COMP4336/9336

Mobile Data Networking 2020 Term 2

Project

Distance Estimation using Wireless Signal Strength

Stage – 3

Name: Rui Li

zID: z5202952

Contents

My approach

Abstract

I use the algorithm in scikit-learn to train a model and make the prediction. After comparing different algorithm (with the method of cross-validation) in scikit-learn, I choose the LinearRegression as my algorithm.

How do I perform this approach?

- Use Microsoft Network Monitor 3.4 to collect the data.
- Use Wireshark to apply the filter and output as CSV files.
- Accessing the data with Pandas
- Pre-processing the data.
- Use Pandas to visualise the dataset.
- Use Scikit-learn to train the model and predict.

How to use my model?

I have upload my approach into GitHub: https://github.com/lrlrlrlr/COMP9336_project_20t2

Dataset

In order to improve the quality of the dataset and make the prediction more accurate, I have collected data for 5 times both indoor and outdoor; for each time, I have spent more than 30min for each distance(from 1m to 10m). After applying the filter, there are **241028** valid data.

Date	Indoor data	Outdoor data	total
05/07/2020	24567	25604	50171
15/07/2020	23308	24588	47896
24/07/2020	23577	22991	46568
26/07/2020	24117	24549	48666
28/07/2020	23241	24486	47727
Total	**118810**	**122218**	**241028**

Table 1.1 Number of valid data

Average RSS

The average RSS for indoor is -38.05dBm, for outdoor is -40.02dBm. In outdoor, the signal is slightly weaker than indoor within same distance.

And there is an interesting thing, the average RSS of this dataset shows that the 3m RSS (33.23dBm) is even higher than 2m's RSS (34.27dBm) indoor!
I think that might have 2 reasons:

- When the distance is 2 meters, I will put the phone next to the corner of the wall, which may affect the RSS.
- The RSS is just fluctuating around -34 dBm with distance between 2-4m, there is no large or significant signal loss.

	Distance (m)										Average
RSS (dBm)	1	2	3	4	5	6	7	8	9	10	
Indoor	31.38	34.27	33.23	34.53	39.02	36.40	40.65	42.19	43.75	45.12	-38.05
Outdoor	31.18	33.61	34.96	35.29	38.07	39.68	42.73	47.05	47.80	49.84	-40.02
Average	31.28	33.94	34.09	34.91	38.55	38.04	41.69	44.62	45.78	47.48	

Table 1.2 Average RSS for Indoor and outdoor

Dataset Distributions

For the RSS distribution, I have drawn the violin plot to observe the distribution for both indoor and outdoor. I found the shape of the violin between indoor and outdoor is different with the same distance. The standard deviation of outdoor RSS Distributions is higher than indoor RSS Distributions. The reason of that may be multipath or fluctuations in transmit power.

