# Wifi Location Modelling - Course 4, Task 3

## IOT Analytics

Primary Objective:

- Evaluate the feasibility of using wifi signals to determine a customer's location in indoor spaces

Business Question:

- Can indoor location be predicted with enough accuracy with our models to warrant the development of a smartphone app to help customers.

## Data Set

For this analysis and modeling we used the UJIIndoorLoc data set, which provides a training data set with 19300 observations and 529 variables. Of these 529 variables, 520 represent different wifi access points (WAPs) and the associated wifi signal strength detected. For these 520 WAPs, signal strength ranges from 0 (strongest) to -100 (weakest), with no-signal-detected values being recorded as 100. The remaining variables either describe the location directly (building, etc) or are data capture data points (phone used, longitude/latitude, user).

## Exploratory Data Analysis

The data set contains the longitude and latitude coordinates associated with each reading. To validate the integrity and high-level accuracy of this GPS-based data, the original building layout described within the documentation was compared to the longitude and latitude pairs of the entire data set. As can be seen in Figure 1, no data points appear to exist outside of the expected regions, validating the quality of the longitude and latitude readings.

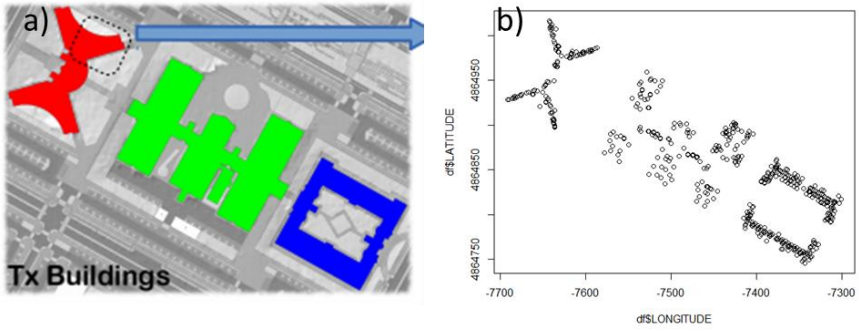


*Figure 1: a) Building layout documented in the ULIIndoorLoc data set, b) Longitude/Latitude signal pairs in the data set*

Signal strength, when detected, varied significantly across the data set. Through data stacking and removing the "no-signal" values, we are able to see the distribution of received signal intensities across all of the WAPs in our data set. As shown in Figure 2, the majority of the reported signals were in the weaker range (-80dB and below).

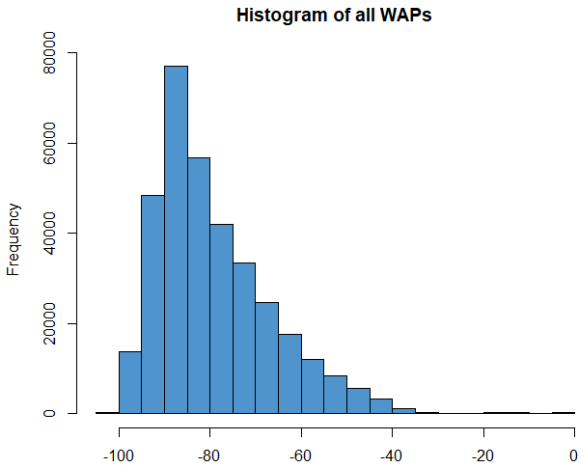


*Figure 2: Distribution of WAP signal intensity*

## Data Preparation

After initial cleaning, the 4 variables used to directly describe the location (BUILDINGID, FLOOR, SPACEID, RELATIVEID) combined into a single variable, with letters segregating for easier human reading (ex: "B0 F1 S106 R2" = Building 0, Floor 1, Space 106, Relative 2). This combined variable, named "Location" is our dependent variable. This, as well as other categorical variables were converted to type *factor* for this classification modelling. The TIMESTAMP variable was removed – while it could help predictions in this data set, as it was deemed irrelevant for predicting in a live use case when a user queries for location in real-time.

## Training Sets

### "SAMP"

The first data set is merely a 25% sample of the data set created from Data Preparation. This set contains 5047 observations and 525 variables.

### "SP_SAMP"

The RELATIVEID variable is used to indicate whether the test subject was directly within a given SPACEID, or standing in the hallway next to it. In practical usage if these models could indicate the building, floor, and space for user, they are likely able to use common sense to determine if they are in an actual room or a hallway. As such, this data set was created to ignore the RELATIVEID, with our dependent variable being a combination of BUILDINGID, FLOOR, and SPACEID. This data set also used the same 25% sampling as "SAMP". The goal is to determine if higher predictive accuracy can be achieved by removing the need to predict RELATIVEID, thereby reducing the number to total factors in the dependent variable.

