_packets_80211g	141.2172	24.26665	121.72803	6266.342
total_data_packets_80211n	119.1736	25.48461	118.35772	2761.935
dayofweek	17.67292	53.32225	64.03148	3831.737

Table 4 Importance table for model 4.

```
Number of trees: 100
No. of variables tried at each split: 2

OOB estimate of error rate: 27.77%
Confusion matrix:
      0      1 class.error
0 51311  47670   0.4816076
1 30144 151124   0.1662952
```

The confusion matrix after prediction of the test data with the random forest created is given below.

```
          Predicted
Observed     0     1
       0 22278 20566
       1 12895 64633
```

The above model only considered our original short list of variables after some edits, i.e. removal of the mac_source_address, taking only the day of the start_datetime which was then converted into a categorical variable representing the day of the week, total_retry_packets_nonbeacon was converted into a binary variable indicating whether retransmission took place or not. Note the number of trees used, estimate of the OOB, out of box, error rate

Start_datetime
Mac_source_address
Total_retry_packets_nonbeacon
Total_data_packets_80211g
Total_data_packets_80211n
Average_latitude_dd
Average_longitude_dd.

Since the error seemed high in the fourth model above, the model was rerun but with all the variables to see if there would be any difference. Note that all models hence forth have included proximity=FALSE. A list of the models considered can be found in table 13.

The **fifth model** considered was:

nonbeacon_cat~channel_centre_freq_ghz+channel_bw_mhz+channel_occupancy_data+channel_occupancy_total+total_packets+average_latitude_dd+average_longitude_dd+ total_data_packets_80211a + total_data_packets_80211b+total_data_packets_80211g+total_data_packets_80211n+ total_data_packets_80211ac + max_data_rate_mbps + min_data_rate_mbps+ average_data_rate_mbps + std_data_rate_mbps + med_data_rate_mbps + average_altitude_m + dayofweek

Variables in model	**0**	**1**	**MeanDecreaseAccuracy**	**MeanDecreaseGini**
Channel_centre_freq_ghz	51.586281	53.417186	61.56373	5052.9176
Channel_bw_mhz	0	0	0	0
channel_occupancy_data	32.482188	27.662882	57.14343	10449.3726
channel_occupancy_total	32.028093	26.797372	46.21612	17472.2019
total_packets	47.009393	22.197395	46.26861	19172.1817
average_latitude_dd	33.069896	36.023284	44.30089	7161.0729
average_longitude_dd	29.553898	29.027712	32.75051	7666.7024
total_data_packets_80211a	0	0	0	0
total_data_packets_80211b	44.938419	43.644528	64.47187	10714.064
total_data_packets_80211g	12.442681	15.642695	21.24841	1346.4138
total_data_packets_80211n	14.351611	23.700295	29.25236	856.5648
total_data_packets_80211ac	0	0	0	0
max_data_rate_mbps	9.642712	17.687075	19.39531	1279.7473
min_data_rate_mbps	14.107398	14.292719	21.84789	1138.6607
average_data_rate_mbps	10.028096	18.657127	22.3797	1605.7349
std_data_rate_mbps	15.415915	19.401214	26.027	4310.1068
med_data_rate_mbps	10.105444	16.025346	22.27574	962.8847
average_alitutude_m	35.612684	32.365065	46.66307	8613.5984
dayofweek	14.35983	9.625012	18.45179	4360.9417

Table 5 Importance table for model 5.

Type of random forest: classification
Number of trees: 100
No. of variables tried at each split: 4

OOB estimate of error rate: 18.08%
Confusion matrix:

	0	1	class.error
0	71755	27722	0.2786775
1	23083	158430	0.1271700

Confusion matrix after prediction using the test data.

	Predicted		
Observed	0	1	class error
0	30397	11951	0.2822
1	9718	67565	0.1257

Normally in a classification type situation a training and test set are required to get an unbiased estimate of the test set error. However, with the random forest algorithm there is no need for cross validation. It is done internally during the run. Each tree is constructed using a different bootstrapped sample. One third of the cases are left out of the bootstrapped sample when constructing the ith tree. The cases left out in the construction of the ith tree are used for the classification of the kth tree. Through this test set classification is obtained for about one third of the trees. At the end of the run, let j be the class that got most of the votes every time case n was OOB. The proportion of times j is not equal to the true class of j average over all cases is the OOB error estimate. This is unbiased for many tests. For more details refer to Leo Breiman and Adele Cutler in [25]. As a result a test set is not required. This was only learned after the models here were run on a training set then the result compared to a test set. In the fifth model above we see the result of the confusion matrix give from the random forest algorithm in R and the confusion matrix constructed using the test data actually give the same error. The confusion matrix or error matrix is a special type of contingency table with two dimensions, observed and predicted values with the same set of classes for each observed and predicted. It provides a visualization of the performance of an algorithm, usually a supervisor learning algorithm. [29] Above, in the confusion matrix created after the prediction using the set aside test data, the calculated class errors are the same as those provided in the confusion matrix given with the output of the random forest function in R.

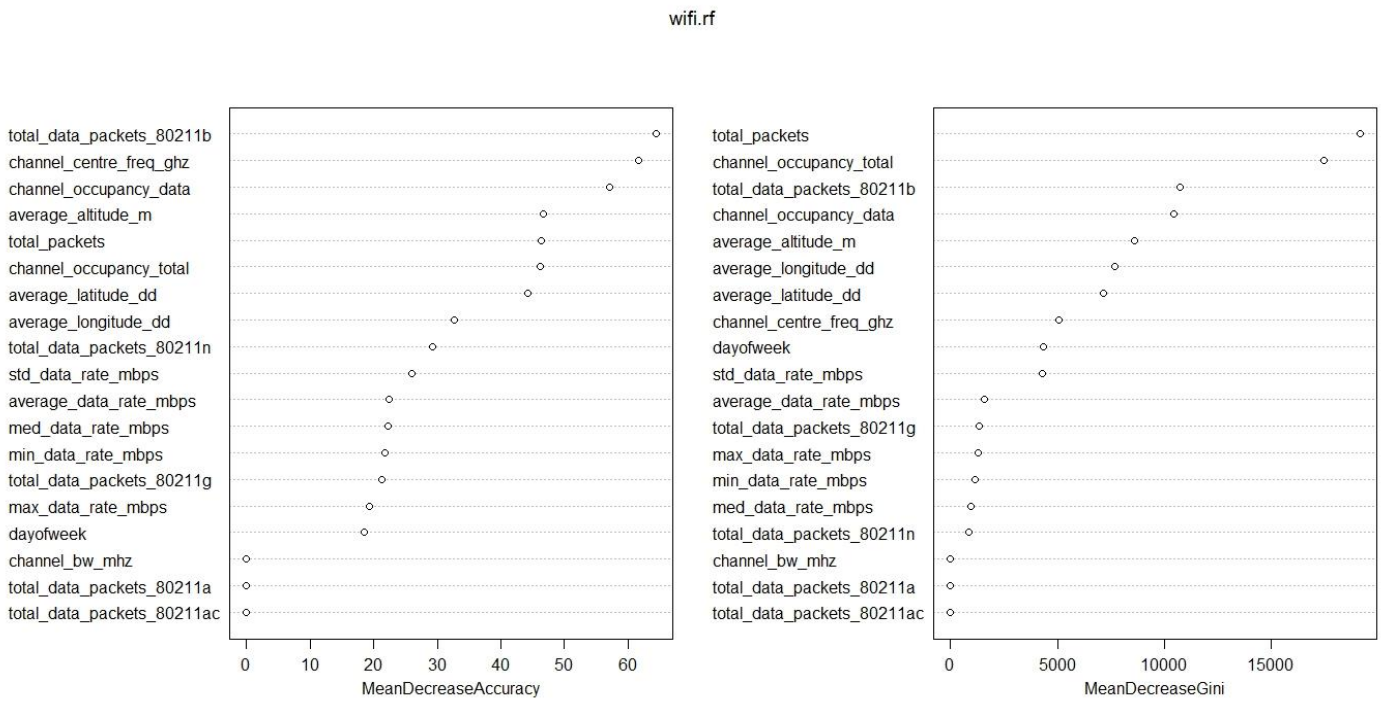

Figure 1 Variable importance plots for model 5.

