RSSI ANALYSIS FOR INDOOR PROPAGATION IN WIRELESS 5G

Report submitted to the SASTRA Deemed to be University as the Requirement for the course

BECCEC707: MINI PROJECT- II

Submitted by

Malavika Venkatanarayanan (Reg no: 121004151)

December 2020

School of Electrical and Electronics Engineering

SASTRA DEEMED TO BE UNIVERSITY

(A University established under section 3 of the UGC Act, 1956)

THIRUMALAISAMUDRAM

THANJAVUR-613 401

School of Electrical and Electronics Engineering

THANJAVUR-613 401

Bonafide Certificate

This is to certify that the mini project titled "**RSSI Analysis for Indoor Propagation Model in Wireless 5G**" submitted as a requirement for the course **BECCEC 707: MINI PROJECT- II** for B.Tech. ECE programme is a Bonafide record of the work done by **Miss. Malavika Venkatanarayanan (Reg. No:121004151**) during the academic year 2020-2021, in the school of Electrical and Electronics Engineering, under my supervision.

Signature of Project Supervisor :

Name with Affiliation : **Dr. Ramya Vijay/AP-III**

Date : **24.12.2020**

Project *Vivavoce* held on 31.12.2020

EXAMINER-I **EXAMINER-II**

School of Electrical and Electronics Engineering

THANJAVUR -613 401

DECLARATION

I declare that the thesis titled "**RSSI Analysis for Indoor Propagation Model in Wireless 5G**" submitted by me is an original work done by me under the guidance of **Dr. Ramya Vijay, Assistant Professor III, School of Electrical and Electronics Engineering,** SASTRA Deemed to be University during the final semester of the academic year 2020-21, in the **School of Electrical and Electronics Engineering**. The work is original and wherever I have used materials from other sources, I have given due credit and cited them in the text of the thesis. This thesis has not formed the basis for the award of any degree, diploma, associate-ship, fellowship or other similar title to any candidate of any University.

Signature of the candidate(s) : **Malavika Venkatanarayanan**

Name of the candidate(s) : **Malavika Venkatanarayanan**

Date : **24.12.2020**

ACKNOWLEDGEMENTS

First and foremost, I thank The God Almighty for giving me the potential, health and endurance to work on this project . I express my gratitude to **Dr.S.VAIDHYASUBRAMANIAM,** Vice- Chancellor, SASTRA Deemed to be University, for encouraging me by proving all the necessary arrangements for the course of study. I also thank **Dr. R. CHANDRAMOULI**, Registrar, SASTRA Deemed to be University, for providing me the opportunity to work in the SEEE Department, SASTRA Deemed to be University, for the course of the work.

I render my solicit thanks to **Dr. K. THENMOZHI**, Dean, SEEE and **Dr. K. SRIDHAR**, Associate Dean, ECE, SASTRA Deemed to be University, for introducing the mini project in our curriculum. I owe a debt of earnest gratitude towards our guide **Dr. RAMYA VIJAY**, ECE for her continued support and guidance throughout the course of project. The intriguing questions raised by my panel members **Dr. R.K. PRASATH** and **Dr.M.CHANDRASEKAR** from the department of ECE helped me in shaping this project to its worth. I also extend my sincere thanks to my parents for their continuous moral support and friends without whose thought- provoking suggestions and encouragement, along with my guide, this work would not have been what it is today.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

English symbols (in alphabetical order)

ANN	Artificial Neural Networks
Dbm	Decibel-milliwatts
5G	5th Generation
3GPP	3rd Generation Partnership Program
LOS	Line of Sight
RSSI	Received Signal Strength Indicator
MHz	mega Hertz
mw	milli watt
PDP	Power Delay Profile
V	volts
WSN	Wireless sensor networks