### "FE"

For the feature engineered data set, the goal was to remove the many, many data points in the WAP variables where no signal was detected and streamline the data set. While it can be important for modelling to know which WAPs *weren't* detected, this data set focuses on isolating the strongest three WAP signal intensities for each row, along with the associated WAP ID's. As seen in Figure 3, the strongest WAP signal for the row is set as "sig1", with the associated WAPID as "sig1_WAP", and so on for "sig2" and "sig3". The "FE" data set was not sampled, and consists of 19300 observations in 11 variables.

| Index | sig1 | sig1_WAP | sig2 | sig2_WAP | sig3 | sig3_WAP | LONGITUDE | LATITUDE | USERID | PHONEID | Location |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | -53 | 173 | -54 | 172 | -67 | 90 | -7541.264 | 4864921 | 2 | 23 | B1 F2 S106 R2 |
| 2 | -46 | 90 | -46 | 90 | -62 | 172 | -7536.621 | 4864934 | 2 | 23 | B1 F2 S106 R2 |
| 3 | -61 | 173 | -61 | 172 | -66 | 90 | -7519.152 | 4864950 | 2 | 23 | B1 F2 S103 R2 |
| 4 | -55 | 172 | -56 | 173 | -67 | 104 | -7524.57 | 4864934 | 2 | 23 | B1 F2 S102 R2 |

*Figure 3: FE data set showing top-3 WAP signals and WAP IDs for the first 4 observations*

## Modelling

For this classification task, three models were chosen (kNN, Random Forest, C5.0 Decision Tree) and applied to each of these three data sets. KNN was selected because it is generally considered good for data sets with large numbers of observations. Random Forest was chosen because, although

considered to have high convergence times, it can generally provide high accuracy. C5.0 was chosen because, like KNN, it is a low bias/high variance algorithm that does well with large data sets.

| Algorithm | DataSet | Min | 1st Quartile | Median | Mean | 3rd Quartile | Max | Test Set Accuracy | Model Generation Time (s) |
|---|---|---|---|---|---|---|---|---|---|
| KNN | Samp | 0.304 | 0.310 | 0.318 | 0.322 | 0.330 | 0.347 | 0.390 | 520 |
| KNN | Sp_Samp | 0.322 | 0.330 | 0.334 | 0.344 | 0.359 | 0.373 | 0.437 | 529 |
| KNN | FE | 0.678 | 0.680 | 0.690 | 0.689 | 0.696 | 0.702 | 0.711 | 5825 |
| RF | Samp | 0.887 | 0.896 | 0.900 | 0.899 | 0.903 | 0.909 | 0.958 | 14545 |
| RF | Sp_Samp | 0.893 | 0.898 | 0.904 | 0.903 | 0.908 | 0.912 | 0.967 | 6510 |
| RF | FE_mini | 0.496 | 0.496 | 0.496 | 0.496 | 0.496 | 0.496 | 0.563 | 1344 |
| C50 | Samp | 0.867 | 0.877 | 0.881 | 0.879 | 0.882 | 0.888 | 0.928 | 2544 |
| C50 | Sp_Samp | 0.867 | 0.874 | 0.891 | 0.883 | 0.891 | 0.892 | 0.937 | 1647 |
| C50 | FE | 0.974 | 0.974 | 0.979 | 0.977 | 0.979 | 0.980 | 0.981 | 5039 |

| Algorithm | DataSet | Min | 1st Quartile | Median | Mean | 3rd Quartile | Max | Test Set Kappa | Model Generation Time (s) |
|---|---|---|---|---|---|---|---|---|---|
| KNN | Samp | 0.302 | 0.308 | 0.317 | 0.320 | 0.329 | 0.346 | 0.389 | 520 |
| KNN | Sp_Samp | 0.321 | 0.329 | 0.333 | 0.342 | 0.358 | 0.371 | 0.436 | 529 |
| KNN | FE | 0.678 | 0.679 | 0.690 | 0.688 | 0.695 | 0.702 | 0.711 | 5825 |
| RF | Samp | 0.887 | 0.896 | 0.900 | 0.899 | 0.903 | 0.908 | 0.958 | 14545 |
| RF | Sp_Samp | 0.893 | 0.898 | 0.903 | 0.903 | 0.907 | 0.912 | 0.967 | 6510 |
| RF | FE_mini | 0.496 | 0.496 | 0.496 | 0.496 | 0.496 | 0.496 | 0.562 | 1344 |
| C50 | Samp | 0.867 | 0.876 | 0.881 | 0.879 | 0.882 | 0.888 | 0.928 | 2544 |
| C50 | Sp_Samp | 0.866 | 0.874 | 0.891 | 0.883 | 0.891 | 0.892 | 0.937 | 1647 |
| C50 | FE | 0.974 | 0.974 | 0.979 | 0.977 | 0.979 | 0.980 | 0.981 | 5039 |

