



Predicting

Automatic Indoor localization remains a problem unsolved mainly due to the lack of Global Positioning Signal (GPS). Wi-Fi fingerprinting has been a popular approach to solve this problem by utilizing the intensity of values across multiple Wireless Access Points (WAPs) to localize a target with respect to the location of the WAPs. Four machine learning methods, K-Nearest Neighbor (KNN), Decision Tree (DT), Random Forest (RF), and Support Vector Machine (SVM), were implemented to predict a target's latitude, longitude, and altitude. The K-Nearest Neighbor (KNN) model had the highest accuracy estimating the target with a mean error of 1.78 meters.

Data

The dataset was pulled from UJIIndoorLoc and consists of 21,048 samples that span across three buildings and four floors.

Each sample consists of:

- Features: 520 WAPs
- Quantitative Target Labels: Latitude and Longitude
- Categorical Target Labels: Building ID and Floor ID



Figure 1. Visual of Buildings in Dataset

Data Preprocessing

The raw data consists of 520 WAP intensity values that range from -104dB to 0dB with null signals denoted at 100dB. The following steps were performed for preprocessing:

- Null and WAP values less than -98dB were replaced with -98dB
- WAP values normalized between 0 and 1
- Number of features reduced to 96 features, or 95% of the explained variance using Principal Component Analysis (PCA)
- Removed samples with less than 9 active WAPs
- Removed samples labeled Phone ID = 17