ABSTRACT

KEYWORDS: *RSSI, Angle of Arrival, Linear Regression, ANN, Wireless 5G*

Being the next generation wireless network, 5G has proved to be super-fast and dense as expected. However, implementing such a network poses a problem. Even though it is superfast, the 'millimetre band' waves that are used for wireless 5G do not necessarily travel for long distances. The power at the receiver mainly depends, as we all know on the Distance, Attenuation and the angles. RSSI basically says how well a particular radio can hear the signal. In this project we attempt to use linear regression and Artificial Neural Network to find out the relation between RSSI and distance and also prove a correlation between RSSI and the AOA. The proposed idea in this paper can be applied to various indoor environments to collect more data which will help us glean insightful ideas about the propagation of radio waves. The goal is to model the dependency of the distance R on the RSSI, mathematics can serve as a descriptive tool.

Specific learning

- Log of the Distance and RSSI are linearly related
- Angle has a non-linear relation to the received power.

CHAPTER 1
INTRODUCTION

1.1 OBJECTIVE

Objective 1:

Observe the effects of channel parameters Using NYUSIM and collect data.

Objective 2:

Data analysis in Python to determine how the input data labels are related

Objective 3:

Model using Multiple Linear Regression and attempt to predict the distance

Objective 4:

Model using ANN to find out the distance

1.2 Received Signal Strength Indicator (RSSI)

By textbook definition, "RSSI is an estimated measure of power level that an RF client device is receiving from an access point or router". In simpler words, It is the power of the signal received at the receiver end. It is usually expressed in Dbm. When the distance increases the power of the signal weakens and so the overall throughput decreases. It is also relative to the quality of the received signal. Here we use Dbm instead of Db in reference to milliwatt (mW). The basic relation between the distance and RSSI is given by equation [1]. The relation between RSSI and power is given below.

$$RSSI=10\log_{10}(P/1mW) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots[1]$$

The very famous Friss Transmission [2] helps us to calculate the Receiver power. This formula states that the Received power degrades with the square of the distance. Friss formula was based in free space for omnidirectional antenna but then, The same rule applies to normal conditions too. The larger the distance, the weaker the signal becomes. The signal depends on various attenuation, loss, noise fading also. Here, we use the signal power to predict the distance using localisation algorithms. We see that in a lin-log scale, the plot between received power and distance will be linear

$$\frac{P_r}{P_t} = G_t \cdot G_r \cdot \left(\frac{\lambda}{4 \cdot \pi \cdot R}\right)^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots..[2]$$

Pr	Received Power
Pt	Transmitted power
Gt	Gain at the receiver
Gr	Gain of transmitting antenna
λ	wavelength
R	Distance between the transmitter and receiver

Table 1.1 Friss Transmission formula parameters

CHAPTER 2

2.1 NYUSIM

NYUSIM is a millimetre wave channel simulator. It has the option of human blockage and spatial consistency and various other parameters. The entire model is based on time clusters and spatial lobes. It includes 28 input parameters that is split into Channel parameters and Antenna properties. The default frequency of 28GHz is used to generate data. The environment parameter is set to NLOS and LOS for analysis purposes. The default value of pressure, temperature and humidity are used here. Using this simulator, we change the parameters and analyse the data. We also get the dataset required for linear regression.

