Indoor Positioning System

Pascal Caraccioli Salinas

Vix Edwards

Tram Anh Hoang

Sung J Oh

Portland State University

STAT4/510: Basic Consulting Skills

EXECUTIVE SUMMARY

This project summarizes the findings and recommendations of an indoor positioning system (IPS) using Wi-Fi strength signals.

The primary objective is to implement a robust model that can accurately predict the location of a device connected to the local Wi-Fi network. We implemented two supervised machine learning methods (XGBoost and K-Nearest Neighbors) to create two separate models. Notably, the two models have predictive skills, but smaller errors correspond to XGBoost. This last approach also has the advantage of being computationally efficient.

Our recommendations to improve the accuracy of the models without incurring additional costs include optimizing the location of the routers throughout the floor plan to ensure an even distribution of the signals.

# Background

Rephrase the problem and any relevant issues present. Define any technical terms needed for the paper.

Background of the problem. Discuss other related analyses (information on authors that have addressed similar problems and how they approached solutions)

# Data

The following section, describes and characterizes the data set provided by the client.

The client gridded 540m2 of their corporate building into {insert number} 1 -meter-by-1-meter cells in which measurements of signal strength from 6 access points (Wi-Fi routers) were obtained from a handheld device connected to a local Wi-Fi network. The data are subdivided into two sub-sets, “offline” and “online”, distinguished by the fact that the offline data set was sampled at fixed locations and orientations, versus the online data set that was sampled at random locations and orientations.

The offline data, intended to train a model, was collected at all the 166 fixed points within the grid. At each location, the device was oriented in 45-degree increments, starting from 0 up to 360 degrees, resulting in 8 angles (i.e., 0, 45, 90, 135, 180, 225, 270, and 315). Signal strength for the access points was measured for each orientation a total of 110 times.

The offline data, aimed at model training, was systematically gathered at all 166 fixed points within the grid. At each location (x, y), the device was oriented in 45-degree increments from 0 to 360 degrees, resulting in 8 angles, and the signal strength from each access point was measured 110 times. This translates to 110 samples per angle at each location, totaling 880 samples per location and accumulating to a comprehensive dataset of 146,080 observations.

The online data was designed to simulate real-world data (i.e., locations that are not bounded to a grid, and which a device can be oriented at random.) Specifically for the online data, 60 combinations of orientation/locations were randomly selected, and then sampled 110 times, resulting in 6600 measurements in total.

More details of the floor plan, and location of online and offline data can be seen in Figure 1. Circles serve as markers for the positions where offline measurements were conducted, while black squares indicate the locations of the six access points. The positions of the access points were provided in a separate file by the client.

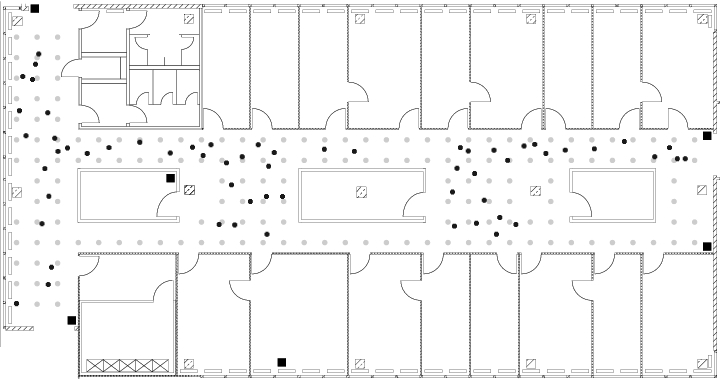


Figure 1: Flooplan location. Access points are squares. Grey dots are offline data locations and black dots are online data locations.

According to documents provided by the client, the data contains the following variables:

• time: time in miliseconds since midnight 01/01/1970 UTC  
• scanMac: IP address of the scanning device, in mm:mm:mm:ss:ss:ss. • pos: the 3-D coordination of the scanning device (x,y,z)  
• orientation: the scanning device’s orientation.  
• mac: the IP address of the access points.  
• signal: signal strength in dBm.  
• channel: the channel frequency.  
• type: type of device (access point = 3, device in adhoc mode =1)

The raw data are stored in a .txt file. The first 3 lines of the data are characterized by the hash (#) symbol, followed by a row that contains all the variables in one line, separated by a semi-colon. A sample of the initial format of the data can be seen below. We start by eliminating the rows that start with the hash symbols from the data set using a strsplit function. The resulting data set contains a total of 146080 rows, and therefore 5312 rows were eliminated. This value, if divided by 3 (first 3 rows contain a hash) is 1770, which is close consistent with the expected number of locations (166) and angles (8). This means that each stack location/orientation combination that contains 110 samples (from herein Location\_orientation stack) was separated by 3 hash symbols.

Some variable names do not correspond to the ones given by the client, for example, orientation is degree in the data set, and the variables type, channel, signal do not have an explicit name. We also note that for pos the x,y,z variables are grouped together, and the mac variable includes signal, channel and type separated by comas. We can then distinguish between single variables (defined by a name and have one value), and secondly composite variables (defined by containing multiple values for one parameter).