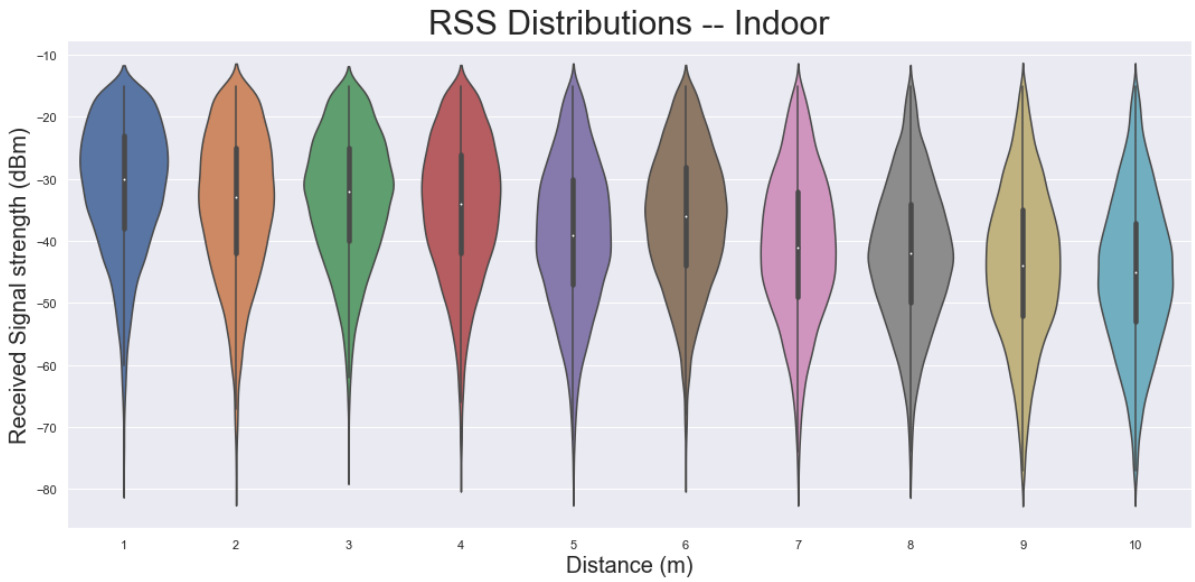


figure 1.3 RSS Distributions Indoor

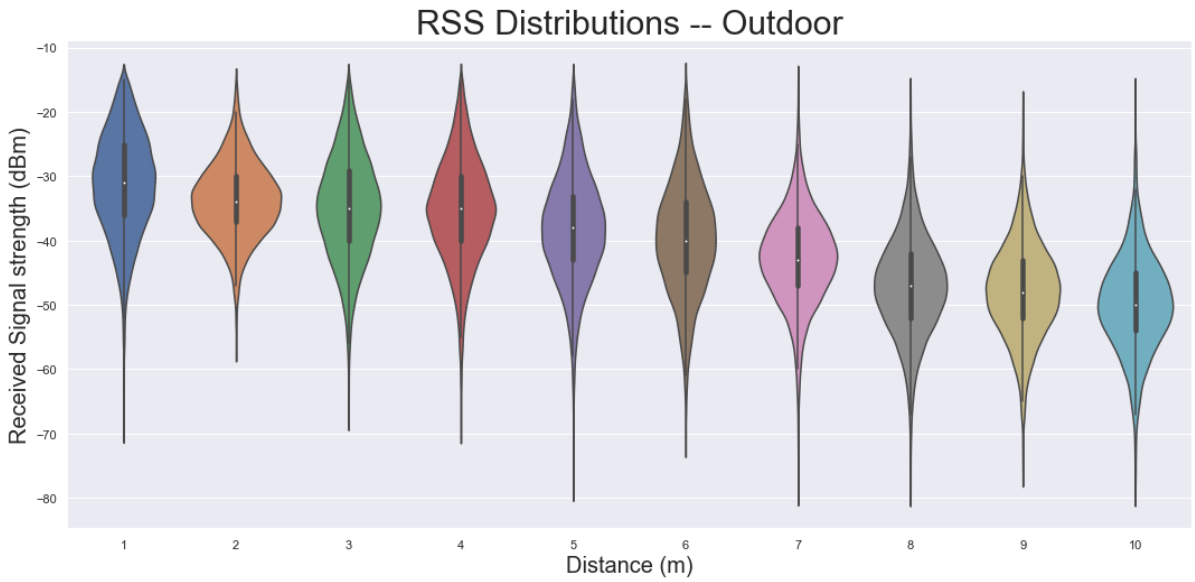


figure 1.4 RSS Distributions Outdoor

Dataset Describe

distance(m)	count	RSS mean (dBm)	RSS standard deviation (dBm)
1	10519	-31.38	10.58
2	11135	-34.27	11.59
3	11960	-33.23	10.33
4	12637	-34.53	11.15
5	11775	-39.02	11.76
6	10789	-36.40	10.82
7	11617	-40.65	11.81
8	13022	-42.19	11.21
9	12435	-43.75	12.26
10	12921	-45.12	12.17

Table 1.5 Number of valid data

distance(m)	count	RSS mean (dBm)	RSS standard deviation (dBm)
1	11497	-31.18	7.89
2	12287	-33.61	5.83
3	12358	-34.96	8.11
4	11210	-35.29	8.06
5	13664	-38.07	8.32
6	11744	-39.68	8.62
7	12400	-42.73	7.17
8	12602	-47.05	7.57
9	11843	-47.80	7.28
10	12613	-49.84	7.48

Table 1.6 Number of valid data

Algorithm chosen

To choose the most suitable algorithm, I have done a lot of research of the principles of supervise machine learning. I have tried 8 different algorithms, including:

- Classifier
- Linear Discriminant Analysis
- Regression
- Neural network

Classifier

SVC

Support Vector Classification (SVC) is the use of support-vector machines (SVMs). Which are supervised learning models with associated learning algorithms that analyse data used for classification and regression analysis.

DecisionTreeClassifier

A decision tree is a flowchart-like structure in which each internal node represents a “test” on an attribute (e.g. whether a coin flip comes up heads or tails), each branch represents the outcome of the test, and each leaf node represents a class label (decision taken after computing all attributes) (Wikipedia, 2020).