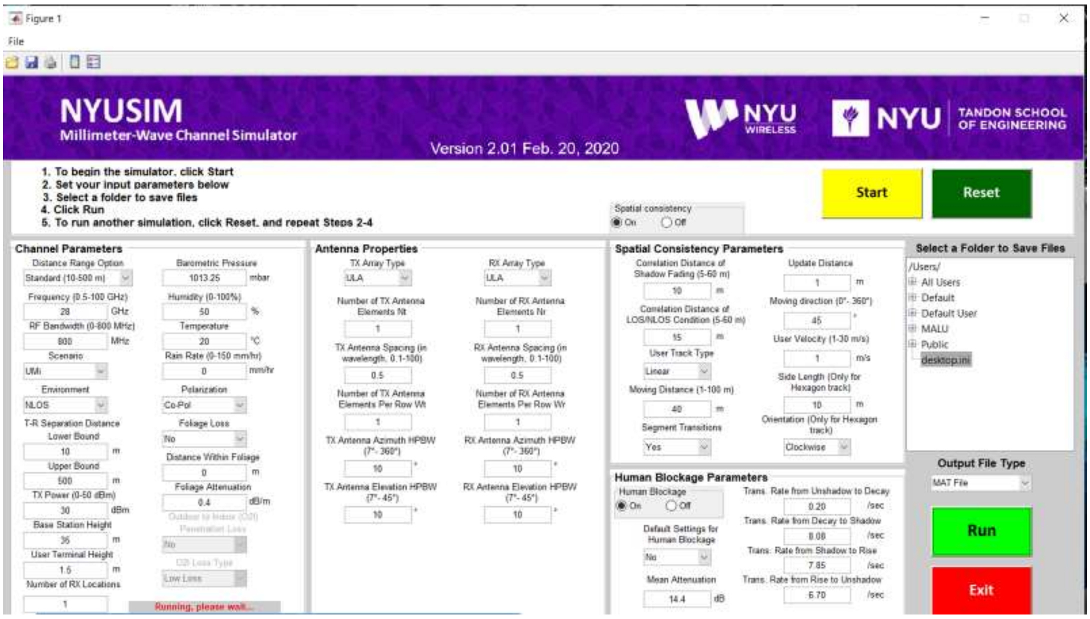

Figure 2.1. NYUSIM GUI

2.2. Simulation

The posture of a person holding a mobile is random and is not static. So it can be considered as a stochastic process since users can change the way they hold the device frequently . The change in antenna inclination angle or the angle of arrival of the signal will results in varied received signal intensity. Thus, SSCM was introduced to accommodate this change in Wireless 5G. NYUSIM uses Statistical Spatial Channel Model (SSCM) developed from extensive field measurement. It also makes use of time clustering algorithm. Using this novel simulator, we obtain a few following plots that visually describe the data. With the spatial consistency and the human blockage parameters off we get the following plots. Fig 2.2 shows the AOA power spectrum obtained during one of the simulations. This shows us that there is indeed a relation between the both. Similarly, Fig2.3 shows the directional PDP