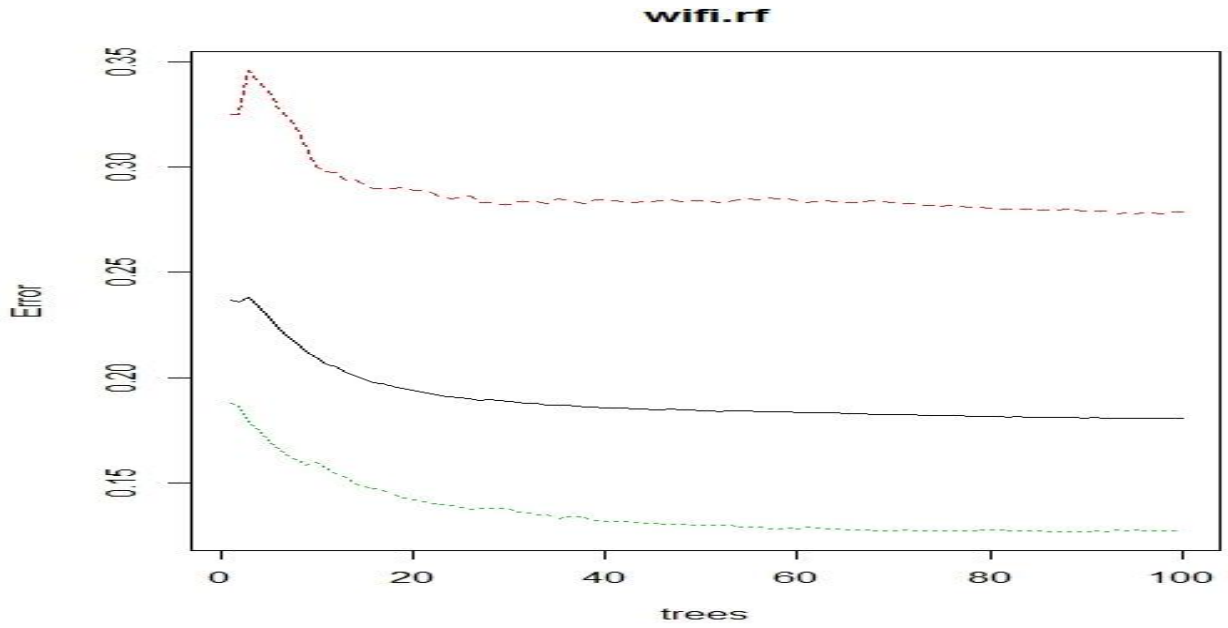

Figure 2 Graph for model 5

The Gini coefficient, also called Gini ratio or Gini index, is a measure of the dispersion that represents income distribution. It is commonly used as a measure of inequality. It measures the inequality among values of a frequency distribution. A Gini coefficient value of 0 represents perfect equality, that is, all values are the same.

For example, everyone has the same income. A Gini coefficient of 1 represents a maximum inequality among values. For example, one person makes all the income and others none. Values greater than 1 are possible. This occurs when some persons represent a negative contribution to the total, e.g. having a negative wealth or income. [29] With the randomForest function in R, the variable importance plot tells us how important each variable is in classifying the data. The variables are listed along the vertical axis and their importance on the horizontal axis as show in the right hand plot in figure 5, MeanDecreaseGini plot. The variables are ordered from most to least important from top to bottom. This helps decide how many important variables to choose. The mean decrease in accuracy represents how much a variable affects the accuracy of the random forest. This is determined during the calculation of the out of bag error, OOB. If the accuracy of a random forest due to excluding a particular variable decreases, then the more important that variable is. This means, variables with a larger mean decrease accuracy are more important for classifying the data. [26, 27] The importance table gives a list of the importance of each independent variable. It gives a summary of the mean decrease Gini and mean decrease accuracy. Table 5 is the importance table for model 5.

Considering both table 5 and figure 1 we see that both indicate the same level of importance for each variable given. The least important variables are quite clear as they are the same in both the MeanDecreaseGini and MeanDecreaseAccuracy plots in figure 5. The other variables can be grouped in 3 groups of "importance", most important, important, and less important. Considering the MeanDecraseGini plot, total_packet, channel_occupancy_total are in the most important group of variables; total_data_pacekts_80211b, channel_occupancy_data, average_altitude_m, average_longitude_dd, average_latitude_dd, channel_centre_freq_ghz, dayofweek, std_data_rate_mbps, are important variables; average_data_rate_mpbs, total_data_packets_80211g, max_data_rate_mbps, min_data_rate_mbps, med_data_rate_mbps, total_data_packets_80211n are less important. A similar grouping of the variable can be done for the MeanDecreaseAccuracy.

Figure 2 is a graph of the plot of the margin function in R applied to the random forest created. From this plot we get an idea of how many trees may be needed in the forest to get the result. From figure 2 is looks as though between 40-60 trees are required for the forest to get the result. In model 6 below we consider less trees to see what difference there is.

The **sixth model** considered is given below. It is similar to model 5 except less trees were considered, only 31, and three new variables were included, total_clients, average_rssi_dbm and num_ap. The OOB estimate error has decreased a bit; the three least important variable in model 5 are also the least important in model 6; the MeanDecreaseGini values can be divided up into four groups this time however, there seems to be a more gradual change in importance of the variables based on the MeanDecraseAccuracy values. Total_clients is the most important variable in both plots.

nonbeacon_cat ~ channel_centre_freq_ghz + channel_bw_mhz + channel_occupancy_data + channel_occupancy_total + total_packets + average_latitude_dd + average_longitude_dd + total_data_packets_80211a+total_data_packets_80211b+total_data_packets_80211g+ total_data_packets_80211n + total_data_packets_80211ac + max_data_rate_mbps + min_data_rate_mbps + average_data_rate_mbps + std_data_rate_mbps + med_data_rate_mbps + average_altitude_m + dayofweek + total_clients + average_rssi_dbm + num_ap

Variables in Model	**0**	**1**	**MeanDecreaseAccuracy**	**MeanDecreaseGini**
Channel_centre_freq_ghz	35.77769	24.15517	35.400686	3860.7797

Channel_bw_mhz	0	0	0	0
channel_occupancy_data	21.98206	17.24078	28.825906	8164.2132
channel_occupancy_total	11.71016	18.055	22.51272	15999.6042
total_packets	19.1335	12.76524	18.75332	13430.0605
average_latitude_dd	12.8595	14.04638	18.874887	5622.1248
average_longitude_dd	14.88227	11.79022	14.266941	6244.6882
total_data_packets_80211a	0	0	0	0
total_data_packets_80211b	20.53272	28.1195	32.851518	8748.4815
total_data_packets_80211g	8.052041	7.795184	9.944005	1619.0016
total_data_packets_80211n	8.327027	11.28272	12.74407	833.1405
total_data_packets_80211ac	0	0	0	0
max_data_rate_mbps	5.618827	9.873518	10.61564	1048.4685
min_data_rate_mbps	8.6822	9.814764	13.155044	1060.4746
average_data_rate_mbps	8.700609	8.678168	10.82752	1601.338
std_data_rate_mbps	9.040094	8.051995	10.899074	3454.0421
med_data_rate_mbps	8.931583	8.454409	12.145565	850.4232
average_alitutude_m	21.43942	10.54313	16.516371	6836.5061
dayofweek	11.70202	3.717438	8.596315	3860.761
total_clients	32.33715	36.21425	47.647833	19675.1406
average_rssi_dbm	22.9571	22.35886	32.97485	6484.5168
num_ap	21.14173	15.93521	23.739467	5889.2814

Table 6 Importance table for model 6.

Number of trees: 31
No. of variables tried at each split: 4
OOB estimate of error rate: 16.83%
Confusion matrix:

	0	1	class.error
0	74030	25447	0.2558079
1	21847	159666	0.1203605

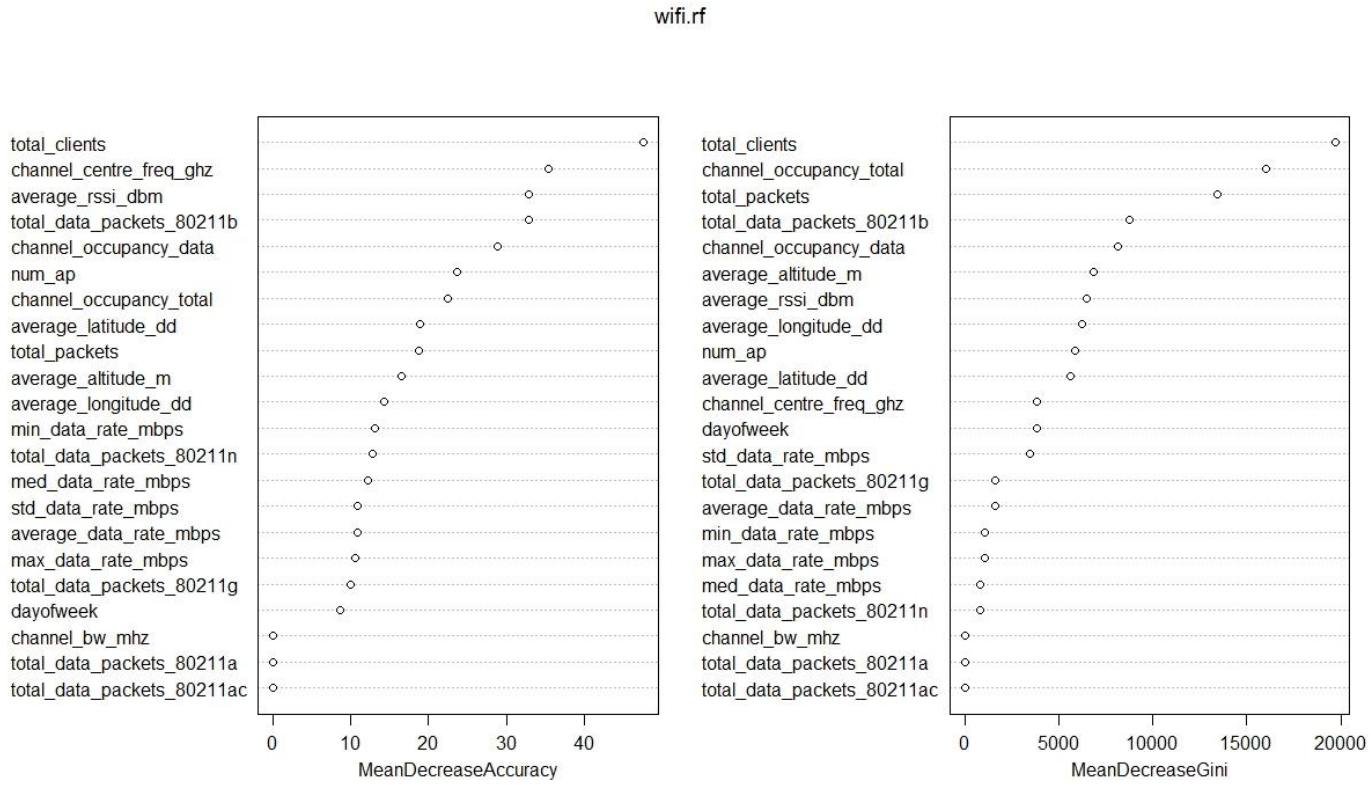

Figure 3 Variable importance plots for model 6

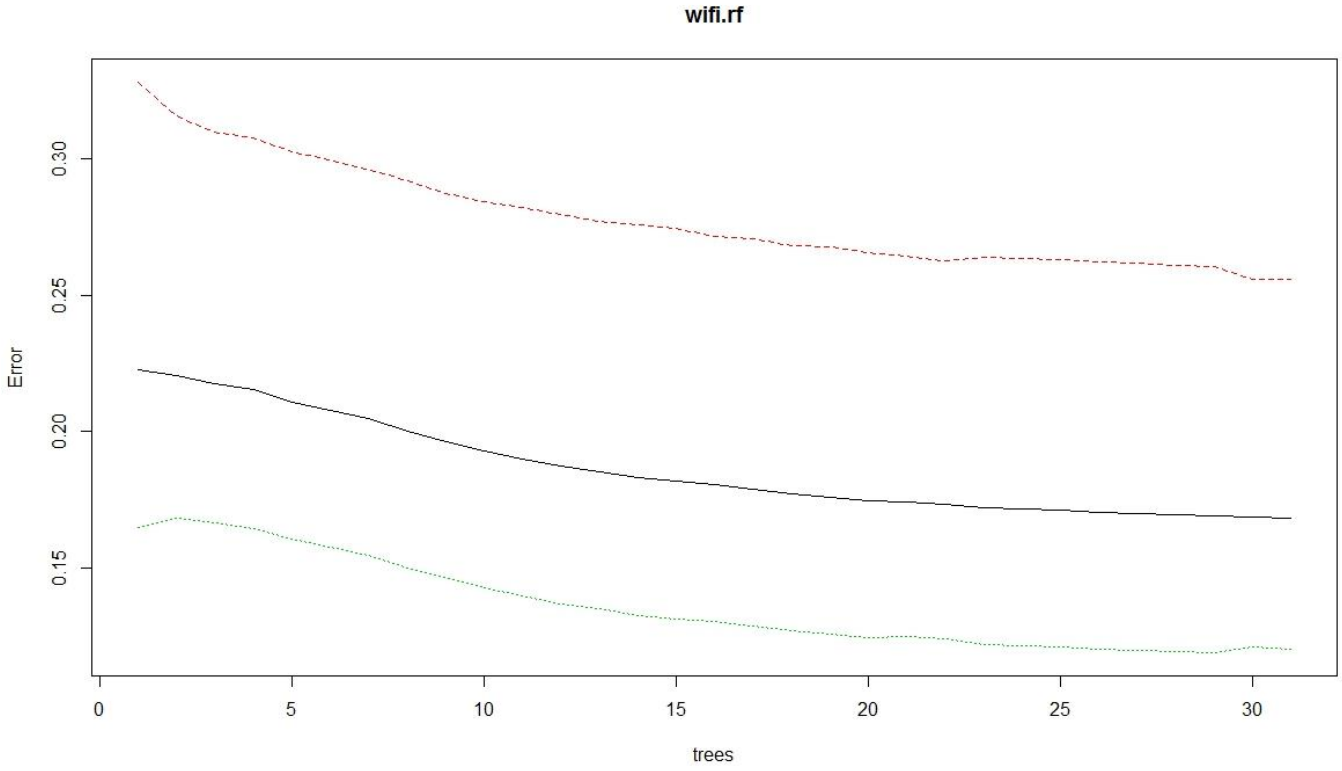

Figure 4 Graph for model 7

The **seventh model** considered is given below. It is exactly the same as model 6 except 100 trees are considered. The OOB estimate error has gone down a bit. From here on in all the models consist of 100 trees. The observations are the same as for model 6. Total_clients is more clearly the most important variable according to the two plots in figure 7.

nonbeacon_cat ~ channel_centre_freq_ghz + channel_bw_mhz + channel_occupancy_data + channel_occupancy_total+ total_packets + average_latitude_dd + average_longitude_dd + total_data_packets_80211a + total_data_packets_80211b
+ total_data_packets_80211g + total_data_packets_80211n + total_data_packets_80211ac + max_data_rate_mbps + min_data_rate_mbps + average_data_rate_mbps + std_data_rate_mbps + med_data_rate_mbps + average_altitude_m + dayofweek + total_clients + average_rssi_dbm + num_ap

Variable	**0**	**1**	**MeanDecreaseAccuracy**	**MeanDecreaseGini**
Channel_centre_freq_ghz	43.84827	44.79464	61.96988	3902.7652
Channel_bw_mhz	0	0	0	0
channel_occupancy_data	35.05334	35.39494	51.52306	8632.7022
channel_occupancy_total	21.15951	25.20082	37.93654	13266.8227
total_packets	32.86832	23.64913	37.33153	15090.0415
average_latitude_dd	27.57029	31.52375	39.14302	5673.4727
average_longitude_dd	23.55039	22.3508	25.34251	6214.6445
total_data_packets_80211a	0	0	0	0
total_data_packets_80211b	40.73673	48.62987	56.71802	9792.6661
total_data_packets_80211g	13.12346	12.1936	20.61534	1187.9951
total_data_packets_80211n	16.37262	21.53019	28.97596	843.5343
total_data_packets_80211ac	0	0	0	0
max_data_rate_mbps	14.02575	15.86346	20.63202	1220.3784
min_data_rate_mbps	15.41629	15.54309	23.62571	943.2085
average_data_rate_mbps	33.91812	23.94595	34.78432	7068.4452
std_data_rate_mbps	18.733	17.90044	26.73446	3516.5337
med_data_rate_mbps	14.22227	18.34078	23.89957	886.5107
average_alitutude_m	33.91812	23.94595	34.78432	7068.4452
dayofweek	14.59079	10.29268	18.45623	3966.9087
total_clients	84.80059	70.70511	97.12626	20071.0442
average_rssi_dbm	34.76384	37.06015	48.85297	6522.9095
num_ap	31.09069	30.58756	47.40033	5865.4441

Table 7 Importance table for model 7.

Number of trees: 100
No. of variables tried at each split: 4

OOB estimate of error rate: 15.86%
Confusion matrix:

	0	1	class.error
0	74720	24757	0.2488716
1	19807	161706	0.1091217

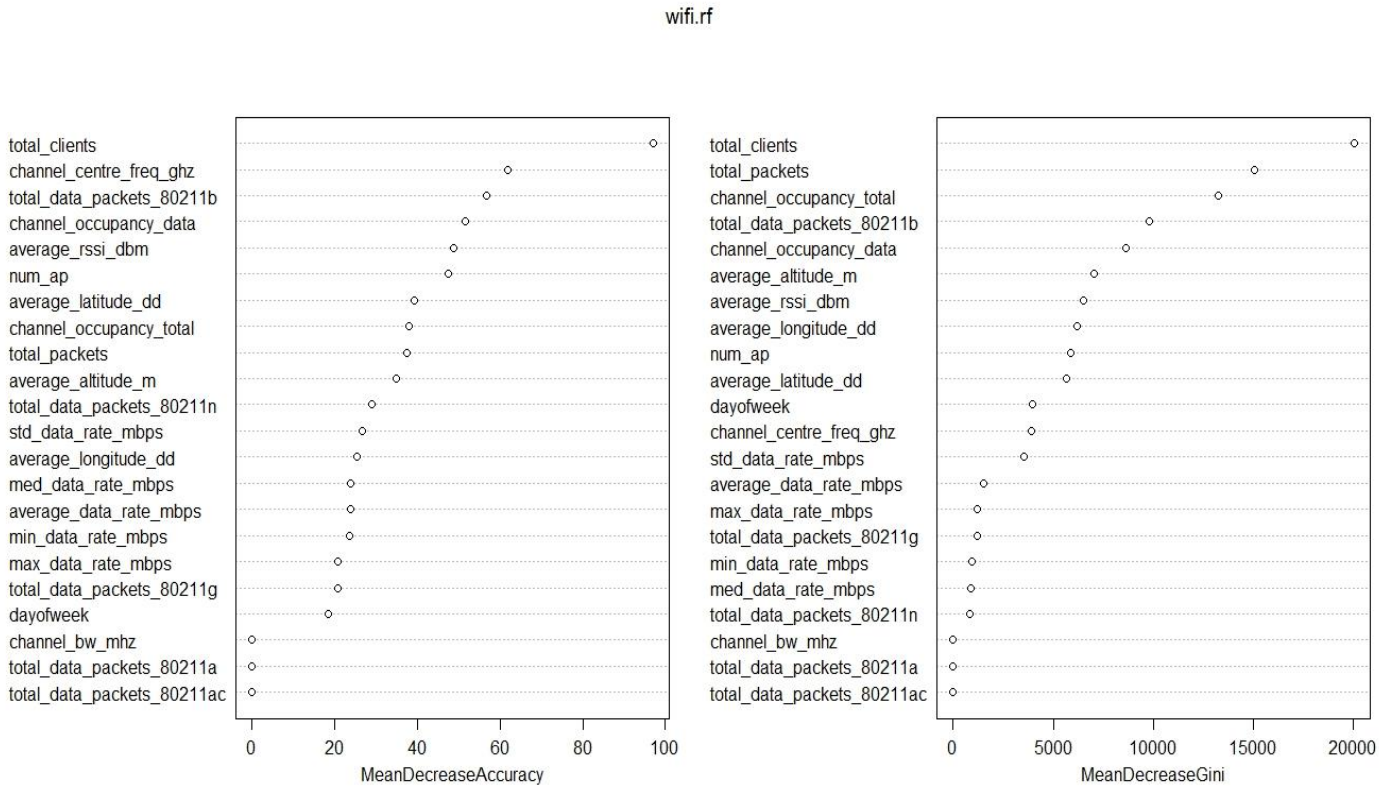

Figure 5 Variable importance plots for model 7

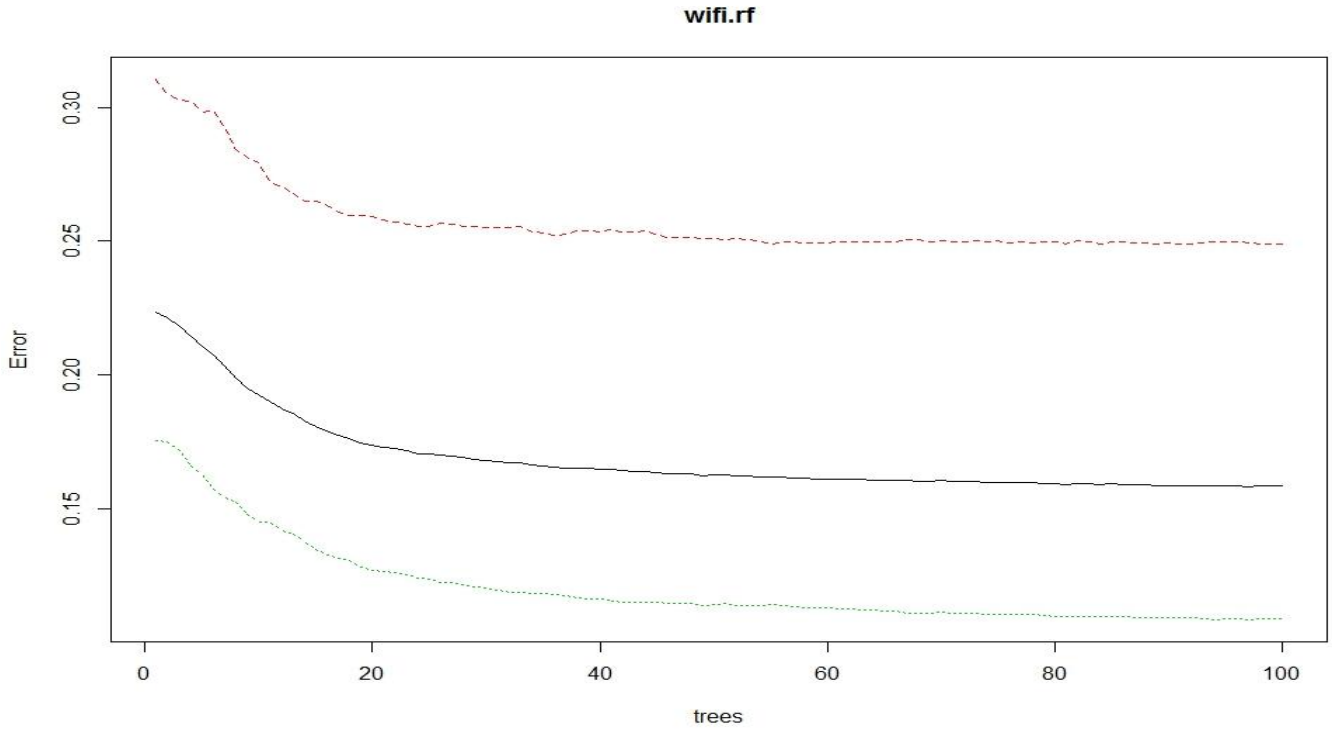

Figure 6 Graph for model 8

Model 8

nonbeacon_cat ~ channel_centre_freq_ghz + channel_occupancy_data + channel_occupancy_total+ total_packets + average_latitude_dd + average_longitude_dd + total_data_packets_80211b + total_data_packets_80211g + total_data_packets_80211n + max_data_rate_mbps + min_data_rate_mbps+ average_data_rate_mbps + std_data_rate_mbps + med_data_rate_mbps + average_altitude_m + dayofweek + total_clients+ average_rssi_dbm + num_ap

Variables in model	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
Channel_centre_freq_ghz	59.71981	49.73907	70.58816	4313.9658
channel_occupancy_data	47.07576	38.39433	60.80514	9122.5688
channel_occupancy_total	22.03892	29.78227	38.27361	14759.1786
total_packets	31.91898	27.42407	44.54605	13816.5679
average_latitude_dd	39.39056	39.86367	53.56458	6271.0383
average_longitude_dd	27.21004	24.49929	28.73408	6563.244
total_data_packets_80211b	51.93726	56.54712	69.64588	9810.1539
total_data_packets_80211g	12.39992	16.89353	21.53979	1235.2061
total_data_packets_80211n	18.81734	20.57899	29.09497	846.0178
max_data_rate_mbps	12.10718	16.36015	17.40937	1196.3836
min_data_rate_mbps	12.42984	15.59571	20.96867	908.2359
average_data_rate_mbps	10.92901	25.25462	22.78405	1440.4299
std_data_rate_mbps	19.40745	18.17449	30.61553	3815.2782
med_data_rate_mbps	12.54846	15.41448	22.03154	883.8675
average_alitutude_m	40.40781	33.77546	47.02678	7996.8434
dayofweek	16.41697	10.73799	19.37819	4461.4383
total_clients	99.96332	72.35925	101.94178	21995.1189
average_rssi_dbm	45.90964	42.12292	56.24213	7236.2027
num_ap	34.17953	30.18806	42.31992	6645.391

Table 8 Importance table for model 8.

Number of trees: 100
No. of variables tried at each split: 4
OOB estimate of error rate: 15.76%
Confusion matrix:

	0	1	class.error
0	74757	24720	0.2484997
1	19578	161935	0.1078600

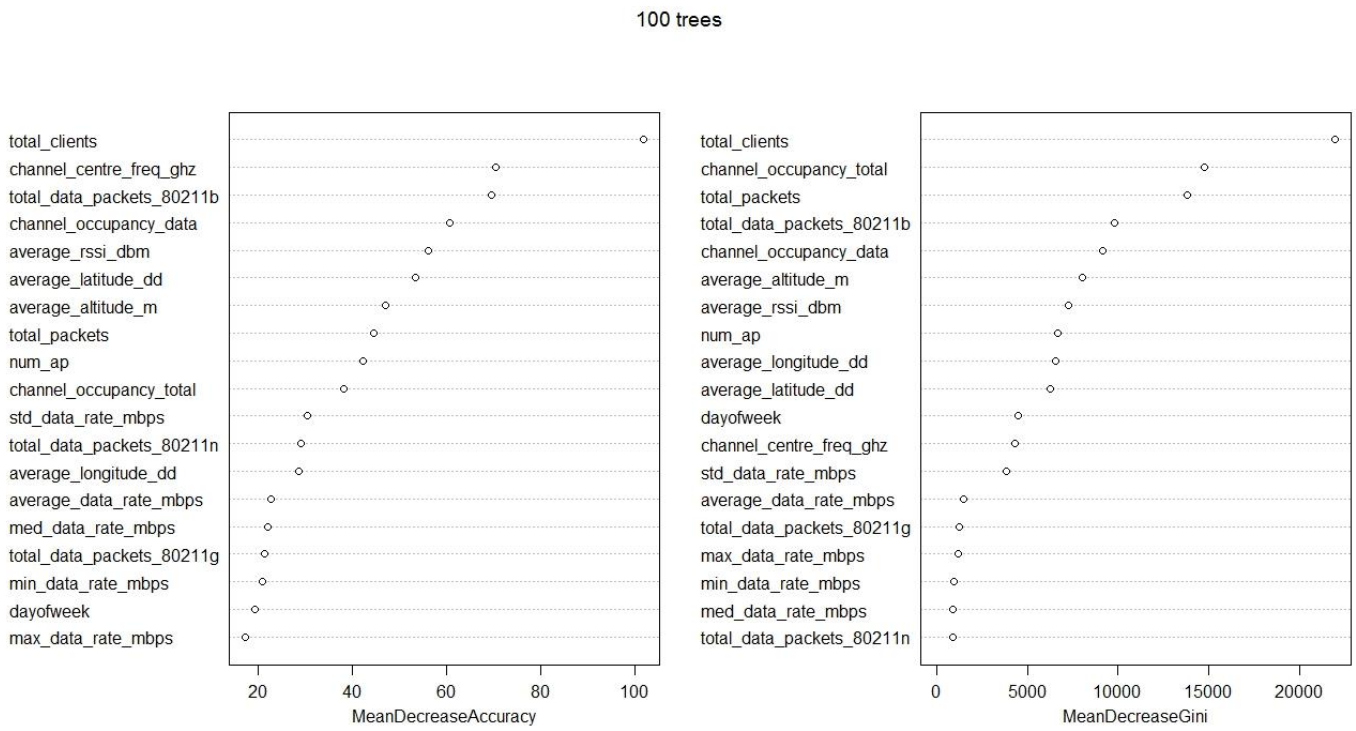

Figure 7 Variable importance plots for model 8.