KNeighborsClassifier

k-nearest neighbors algorithm is a non-parametric used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space (Wikipedia Contributors, 2019).

GaussianNB

Gaussian Naive Bayes is a variant of Naive Bayes that follows Gaussian normal distribution and supports continuous data. ... Naive Bayes are a group of supervised machine learning classification algorithms based on the Bayes theorem. It is a simple classification technique, but has high functionality(Perez, 2006).

inear Discriminant Analysis

Linear discriminant analysis (LDA), normal discriminant analysis (NDA), or discriminant function analysis is a generalization of Fisher's linear discriminant, a method used in statistics, pattern recognition, and machine learning to find a linear combination of features that characterizes or separates two or more classes of objects or events. The resulting combination may be used as a linear classifier, or, more commonly, for dimensionality reduction before later classification (Wikipedia Contributors, 2019).

Regression

LogisticRegression

Logistic regression is a statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression is estimating the parameters of a logistic(Wikipedia Contributors, 2019).

LinearRegression

In statistics, linear regression is a linear approach to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). The case of one explanatory variable is called simple linear regression(Wikipedia Contributors, 2019).

Neural Network

MLPClassifier

A multilayer perceptron (MLP) is a class of feedforward artificial neural network (ANN). ... MLP utilizes a supervised learning technique called backpropagation for training. Its multiple layers and non-linear activation distinguish MLP from a linear perceptron. It can distinguish data that is not linearly separable (Wikipedia Contributors, 2019).

K-fold cross-validation

K-fold cross-validation is a statistical method used to estimate the performance of the machine learning models (Bengio 2004). Therefore, it is commonly used to select a model for a given modelling problem.

I perform a cross validation (cv=5) to compare the accuracies and mean square error of the algorithms:

Algorithms	ACCURACY		MSE	
	indoor	outdoor	indoor	outdoor
SVC	**0.557572**	**0.432155**	7.264182	7.457828
DecisionTreeClassifier	0.557556	0.430916	7.264199	7.377803
KNeighborsClassifier	0.4952	0.38686	7.518103	7.862156
GaussianNB	0.224106	0.334433	12.79446	7.35948
LinearDiscriminantAnalysis	0.213901	0.341746	12.94205	7.55768
LogisticRegression	0.180013	0.335556	12.99915	7.474387
MLPClassifier	0.111522	0.111111	13.77123	8.4691539
LinearRegression	-	-	**6.234443**	**5.942667**

Table 1.7 Scikit-learn Algorithm performance comparison

PS: There is no accuracy score for LinearRegression due to the prediction output of the model is float but actual distance is int. (e.g. LinearRegression(-20dBm) → 3.15m, but actual distance is 3.00m).

For accuracy, the SVC (C-Support Vector Classification) model is a good choice. But for MSE, the LinearRegression is the best choice.

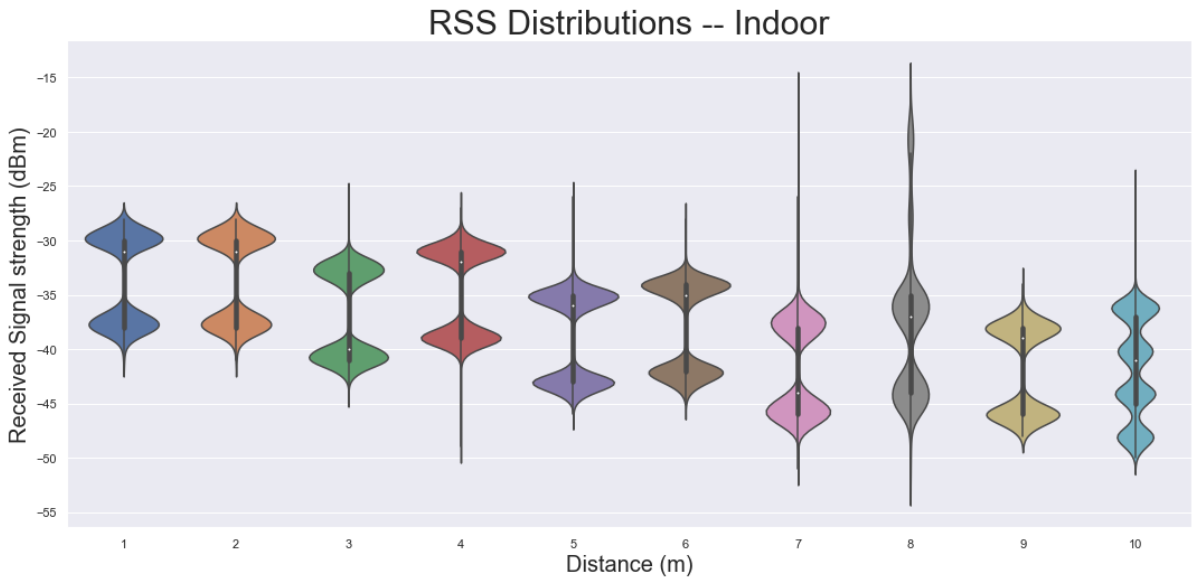
I think the RSS-Distance model is much like a linear model as the data distribution. Therefore, the LinearRegression algorithm can always get a closer prediction result than other algorithms in this case (while it always cannot get the correct result).