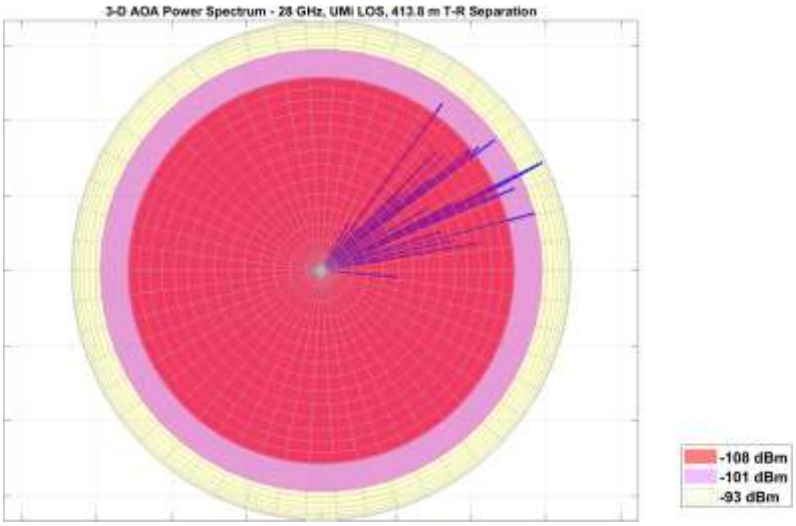

Figure 2.2. 3-D AOA power spectrum plot

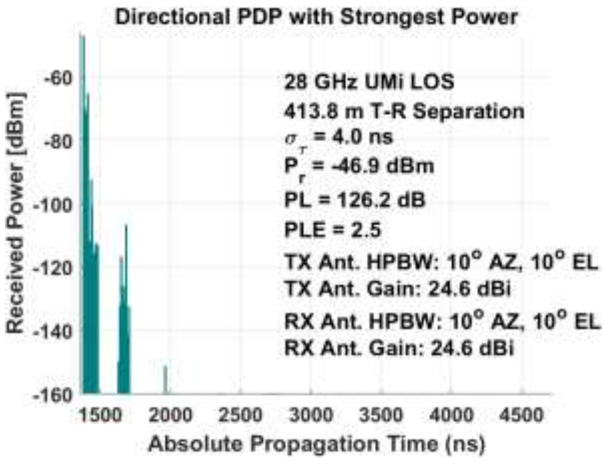

Figure2.3. Directional Power Delay Profile

When the human blockage and spatial consistency is turned ON, we get a range of values. We have used the data derived from this simulation in our model. Figures 2.4 and 3.5 show the PDP plots of LOS and NLOS.

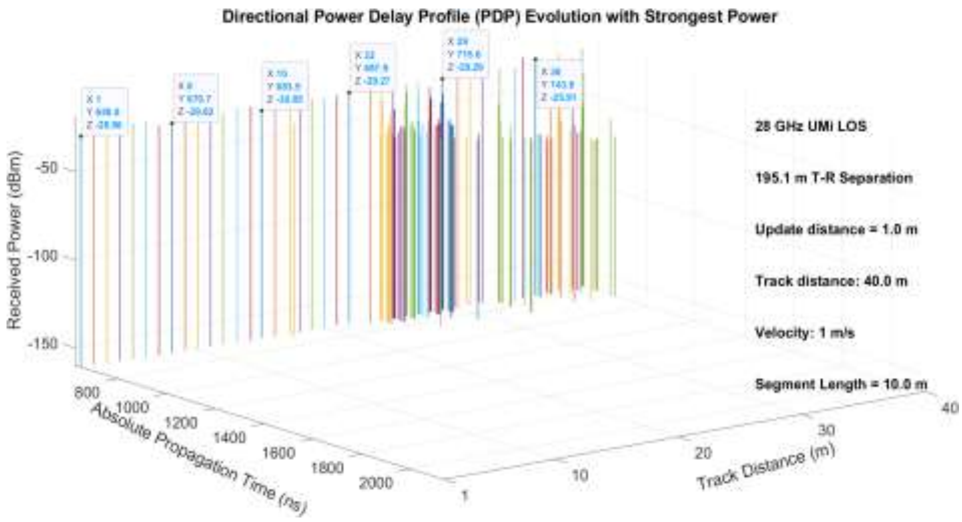

Figure2.4 PDP of the signal under the condition of LOS

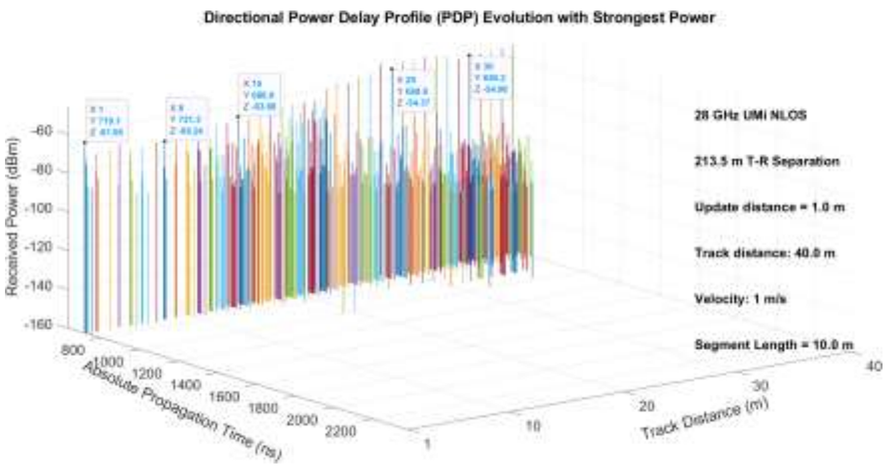

Figure2.5 PDP of the signal under the condition of NLOS

From the plots, we see that the intensity of power on the receiver side can change with different inclination angles at the receiving point. We see how the RSSI of the signal varies with distance, AOA, Human blockage. These variations can be analysed using ray tracing method to glean more insightful information.