We use these patterns to create a matrix with the variables. For this, we created a function that first, separated all the data separated by a semi-colon, a comma, or an equal symbol. Then, we selected the rows corresponding to mac, signal, type and created a matrix that has the information for the specific access point. Lastly, we bind all the information together in a large data frame that contains one row per location/orientation and access point.

Before further exploration and analysis of the data, we conducted converted the variables into the correct types (as defined by the documents provided by the client).

The summary is as follows:

1. 1)  The variables position, orientation, signal and channel were converted to numerical values.
2. 2)  The variable time was converted into a time value using as origin midnight on January 1st, 1970. The original variable was kept in the data set as rawTime in case it becomes necessary for future analysis.
3. 3)  The variable type has binary values of 1 and 3. The documentation explains that type = 3 corresponds to ad-hoc devices, that are not needed for the development and testing of the IPS and therefore, after

removing the rows with a value of type equal to 3, we remove the variable from the data set.

1. 4)  For the exploration, we remove the scanMac, as information given by the client indicates that one

devices was used.

The first rows of the formatted data can be seen in Table 1.

A quick analysis of the numerical data shows that posZ has only zero values (Table 2). This seemingly anomalous value is due to the fact that all of the readings were taken on one floor of the building. We, therefore, removed the posZ variable from the data set. Furthermore, we detect anomalous values for orientation.

# Methods

**K-Nearest Neighbors (KNN) Algorithm**

Your indoor environment is like a canvas with distinct features. Such as Wi-Fi signal strength, and orientation of devices. These features define the nuances of your space, creating a unique fingerprint for each location. Imagine each location as a "friend" in your indoor space. KNN is like asking, "Which locations are most similar to this one?" We consider the proximity of these friends based on the features you've gathered. The algorithm doesn't assume a specific pattern; it adapts to the individual characteristics of your space.

During our comprehensive exploration of various machine learning models to utilize for this project, we carefully considered the KNN because it is a powerful and intuitive method for certain applications. We have discovered challenges that would make this a costly approach for this project though. Some key considerations are the computational demands of KNN, this model involves calculating the distances between data points, which can be computationally demanding especially given the high dimensionality of your data set. KNN tends to degrade as the number of features increases. The “k” part of KNN is how many neighbors we wish to consider in making our prediction, increasing the k value improves accuracy but dramatically drives up the computational costs.

**A diagram of mathematical calculations

Description automatically generated**

This is the technical representation of the K-NN method. This method had underlying assumptions of stationarity, which is to say that the neighboring points are not moving. It also assumes equal importance of features; this algorithm isn’t equipped to deal with particular weights without additional preparation of the features.

**XGBoost**

State the assumptions, explain method

# Results

State the hypotheses

Show the evidence in favor or against the hypotheses

Present testing results and interpret such results

# Conclusions and recommendations

Discuss the broad implications of the project

Use rmodel result to answer the client’s questions and goals of the project

Discuss (i) issues that merit further exploration, (ii) interesting findings that are not part of the client’s questions

Mention reservations about the analyses that may require more complex modeling (due to modeling assumptions not holding, other reasons)

References

Last Name, A. B. (Year). Article Title. Journal Title, Pages #-#. URL. URL.

Last Name, C. D. (Year). Book TitleBook Title URL.

Last Name, D. E., Last Name, F. G. (Year). Report TitleReport Title URL.

Last Name, H. I. (Year, Month Day). Article Title/Headline. Periodical.Periodical.

Organization Name. (Year, Month Day).Webpage Title. URL.

Footnotes

1For APA reports, add footnotes manually on their own page following references. Do not use the **Insert** **Footnotes** method on the **References** tab as they will not be formatted correctly. For APA formatting requirements, it’s easier to type your own footnote references and notes. To format a footnote reference, select the number and then, on the **Home** tab, in the **Styles** gallery, click **Footnote Reference**. The body of a footnote, such as this example, uses the **Normal** text style. If you delete this sample footnote, don’t forget to delete its in-text reference at the end of the sample Heading 2 paragraph on the first page of body content in this template.

Tables

Table 1

Table Title

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column Head | Column Head | Column Head | Column Head | Column Head |
| Row Head | 123 | 123 | 123 | 123 |
| Row Head | 456 | 456 | 456 | 456 |
| Row Head | 789 | 789 | 789 | 789 |
| Row Head | 123 | 123 | 123 | 123 |
| Row Head | 456 | 456 | 456 | 456 |
| Row Head | 789 | 789 | 789 | 789 |

Note: Place all tables for your paper in a tables section, following references and footnotes. Start a new page for each table, include a table number and table title for each, as shown. All explanatory text appears in a table note that follows the table, like this one. Use the **Table/Figure** style, available on the **Home** tab, in the **Styles** gallery, to get the spacing between table and note. Tables in APA format can use single or 1.5 line spacing. Include a heading for every row and column, even if the content seems obvious. A default table style has been set up for this template that fits APA guidelines. To insert a table, on the **Insert** tab, click **Table**.

Figures Title

Figure 1.

Include all figures in their own section, following references, footnotes, and tables. Include a numbered caption for each figure. Use the Table/Figure style for easy spacing between figure and caption.



# Appendix