Improvement compare to stage 1

Dataset

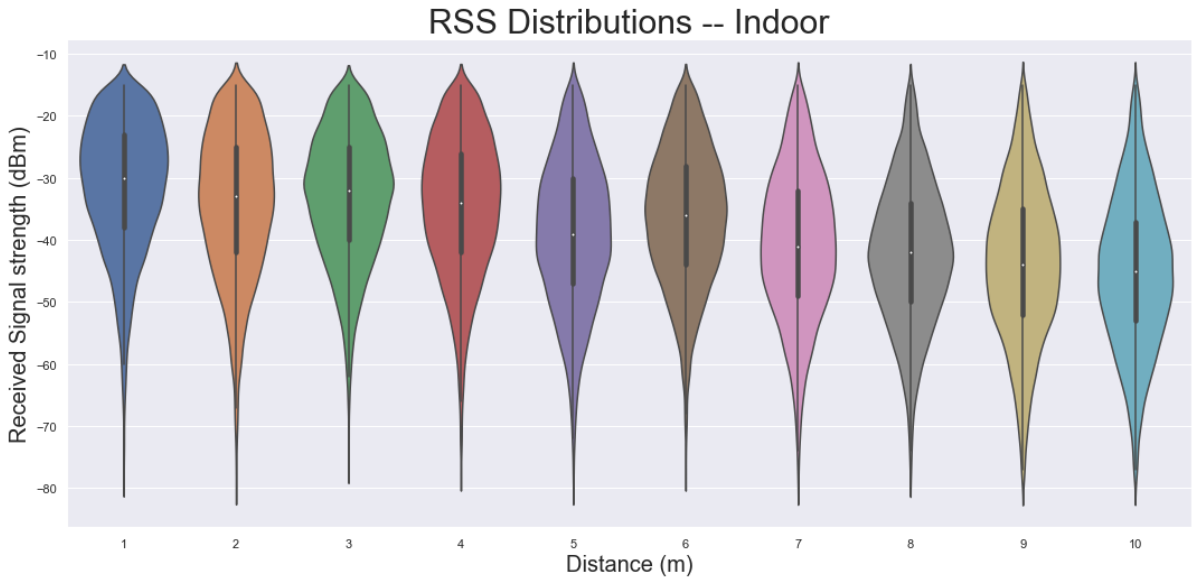
According to my tutor's advice, I need to make sure the data was collected in the same condition every time. For example, I should not hold the AP device or walkaround while data collection. In stage 1, my RSS distribution was combined by 2 datasets.

Stage 1:



Graph 2.1 Graph RSS Distributions for outdoor – Stage1

Stage 3:



Graph 2.2 Graph RSS Distributions for outdoor – Stage3

Performance changed as data grows

	Indoor		Outdoor	
	Stage1	**Stage3**	**Stage1**	**Stage3**
Data amount	60168	118810	14549	122218
Mean squared error	6.234	4.224	5.942	5.051

Table 2.3 Prediction performance change under linear regression model

Noise filtering: Data pre-processing

Since there are is a lot of noise in the dataset, is necessary to apply an algorithm to filter the noise.

For example, when distance is 1m indoor, the mean RSS is around -30dBm, all the data far from -30dBm could be consider as a noise data, maybe is better to drop it.