CHAPTER 3

Regression Analysis and Neural Network

3.1. Linear Regression

Linear regression is a machine learning paradigm that is used to model the relationship between two variables. It is done by fitting the data into a linear equation. One or more variables is taken as the explanatory variable or the independent variable and the other that is supposed to be predicted is a dependent variable. When more than one explanatory variable is available it becomes a multiple linear regression. After writing a code in Python we find out a relation between Distance and RSSI given by [3]. Here, X denotes the distance.

$$\text{Log } X=b_0 +b_1*(\text{RSSI})\text{...[3]}$$

It is quite clear that the above equation is comparable to that of a straight line ($y=mx+c$) in lin-log scale and that the distance is the dependant variable here. The basic equation of multiple linear regression is given by [4].

$$Y=b_0+b_1x_1 +b_2 x_2+\text{......}+b_p x_p+e\text{..[4]}$$

Where b_0 Denotes the y-intercept; b_1 and b_2 denote the regression corefficients ;b_p is the slope coefficient and e is the model error. Now we try to model the dependancy of RSSI on distance , log X and angles to create a similar equation. Thus, creating a hyper plane.

3.2 Dataset

Here, we use the data we got whilst running the simulation in NYUSIM. The dataset dereived from the simulation had 25 values . Initially only 18 values were used to code a simple linear regression model. This can be checked by scatter plots and such. The input to this model contained only RSSI and distance. The output of ths particular fitting can be seen in Fig 3.1.

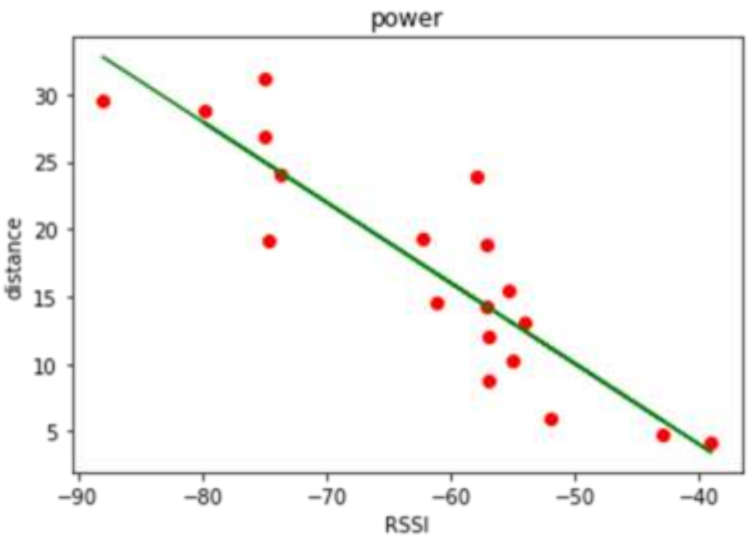

Figure 3.1 Distance Vs RSSI for 18 values

Here, we see that there is almost a liner relationship between distance and power. However, we need to increase the size of the dataset to get a more reliable fitting. The final simulation yielded 25 data samples labeled as Distance,RSSI, Angles and Log X.

Here, the AOA is denoted as Angles and Log X denotes the logarithmic value of the distance.