Another categorical variable was created, timeofday. The time was given based on a 24 hour clock. The hour portion of the start_datetime was considered. The timeofday was "day" is the hour was between 6 and 18; timeofday was assigned to "na" if the hourwas missing or null; timeofday was assigned to 'night' for all other hours, that is 19-5 (19-23 and 00-5).

Model 9

nonbeacon_cat ~ channel_centre_freq_ghz + channel_occupancy_data + channel_occupancy_total + total_packets + average_latitude_dd + average_longitude_dd + total_data_packets_80211b+ total_data_packets_80211g + total_data_packets_80211n + max_data_rate_mbps + min_data_rate_mbps + average_data_rate_mbps + std_data_rate_mbps + med_data_rate_mbps + average_altitude_m + dayofweek + total_clients + average_rssi_dbm + num_ap + timeofday

Variables in model	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
Channel_centre_freq_ghz	58.53399	47.588122	66.22602	4329.7089
channel_occupancy_data	43.07678	33.171643	52.18418	8978.2464
channel_occupancy_total	24.59232	33.37567	43.19647	13613.8464
total_packets	34.18206	27.873227	41.42909	16123.0109
average_latitude_dd	35.3991	26.734624	38.99984	6336.4279
average_longitude_dd	25.86093	19.929826	23.6401	6812.8247
total_data_packets_80211b	44.644	53.906388	64.6182	10177.6625
total_data_packets_80211g	14.55413	10.909295	17.89713	1094.1407
total_data_packets_80211n	17.39995	21.471204	29.98816	808.1248

max_data_rate_mbps	13.07616	14.266464	18.13356	1283.1813
min_data_rate_mbps	11.68621	14.92852	20.24505	988.0047
average_data_rate_mbps	11.05983	22.392592	21.21605	1447.3797
std_data_rate_mbps	18.15146	16.538018	22.89601	3731.031
med_data_rate_mbps	11.43097	20.357825	23.65397	896.9675
average_alitutude_m	48.10815	26.212606	43.71215	7874.6915
dayofweek	15.06859	12.925203	19.08649	4482.0771
total_clients	118.74788	80.107522	113.45046	19918.1659
average_rssi_dbm	46.40008	39.709564	54.76251	7323.0249
num_ap	37.887	34.685035	54.40401	6584.6236
timeofday	10.78064	7.817544	12.64822	803.3632

Number of trees: 100
No. of variables tried at each split: 4
OOB estimate of error rate: 15.85%
Confusion matrix:
     0   1   class.error
0 74713 24593   0.2476487
1 19827 161116 0.1095759

Table 9 Importance table for model 9.

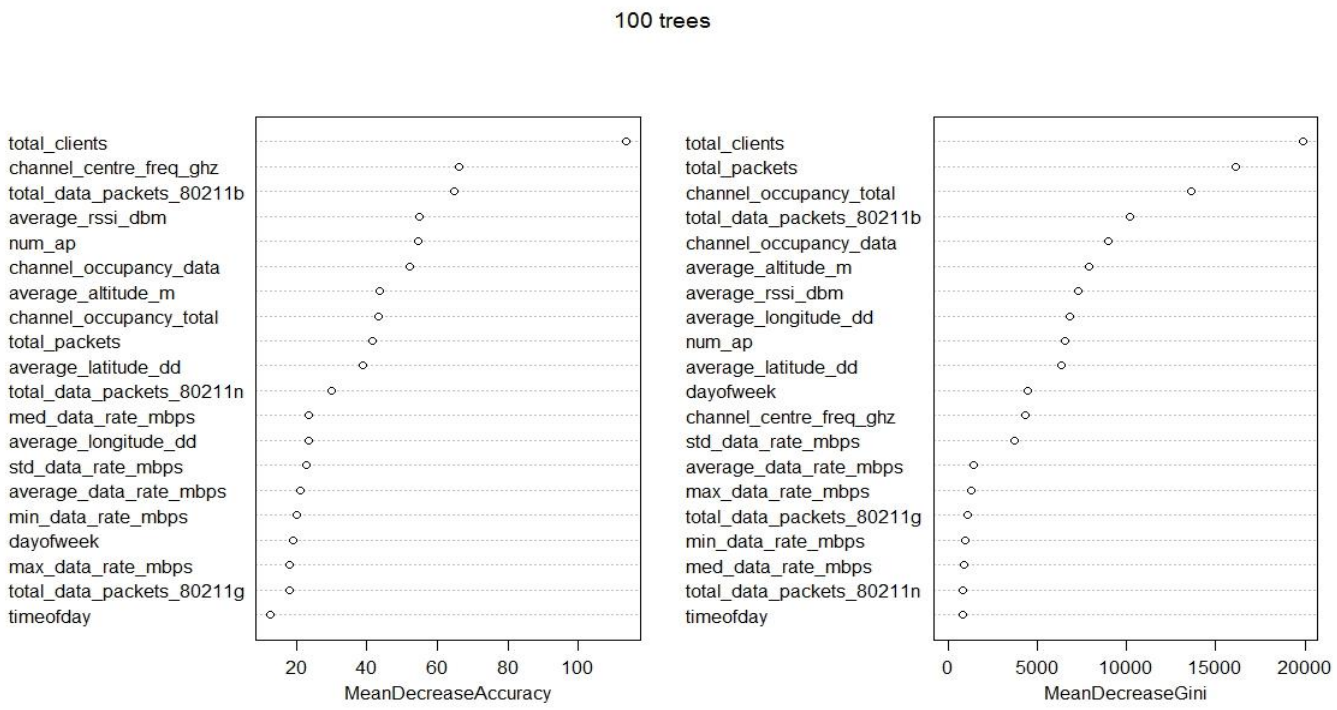

Figure 8 Variable importance plots for model 9.

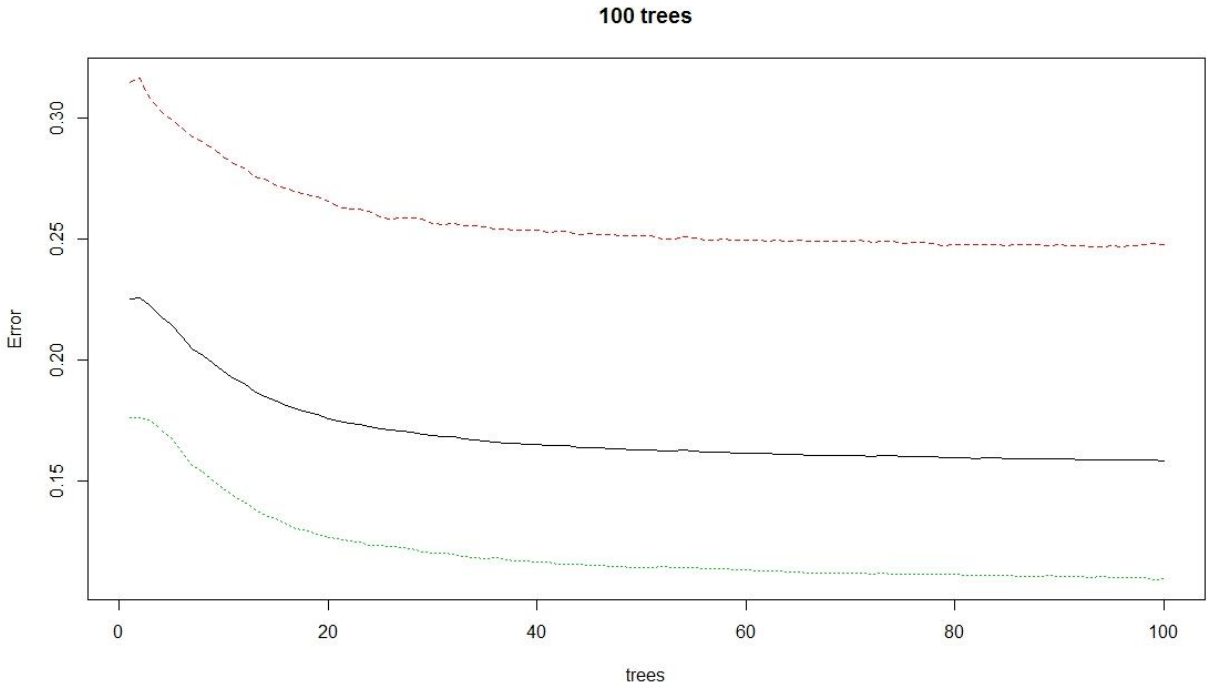

Figure 9 Graph for model 11

Model 10

nonbeacon_cat ~ num_ap + total_clients + channel_centre_freq_ghz + channel_bw_mhz + channel_occupancy_data + channel_occupancy_total + total_packets + total_data_packets_80211a + total_data_packets_80211b + total_data_packets_80211g + total_data_packets_80211n +

total_data_packets_80211ac + max_data_rate_mbps + min_data_rate_mbps + average_data_rate_mbps +std_data_rate_mbps + med_data_rate_mbps + average_latitude_dd + average_longitude_dd + average_altitude_m + average_rssi_dbm + dayofweek + timeofday