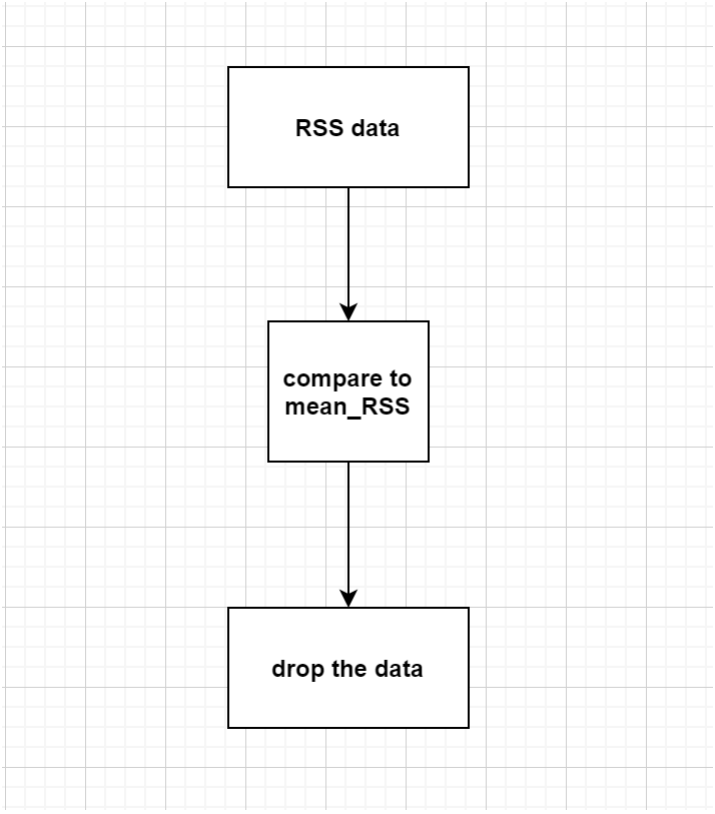


Figure 2.4 the basic idea of noise filtering

	Before	After
Mean Squared Error	6.234	5.460

Table 2.5 linear regression performance comparison before/after noise filtering

Error report

I have calculated the mean squared error, explained variance score and coefficient of determination(R2) to calculate the performance of my model. And I rounded the output data (for example, if predict result is 9.6m, it will be considered as 10m) to calculate the accuracy, it is around 15% of the accuracy. In another word, the model can improve the accuracy by 50% than guess randomly.

Overall, the performance of my model is not good. I think the reason is that the input data is too simple – only RSS. If there are a lot of features in the dataset (such as weather, temperature, etc.), the prediction will be more accurate.

Performance	Indoor	Outdoor
Accuracy (rounded the result)	0.157	0.147
Mean Squared Error	4.224	5.051
Explained Variance Score	0.124	0.384
Coefficient of determination(R2)	0.124	0.384

Table 2.6 LinearRegression model performance

My experience

Which is good

- I have learned how to reference others knowledge to improve my work.
- I have learned a lot of machine learning algorithms!
- I really enjoy doing this project, it is so practical: collect the data, analysing the data, test a new approach, evaluation…

Which need to be improved

- The understanding of the algorithms: I think I just know how to implement it but not use it well. I believe all the algorithms could be improved by adjust the parameter or change some inputs.
- My reference style may have some error, too much reference from Wikipedia, which is not reliable.

Bibliography

Bengio, Y. and Grandvalet, Y., 2004. No unbiased estimator of the variance of k-fold cross-validation. *Journal of machine learning research*, *5*(Sep), pp.1089-1105.

Güvenc, İ., 2003. Enhancements to RSS based indoor tracking systems using Kalman filters (Doctoral dissertation, University of New Mexico).

Wikipedia. (2020). Decision tree. [online] Available at: https://en.wikipedia.org/wiki/Decision_tree#:~:text=A%20decision%20tree%20is%20a%20flowchart%2Dlike%20structure%20in%20which.

Wikipedia Contributors (2019). k-nearest neighbors algorithm. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm.

Perez, A., Larranaga, P. and Inza, I., 2006. Supervised classification with conditional Gaussian networks: Increasing the structure complexity from naive Bayes. International Journal of Approximate Reasoning, 43(1), pp.1-25.

Wikipedia Contributors (2019). Linear discriminant analysis. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Linear_discriminant_analysis.

Wikipedia Contributors (2019). Logistic regression. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Logistic_regression.

Wikipedia Contributors (2018). Linear regression. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Linear_regression.

Wikipedia Contributors (2019). Multilayer perceptron. [online] Wikipedia. Available at: https://en.wikipedia.org/wiki/Multilayer_perceptron.

Scikit-learn.org. 2020. Sklearn.Svm.SVC — Scikit-Learn 0.23.2 Documentation. [online] Available at: <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html> [Accessed 9 August 2020].