3.3 Data Analysis

Data Analysis involves playing around the data using the pandas and NumPy library to find out the correlation between the samples, if any. It helps in getting the useful information about the dataset. After collecting, cleaning and analysing the data, we understand more about the correlation relationship between the different labels. The results of data analysis are given by the figures below. The first assumption whislt doing mutiple linear regression is to assume that there is a linear relationship between the samples. This can be checked by scatter plots and such.

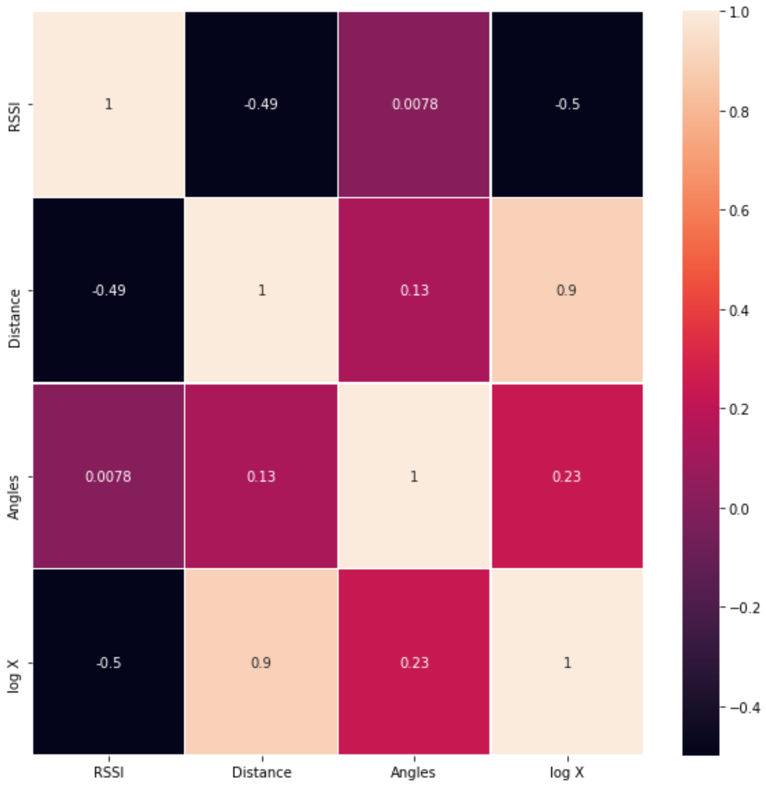

Figure 3.2 Heatmap of the dataset

The heatmap shows the visual representation of correlation between four different samples. The correlation analysis between different samples tells us how they are associated with each other and is a very important aspect of data analysis. The darker the colour, they better they are related. It can be seen that RSSI is better related to the distance and the logarithm of the distance. It is clear that the RSSI is also related to the AOA. Fig3.3 also provides the correlation table between the samples along with the statistical details of the dataset.

```
                RSSI   Distance      Angles      log X
count    25.000000  25.000000   25.000000  25.000000
mean    -55.898400  17.424000   48.160000   1.144616
std      16.012311   9.522398   37.576234   0.350679
min     -88.000000   1.000000    9.000000   0.000000
25%     -62.300000  10.200000   19.000000   1.008600
50%     -56.900000  15.400000   38.000000   1.187521
75%     -43.000000  24.000000   60.000000   1.380211
max     -28.080000  36.000000  134.000000   1.556303
              RSSI  Distance    Angles     log X
RSSI      1.000000 -0.493693  0.007845 -0.499736
Distance -0.493693  1.000000  0.131236  0.900005
Angles    0.007845  0.131236  1.000000  0.234294
log X    -0.499736  0.900005  0.234294  1.000000

Out[90]: RSSI        0.007845
         Distance    0.131236
         log X       0.234294
         Angles      1.000000
         Name: Angles, dtype: float64
```