Variables in model	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
Channel_centre_freq_ghz	51.88993	39.000466	54.82795	3886.8769
channel_centre_freq_ghz	0	0	0	0
channel_occupancy_data	38.8668	36.386288	50.38592	8366.0215
channel_occupancy_total	21.77503	23.3286	35.78535	13725.9064
total_packets	29.97303	26.987459	37.45197	13522.1364
average_latitude_dd	32.38691	27.621482	35.88125	5794.2685
average_longitude_dd	21.34561	24.644008	27.3515	6101.6174
total_data_packets_80211a	0	0	0	0
total_data_packets_80211b	39.34134	43.674846	51.89463	10095.3365
total_data_packets_80211g	12.68334	11.435117	17.88254	1483.6865
total_data_packets_80211n	18.70826	20.657663	27.44742	906.473
total_data_packets_80211ac	0	0	0	0
max_data_rate_mbps	10.42875	15.277508	16.78479	1205.1123
min_data_rate_mbps	12.69455	16.019898	18.5239	1031.9158
average_data_rate_mbps	13.93342	18.063683	19.98327	1552.7406
std_data_rate_mbps	16.09389	18.948084	26.86825	3751.1705
med_data_rate_mbps	11.69532	17.077359	21.76209	879.9155
average_alitutude_m	34.11871	22.781441	33.9491	6983.0391
dayofweek	15.38931	10.356602	20.31988	3963.9166
total_clients	61.52822	56.445273	72.28925	19976.9646
average_rssi_dbm	34.1679	30.070561	38.95748	6568.7768
num_ap	25.14479	21.034883	29.44052	5878.2927
timeofday	12.33077	5.847918	13.60785	735.2871

Table 10 Importance table for model 10.

Number of trees: 100
No. of variables tried at each split: 4
OOB estimate of error rate: 15.93%
Confusion matrix:

	0	1	class.error
0	74802	24744	0.2485685
1	19987	161325	0.1102354

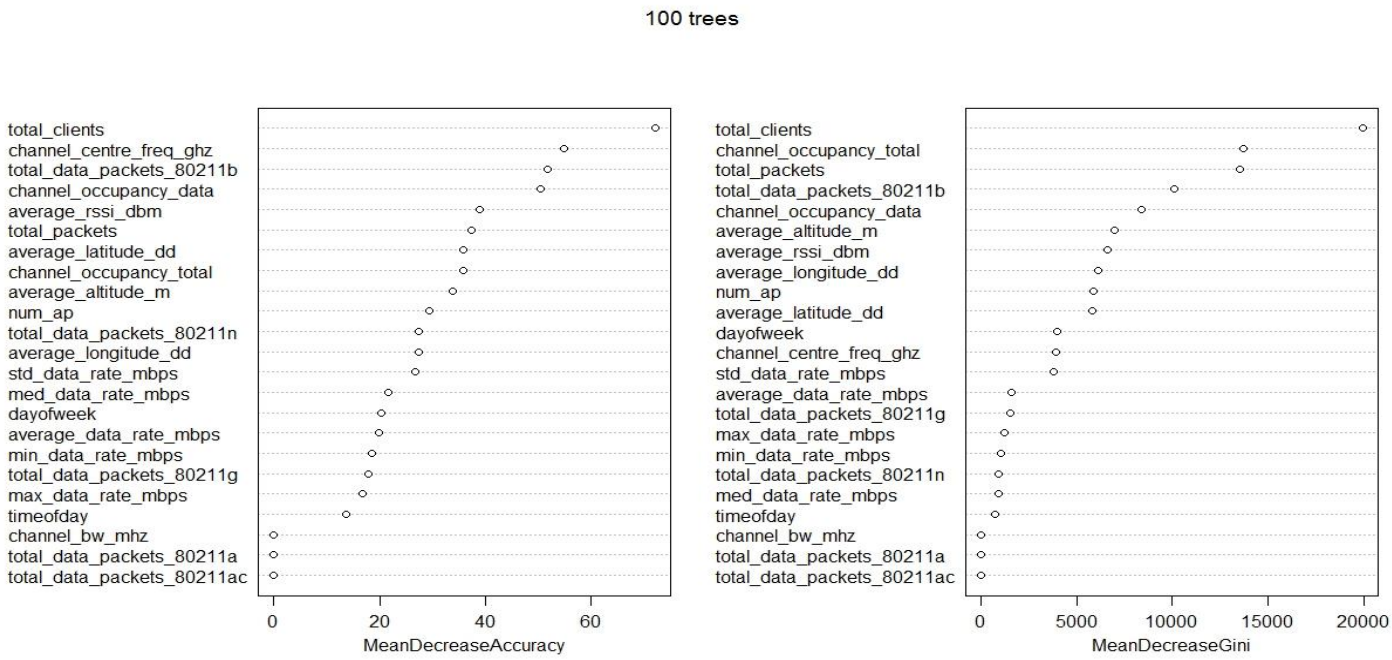

Figure 10 Variable importance plots for model 10.

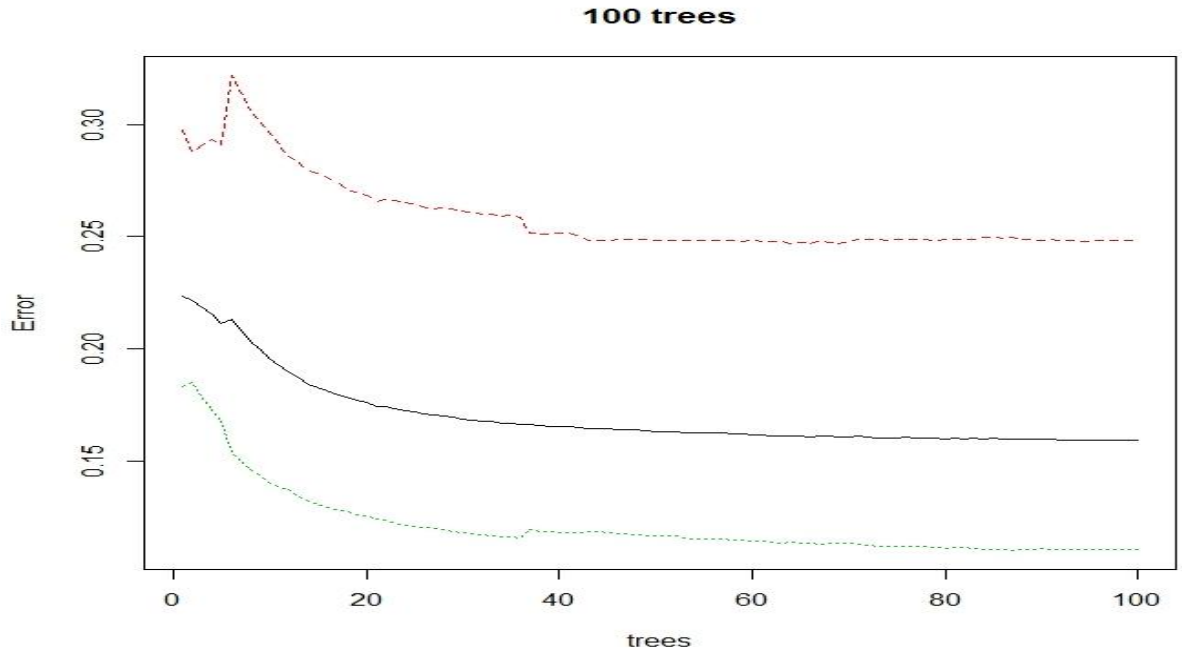

Figure 11 Graph for model 12

Model 11

nonbeacon_cat ~ num_ap + total_clients + channel_centre_freq_ghz + channel_occupancy_data + channel_occupancy_total + total_packets + total_data_packets_80211b + total_data_packets_80211g + total_data_packets_80211n + max_data_rate_mbps + min_data_rate_mbps + average_data_rate_mbps + std_data_rate_mbps + med_data_rate_mbps + average_latitude_dd + average_longitude_dd + average_altitude_m + average_rssi_dbm + dayofweek + timeofday

Variables in model	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
Channel_centre_freq_ghz	52.72896	48.205757	67.71378	4318.6446
channel_occupancy_data	46.72444	32.9158	57.34934	9500.5864
channel_occupancy_total	28.25955	25.969678	46.11849	13600.2864
total_packets	31.62689	31.697082	41.23941	16000.2716
average_latitude_dd	32.92167	29.884706	38.78434	6358.052
average_longitude_dd	27.27365	24.154578	30.32721	6582.452
total_data_packets_80211b	42.36641	54.553506	59.75445	9797.4489
total_data_packets_80211g	13.14355	18.024363	20.12768	1360.5564
total_data_packets_80211n	18.02432	21.332777	29.95782	865.0593
max_data_rate_mbps	14.44651	15.567522	21.23904	1220.4032
min_data_rate_mbps	12.48974	12.505792	20.70615	947.4804
average_data_rate_mbps	12.23072	23.077308	21.47498	1538.026
std_data_rate_mbps	17.90193	18.126257	27.69577	3259.481
med_data_rate_mbps	12.91291	18.065906	22.75742	932.6045
average_alitutude_m	39.91126	29.95528	44.79902	7909.8035
dayofweek	18.21362	12.401767	21.02732	4476.1011
total_clients	98.48723	87.946722	110.42174	20678.5129
average_rssi_dbm	43.28822	38.183097	49.10739	7259.7284
num_ap	26.97683	33.83751	42.03292	6590.2653
timeofday	11.54598	5.614515	11.53564	796.7764

Table 11 Importance table for model 11.