*Figure 4: Modelling results*

As shown in Figure 4, data was collected after training through `resample()`, and was also compared to the `postResample()` data after using the model to predict on the Test data sets. The model generation time is also included, as it was significant for this exercise. Surprisingly, the "FE" data set, while having a much smaller overall data size, took significantly longer to generate a model for all model types. However, it appears worthwhile, as the "FE"-based C5.0 model boasts an impressive median accuracy and kappa of 0.979. This accuracy is maintained when applied to the Test data segment, achieving an accuracy and kappa of 0.981. All three data sets performed well with the C5.0 algorithm. The performance can also be visualized in Figure 5, illustrating the strong performance of the "FE" data set with C5.0 Decision Tree. Also of note, our "SP_SAMP" data set showed no major accuracy gains of the "SAMP" data set, indicating that removing RELATIVEID didn't significantly aid our predictive capabilities.

## Recommendations

This trial has shown that modelling indoor location using wifi signal strength and other complementary variables can successfully provide highly accurate predictions for location. This methodology can be used for future deployments.

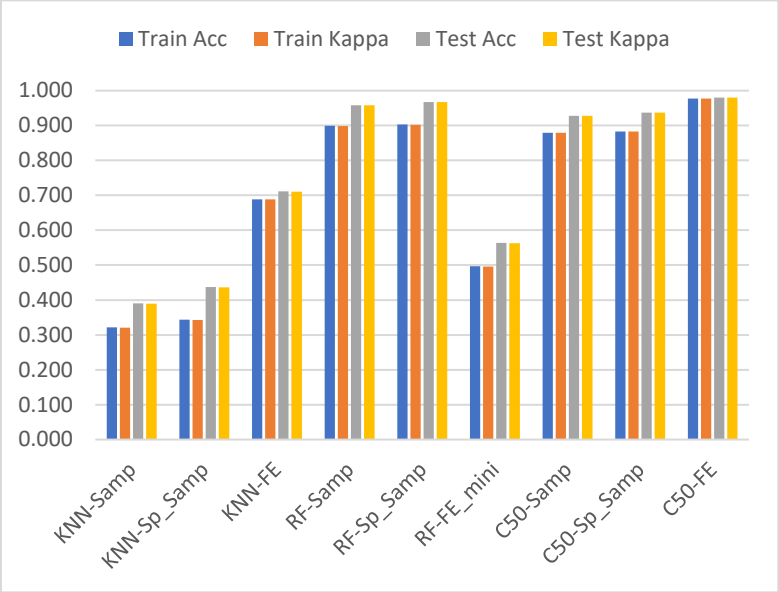


*Figure 5: Model accuracy and kappa for training and test data sets*

Machine learning classification tasks such as these can be quite challenging primarily due to the large number of classes within a location dependent variable. This problem scales directly with the specificity that the algorithm tries to give the user in terms of precise location. As a side effect of having a high number of classes, it can be challenging for data sampling to provide enough instances of each class to generate a model. As a result, models are built on a large number of observations, consuming significant computing resources and time in model generation. This reduces the ability for fine tuning of promising models within a given project deadline. Taking this into account, future deployments of this technology should work to limit the number of classes for the location variable.

For further improving accuracy, time-stamping would be an attractive addition to the data set. In real-time-location-systems (RTLS), wifi tags can send a message that is received by multiple WAPs. As this message has a specific time stamp associated with it, and due to the fact that all WAPs are time synchronized, the transmit delays can be used to triangulate the wifi tag in real-time. Specifically, if at least 3 WAPs receive the message, the tag can be accurately triangulated in 3-D space, which is much more useful in indoor deployments than the 2-D longitude/latitude provided by GPS. However, it will require further investigation on how to institute this technology through a smartphone locationing app instead of a dedicated hardware wifi tag.