Figure 3.3 Correlation and statistical data

We can clearly see that there is a postive correlation betwwen distance and angle which suggests that with an increase or a decrease in one quantity, the other quantity increases or decreases respectively. The negative correlation between distance and RSSI implies that when one quantity increases, the other decreases as expected. This suggests that there is a linear relationship between distance and RSSI and a non linear one between Distance and Angles. Fig3.4 clearly shows the dependance of RSSI on angles and it is clear that it is a non linear relationship

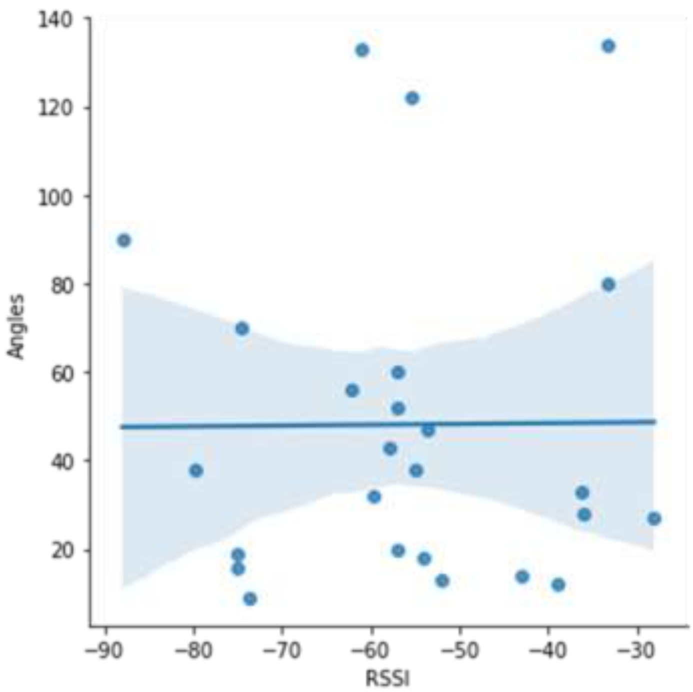

Figure 3.4 Seaborn line plot of Angles and RSSI

3.4 Fitting and Prediction

The data is divided into attributes and labels. Attributes are the independent variables and labels are dependent variables whose values are to be predicted. We have four columns in out dataset and since we have to predict the distance, it is considered as a label.

The data is split into training and testing using Scikit-Learn's built-in train_test_split(). Here, 80% of the data is split into training data and 20% is testing data. After this we train the dataset by calling in the fit() method. In multivariable linear regression, the regression model has to find the optimal coefficients for the attributes. Fig 3.5 shows the regression coefficients obtained for the following dataset using multiple linear regression analysis when Log X is being predicted. While predicting the Distance we get a set of different regression coefficients given by Fig 3.6. It is clear that both distance and Log X have the highest correlation as is indicated by their high value of regression coefficient.

	Coefficient
RSSI	-0.000894
Angles	0.000932
Distance	0.027037

	Coefficient
RSSI	0.011883
Angles	-0.028783
log X	34.059900

Figure 3.5 Coefficients of the labels when Log X is predicted (*left*) and when distance is prediction (*right*)

```
Mean Absolute Error: 3.07802480379714
Mean Squared Error: 15.822196096717075
Root Mean Squared Error: 3.9777124200621987
```