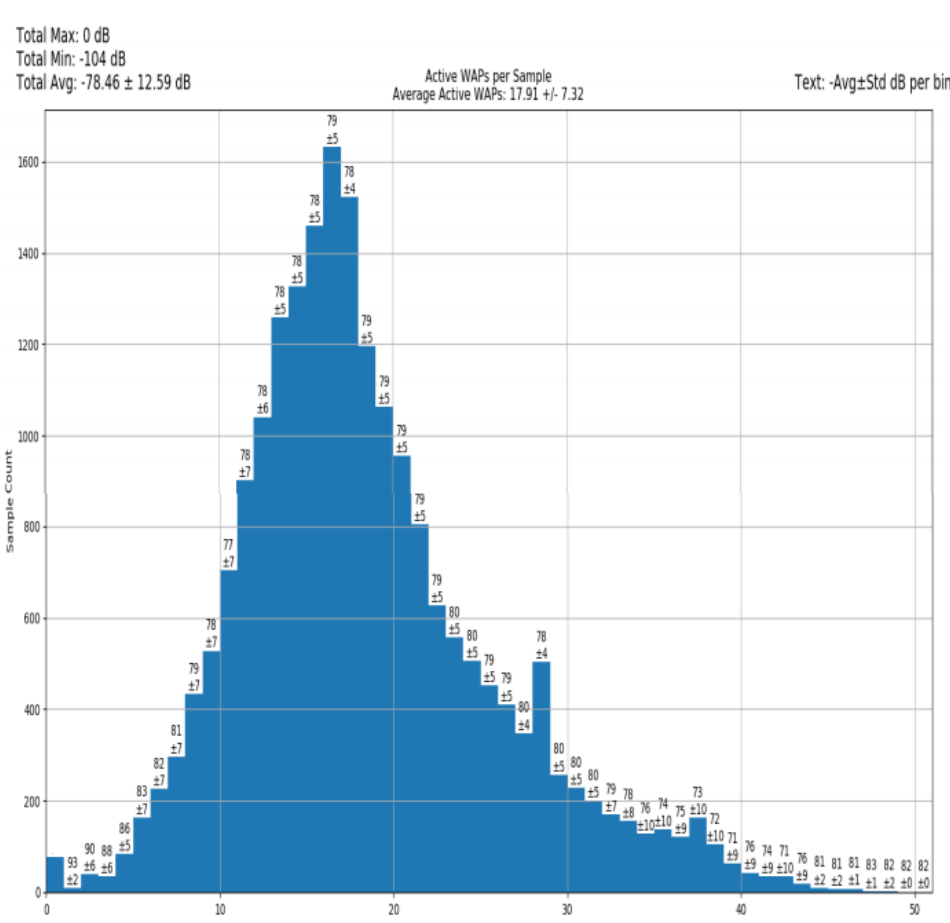


Figure 2. Number of Active WAPs

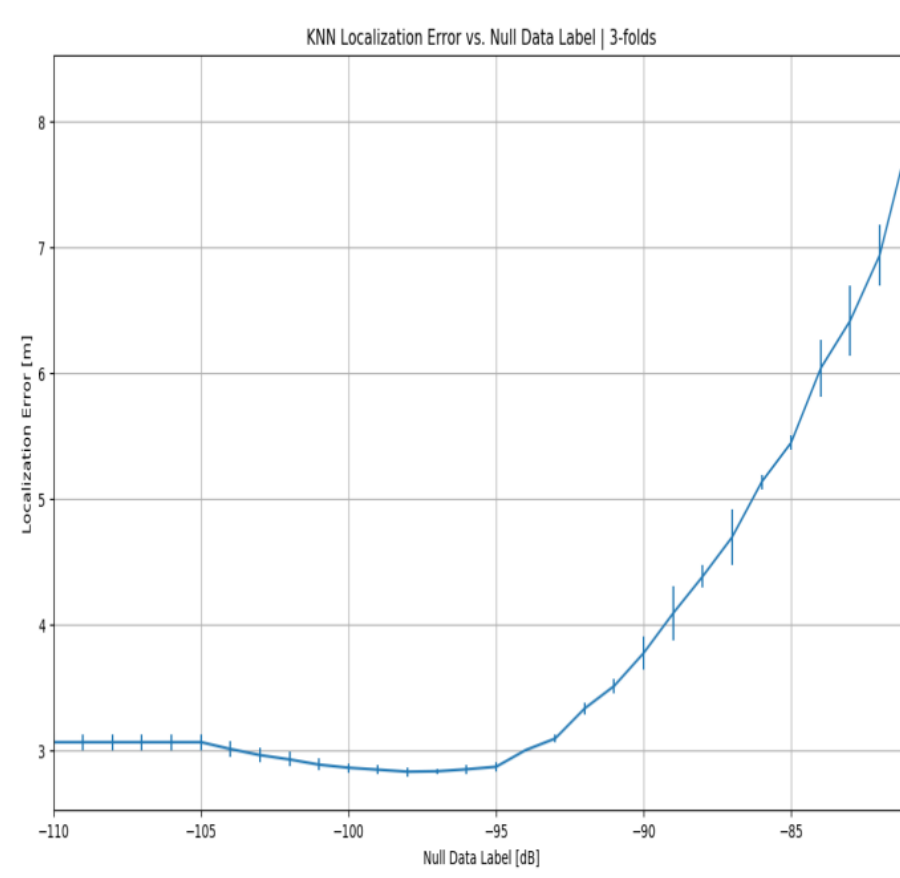


Figure 3. Error with Varying Null

Methods

Four machine learning methods were implemented, and a grid search was performed on each method for hyper-parameter tuning validated by a 5-fold cross validation. The four methods are as follows:

- K-Nearest Neighbor
- Decision Tree
- Random Forest
- Support Vector Machine

Model 1—KNN

K-Nearest Network takes k closest training examples to predict the new target. Parameters:

- $K = 1$
- Distance Calculation: manhattan distance

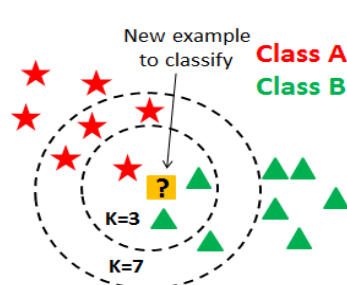


Figure 4. KNN Example

Model 2—DT

Decision Trees are a tree structured flowchart where each node denotes a test of an attribute, and each branch represents the respective outcome.

Model 3—RF

Random Forest builds an ensemble of decision trees and combines the outputs for a more accurate prediction. Parameters:

- Number of Trees: 500

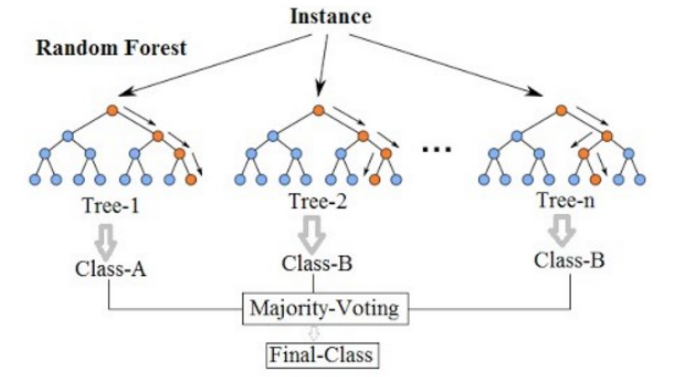


Figure 5. RF Example

Model 4—SVM

Support Vector Machines are models that are associated with learning algorithms for both classification and regression. Parameters:

- $C = 100$
- Kernel: Linear

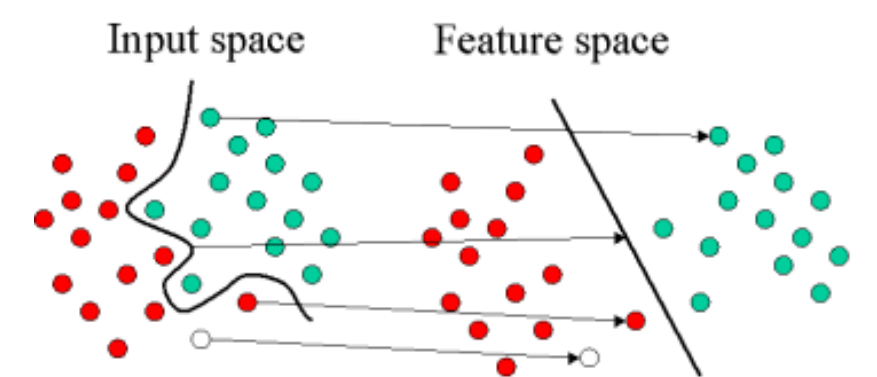


Figure 6. SVM Example

Results

The dataset of 21,048 samples was reduced to 19937 samples after data preprocessing and divided into 80/20 splits of training/test sets, resulting in 15949 samples of training data and 3988 samples of test data.

Model	Training Mean Coordinate Error (m)	Training R2 Score	Test Mean Coordinate Error (m)	Test R2 Score
KNN	0.44	0.99259	1.78	0.99542
DT	0.41	0.99717	4.56	0.99395
RF	1.75	0.99693	4.09	0.99659
SVM	45.56	0.79952	57.19	0.80610

Table 1. Comparison of Model

K-Nearest Neighbor predicted the target's location with the least error, while Random Forest had the highest R2 score.

Discussion

SVM produced terrible results because the WAP intensities overlapped and the data was not separable regardless of the kernel used. Although the KNN model predicted the location of the target with the least mean error, it had a lower R2 score than RF because the outliers were more severe. Since the KNN model was optimized at $k=1$, we believe the KNN model simply memorizing data. We concluded that this dataset was large and comprehensive enough for a KNN to accurately predict a target's location, but a Random Forest creates a better generalized model. For indoor localization using Wi-Fi fingerprinting, the dataset is much more significant than the type of model used for machine learning. Our results had the greatest improvements from data preprocessing rather than optimizing the models.

Future Work

An attribute in the dataset, Phone ID, was not considered to build the model when it could have provided valuable information on distinguishing good vs. bad data. Given more time, we would explore other datasets and compare the results.

References

[1] Joaquin Torres-Sospedra Raul Montoliu Adolfo Martinez Joaquin Huerta. UJI - Institute of New Imaging Technologies, Universitat Jaume I, Avda. Vicente Sos Baynat S/N, 12071, Castelln, Spain. UPV - Departamento de Sistemas Informticos y Computacin, Universitat Politcnica de Valncia, Valencia, Spain. <https://archive.ics.uci.edu/ml/datasets/ujiindoorloc>

[2] Hong, Feng Zhang, Yongtuo Zhang, Zhao Wei, Meiyu Feng, Yuan Guo, Zhongwen. (2014). WaP: Indoor localization and tracking using WiFiAssisted Particle filter. Proceedings - Conference on Local Computer Networks, LCN. 210-217. 10.1109/LCN.2014.6925774.

[3] Pedregosa, F. and Varoquaux, G. and Gramfort, A. and Michel, V. and Thirion, B. and Grisel, O. and Blondel, M. and Prettenhofer, P. and Weiss, R. and Dubourg, V. and Vanderplas, J. and Passos, A. and Cournapeau, D. and Brucher, M. and Perrot, M. and Duchesnay, E. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12. 2825– 2830. 2011.

Contact

Matt Hong: mattjoo.hong@gmail.com
So Sasaki: sosasaki@ucsd.edu

Ryan Clark: rmclark@ucsd.edu
Sophia Huang: jih201@ucsd.edu