Number of trees: 100
No. of variables tried at each split: 4
OOB estimate of error rate: 15.8%
Confusion matrix:

	0	1	class.error
0	74906	24640	0.2475238
1	19744	161568	0.1088952

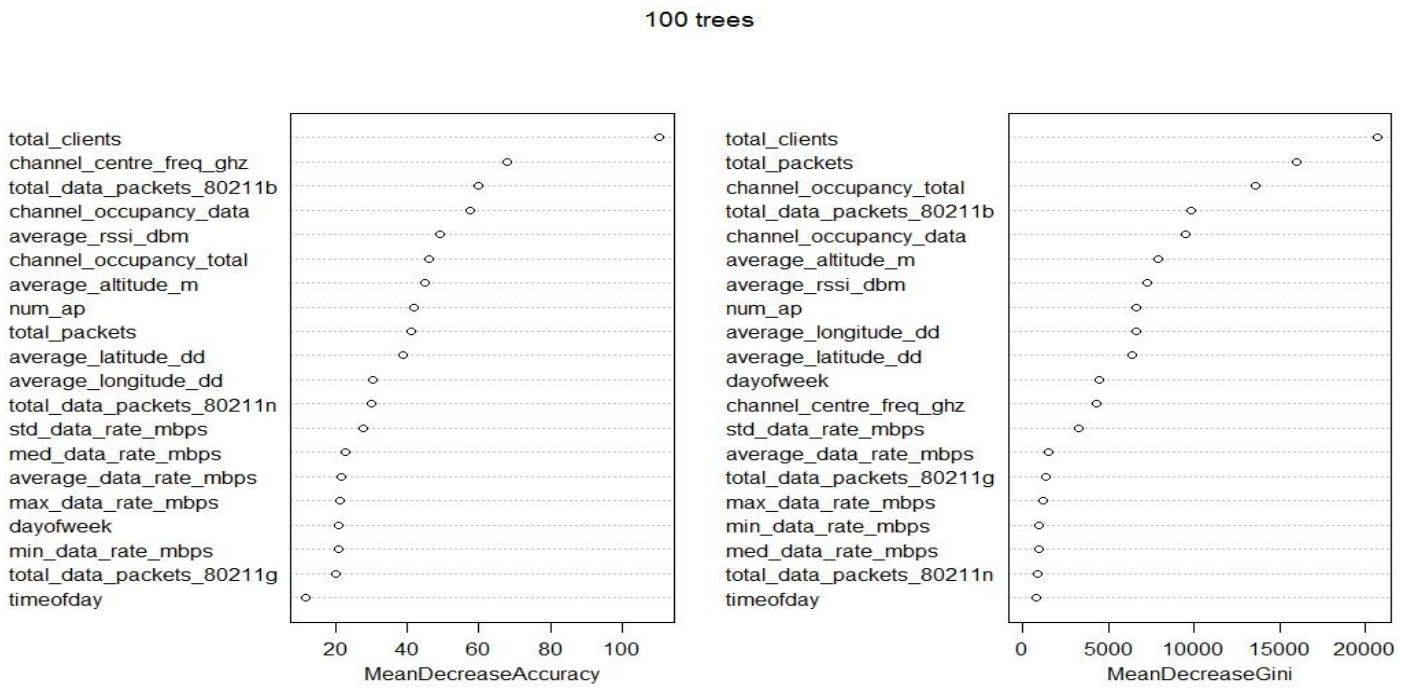

Figure 12 Variable importance plots for model 11.

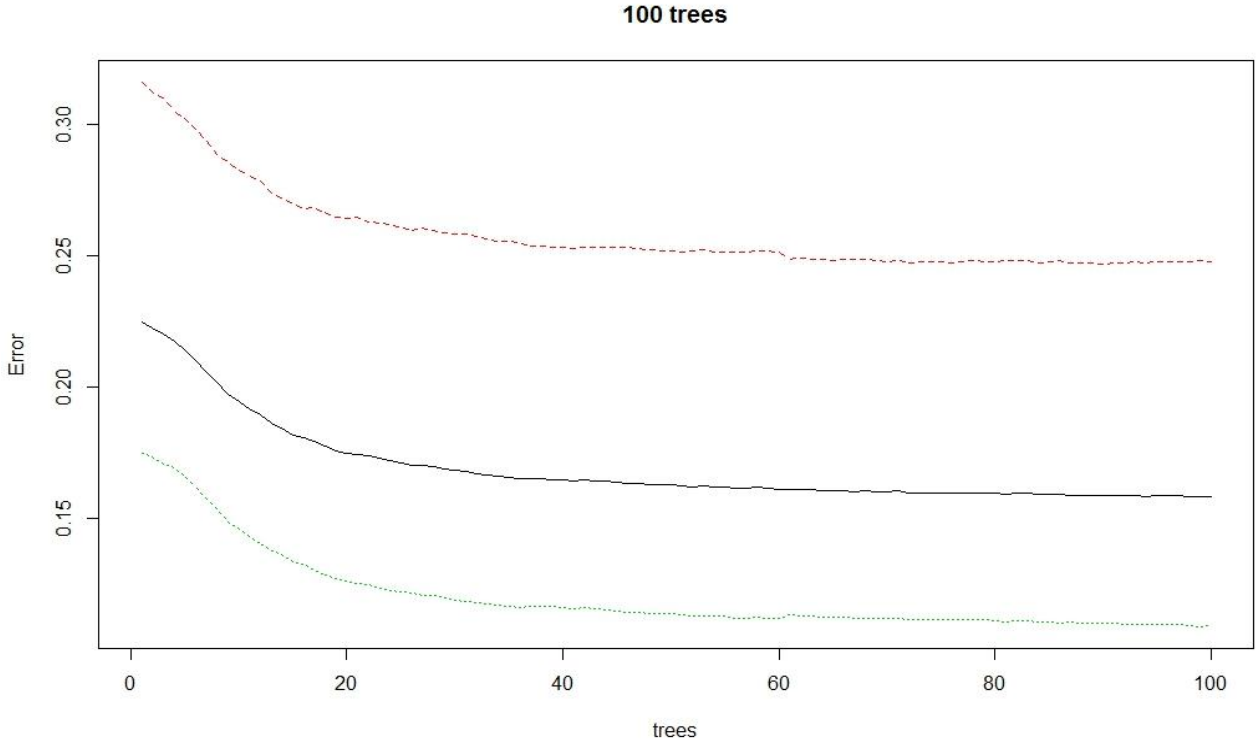

Figure 13 Graph of model 14

Model 12

nonbeacon_cat ~ num_ap + total_clients + channel_centre_freq_ghz + channel_occupancy_data + channel_occupancy_total + total_packets + total_data_packets_80211b + total_data_packets_80211g + total_data_packets_80211n + max_data_rate_mbps + min_data_rate_mbps + average_data_rate_mbps + std_data_rate_mbps + med_data_rate_mbps + average_latitude_dd + average_longitude_dd + average_altitude_m + average_rssi_dbm + dayofweek

Variables in model	0	1	MeanDecreaseAccuracy	MeanDecreaseGini
Channel_centre_freq_ghz	65.46722	46.85435	66.18677	4305.743
channel_occupancy_data	41.61188	35.95935	55.74873	8862.1983
channel_occupancy_total	24.59989	27.31717	42.09208	13104.052
total_packets	36.39978	29.00359	41.88497	15837.7542
average_latitude_dd	35.2867	30.31293	42.55536	6320.9864
average_longitude_dd	26.15033	21.43119	25.20022	6840.9599
total_data_packets_80211b	44.0838	59.05655	66.14875	9516.6439
total_data_packets_80211g	14.32225	13.22406	19.82363	1396.9683
total_data_packets_80211n	21.80771	25.74447	33.83753	795.086
max_data_rate_mbps	12.32714	14.4045	17.76332	1218.9599
min_data_rate_mbps	14.7124	13.021	21.0661	942.7906
average_data_rate_mbps	15.86649	20.42515	24.7729	1562.6461
std_data_rate_mbps	19.8821	17.98821	28.08728	3450.9806
med_data_rate_mbps	12.19841	19.41873	22.3384	897.5115
average_alitutude_m	37.54612	28.15848	42.52056	7946.9036
dayofweek	19.41011	12.04685	24.16124	4472.2692
total_clients	126.2108	79.13554	116.10761	21773.4391
average_rssi_dbm	42.7741	33.63376	44.92003	7323.312
num_ap	29.11218	36.46899	46.5858	6615.6243

Table 12 Importance table for model 12.

Number of trees: 100
No. of variables tried at each split: 4
OOB estimate of error rate: 15.88%
Confusion matrix:

	0	1	class.error
0	74441	24865	0.2503877
1	19650	161293	0.1085977

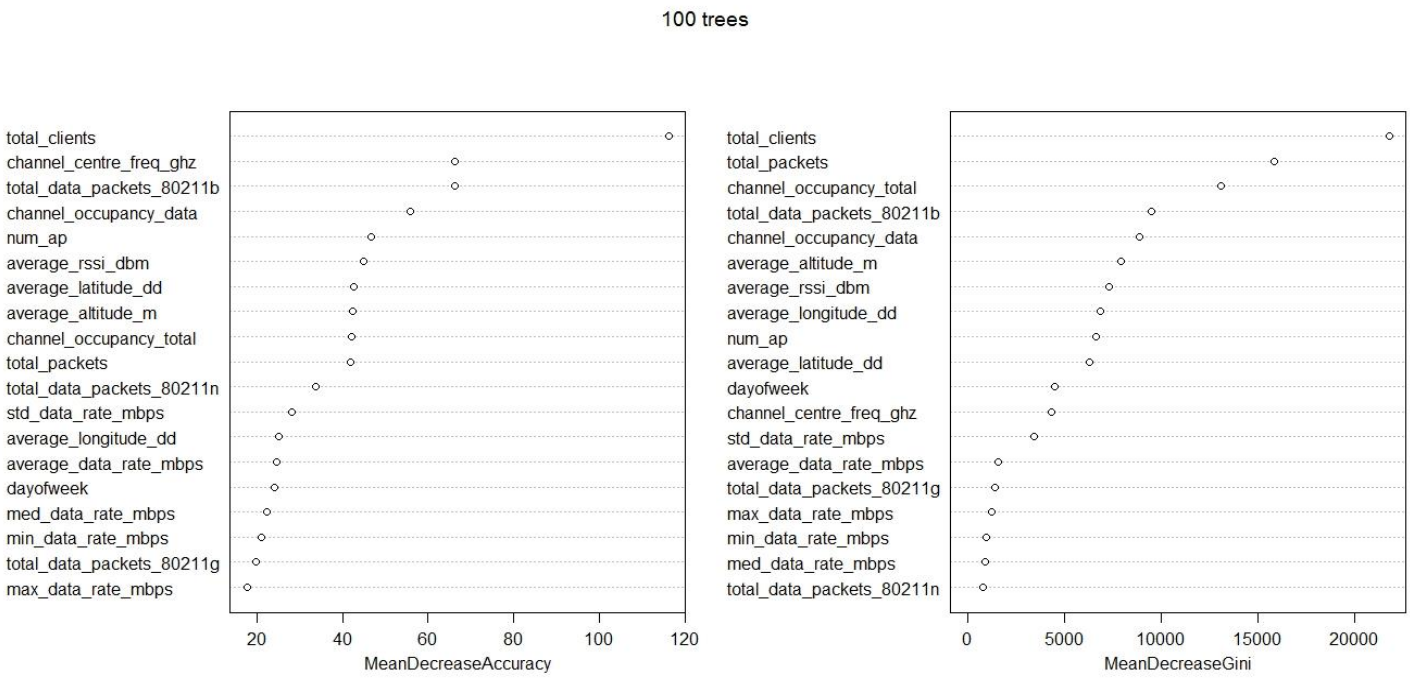

Figure 14 Variable importance plot for model 12.

model #	1	2	3	4	5	6	7	8	9	10	11	12
# trees	5	100	100	100	100	31	100	100	100	100	100	100
warnings	x	x	x									
OOB estimate (%)				27.77	18.08	16.83	15.86	15.76	15.83	15.93	15.8	15.88
Dependent variables												
nonbeacon_cat	x	x	x	x	x	x	x	x	x	x	x	x
Independent variables												
channel_centre_freq_ghz					x	x	x	x	x	x	x	x
channel_bw_mhz					x	x	x			x		
channel_occupancy_data					x	x	x	x	x	x	x	x
channel_occupancy_total					x	x	x	x	x	x	x	x
total_packets					x	x	x	x	x	x	x	x
average_latitude_dd	x	x	x	x	x	x	x	x	x	x		x
average_longitude_dd	x	x	x	x	x	x	x	x	x	x	x	x
total_data_packets_80211a					x	x	x			x	x	
total_data_packets_80211b					x	x	x	x	x	x	x	x
total_data_packets_80211g	x		x	x	x	x	x	x	x	x	x	x
total_data_packets_80211n	x	x		x	x	x	x	x	x	x	x	x
total_data_packets_80211ac					x	x	x			x		
max_data_rate_mbps					x	x	x	x	x	x	x	x
min_data_rate_mbps					x	x	x	x	x	x	x	x
average_data_rate_mbps					x	x	x	x	x	x	x	x
std_data_rate_mbps					x	x	x		x	x	x	x
med_data_rate_mbps					x	x	x		x	x	x	x
average_altitude_m					x	x	x	x	x	x	x	x
dayofweek				x	x	x	x	x	x	x	x	
retry_percentage_nonbeacon												
total_clients						x	x	x	x	x	x	x
average_rssi_dbm						x	x	x	x	x	x	x
num_ap						x	x	x	x	x	x	x
timeofday									x	x	x	
mac_source_address												
start_datetime												
nonbeacon_cat												
cat_80211g		x										
cat_80211n			x									
hour1	x	x	x									
month1	x	x	x									
day1	x	x	x									

Table 13 Summary of all models considered along with the number of trees and whether a warning occurred or not. An 'x' indicates that that particular variable was included in the model. The dependent variable in each case was the categorical variable nonbeacon_cat that was created to represent whether a retransmission had occurred.

Future work

Some other things worth considering may be different models, refining which models would be most optimal; ways in which R could be used to handle larger amounts of data; other tools such as python; look at predictive models for occupancy -- would just changing the dependent variables to one that represents occupancy be sufficient?

References

1. 802.11 WLAN Packet Types, URL: http://www.wildpackets.com/resources/compendium/wireless_lan/wlan_packet_types
2. Apple TV Q&A - Revised April 12, 2012, URL: http://www.everymac.com/systems/apple/apple-tv/apple-tv-faq/what-is-802.11n-differences-between-802.11n-802.11a-802.11b-802.11g.html
3. ATP, Beacons and Moving Block, URL: http://www.railway-technical.com/sigtxt3.shtml
4. Converting Addresses to/from Latitude/Longitude/Altitude in One Step, URL: http://stevemorse.org/jcal/latlon.php
5. Defining Spectrum, URL: http://www.gsma.com/spectrum/what-is-spectrum/
6. Dnl Institute, Decision Tree: A statistical and analytical tool for effective decisions, 2015, URL: http://dni-institute.in/blogs/decision-tree-a-statistical-and-analytical-tool-for-effective-decisions/
7. Dnl Insitute, Random Forest Using R: Step by Step Tutorial, 2015, URL: http://dni-institute.in/blogs/random-forest-using-r-step-by-step-tutorial/
8. Hu, Sanqing; Ouyang, Ye; Yao, Yu-Dong; Fallah, Hosein; Lu, Wenyuan, "A Study of LTE Network Performance based on Data Analytics and Statistical Modeling"
9. "Licensing Procedure for Spectrum Licenses for Terrestrial Services", Spectrum Management and Telecommunications, Client Procedure Circular, Industry Canada, CPC-2-1-23, Issue 4, October 2015.
10. Layman's Introduction to Random Forests, Edwin Chan, URL: http://blog.echen.me/2011/03/14/laymans-introduction-to-random-forests/
11. Machine learning, WhatIs.com, URL: http://whatis.techtarget.com/definition/machine-learning
12. Package 'randomForest', 2015, URL: https://cran.r-project.org/web/packages/randomForest/randomForest.pdf
13. Packet loss, Wikipedia, URL: https://en.wikipedia.org/wiki/Packet_loss#Causes
14. Practical Data Science with R, Nina Zumel, John Mount, Manning Publications Co. 2014.
15. Predictive analytics, Wikipedia, URL: https://en.wikipedia.org/wiki/Predictive_analytics
16. Radio Spectrum Allocations in Canada, URL: https://www.ic.gc.ca/eic/site/smt-gst.nsf/vwapj/2014_Canadian_Radio_Spectrum_Chart.pdf/$file/2014_Canadian_Radio_Spectrum_Chart.pdf
17. Retransmission (data networks), Wikipedia, URL: https://en.wikipedia.org/wiki/Retransmission_(data_networks)
18. "A Primer of Methods of SEA Spectrum Data Analysis", Jennifer Schellinck, Robert Warren, Patrick Boily.
19. SEA Architecture Team, CRC Spectrum Environment Awareness (SEA) Grand Challenge Initial Specification of SEA System Functional Architecture, Communications Research Centre Canada, Document No. SEA-ARCH-141008, Version 1.01, March 16, 2015.
20. SQL Tutorial, URL: http://www.w3schools.com/sql/default.asp
21. STAT4601/5703 Data Mining I course notes, URL: http://people.math.carleton.ca/~smills/

22. Yin, Sixing et al., “Mining spectrum usage data: a large-scale spectrum measurement study.” Mobile Computing, IEEE Transaction on 11.6 (2012), pp 1033-1046 (presentation by Loong Chan)
23. What’s the Difference between 2.4 and 5-Ghz Wi-Fi? (and Which Should You Use), How-To Geek, URL: http://www.howtogeek.com/222249/whats-the-difference-between-2.4-ghz-and-5-ghz-wi-fi-and-which-should-you-use/
24. Why Channels 1, 6 and 11?, metageek, URL: http://www.metageek.com/training/resources/why-channels-1-6-11.html
25. Breiman L, Cutler A, Random Forests, URL: https://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm#missing2
26. R Random Forests Variable Importance, Stackoverflow, URL: http://stackoverflow.com/questions/736514/r-random-forests-variable-importance
27. Random Forests, Metagenomics. Statistics. URL: https://dinsdalelab.sdsu.edu/metag.stats/code/randomforest.html
28. Gini Coefficient, Wikipedia, URL: https://en.wikipedia.org/wiki/Gini_coefficient#Alternatives_to_Gini_coefficient
29. Confusion matrix, Wikipedia, URL: https://en.wikipedia.org/wiki/Confusion_matrix