Figure 3.6 Mean error values

Prediction is the output you get after training the dataset. The output is the target variable based on the attributes. In this project we are predicting both the Distance and the Logarithmic of the distance. The predicted values are displayed in the figures 3.7 and 3.8 below. The coefficient of determination, intercept and slope values are also displayed. The coefficient of determination helps in predicting 'the goodness of fit'. This is the key value of the analysis. We have obtained a coefficient of determination of 0.82 for prediction of Log X and 0.81 for the prediction of distance. This says that 82% and 81% of the values are predictable and is a very good value. Fig 3.6 displays the RMSE and MSE values. The RMSE value is a heuristic indication of the distance between the actual and predicted values. We get a value of 3.9777 indicating a good fit.

```
intercept 0.4442958287236599
slope [-0.00186651  0.03099296  0.00116204]
[ 1.18826132e-02 -2.87830688e-02  3.40598997e+01]
coefficient of determination: 0.8291971181127156
      Actual  Predicted
0   1.082785   0.985941
1   0.612784   0.658105
2   1.113943   0.969099
3   0.778151   0.742419
4   0.681241   0.689591
5   1.187521   1.166761
6   1.008600   0.907240
7   0.939519   0.862364
8   1.274158   1.156782
9   1.164353   1.165388
10  1.155336   1.063795
11  1.285557   1.223818
12  1.281033   1.257033
13  1.378398   1.343253
14  1.428135   1.436974
15  1.380211   1.336147
16  1.459392   1.530185
17  1.494155   1.569857
18  1.469822   1.627425
19  0.000000   0.575226
20  0.903090   0.798061
21  1.176091   1.126965
22  1.342423   1.209928
23  1.462398   1.498154
24  1.556303   1.714890
```

Figure 3.7 Prediction of Log X

```
intercept -10.970230745032666
slope [-2.70722049e-02 -1.98264851e-02  2.43188818e+01]
[ 1.18826132e-02 -2.87830688e-02  3.40598997e+01]
coefficient of determination: 0.8182380988359264
    Actual  Predicted
0     12.1  15.871330
1      4.1   4.749886
2     13.0  17.227356
3      6.0   9.103548
4      4.8   6.483328
5     15.4  16.989914
6     10.2  14.293362
7      8.7  12.856883
8     18.8  21.165157
9     14.6  16.362718
10    14.3  17.482484
11    19.3  20.869401
12    19.1  20.817508
13    23.9  23.268514
14    26.8  25.414123
15    24.0  24.411746
16    28.8  25.930225
17    31.2  27.079130
18    29.5  25.372168
19     1.0 -10.547795
20     8.0  11.316294
21    15.0  15.874395
22    22.0  21.900860
23    29.0  23.908227
24    36.0  27.399238
```

Figure 3.8 Prediction of Distance

3.5 Artificial Neural Network

Artificial Neural Network (ANN) is inspired by the way the Human biological nervous system particularly the CNS and the way it works. It is composed of large number of highly interconnected processing elements that are also called neurons. The network of neurons attempts to solve a particular problem. A single layer neural network is called a Perceptron. It gives a single output. The main parameters to be decided for a neural network include the number of neurons, hidden layers, activation functions, learning rate and weight adjustment strategies.

Since ANN is a powerful supervised machine learning algorithm, we used a different dataset that consisted of 1000 training and 1000 testing data taken form the internet. In our model we have used three dense layers and one output layer as shown in Fig 3.9. Linear activation function was used along with Adam optimizer and Mean absolute error as loss function. After running 200 epochs we got a model accuracy of 57% as shown in Fig 3.10.

```
Model: "sequential_3"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_11 (Dense)             (None, 128)               384
_________________________________________________________________
dense_12 (Dense)             (None, 256)               33024
_________________________________________________________________
dense_13 (Dense)             (None, 256)               65792
_________________________________________________________________
dense_14 (Dense)             (None, 256)               65792
_________________________________________________________________
dense_15 (Dense)             (None, 1)                 257
=================================================================
Total params: 165,249
Trainable params: 165,249
Non-trainable params: 0
```

Figure 3.9 Table showing the no. of parameters

```
trics are: loss,acc,val_loss,val_acc,lr
7500/7500 - 1s - loss: 0.9860 - acc: 0.5625 - val_loss: 1.0388 - val_acc: 0.5540
Epoch 8/200
WARNING:tensorflow:Reduce LR on plateau conditioned on metric `val_accuracy` which is not available. Available me
trics are: loss,acc,val_loss,val_acc,lr
7500/7500 - 1s - loss: 0.9814 - acc: 0.5627 - val_loss: 1.0260 - val_acc: 0.5644
Epoch 9/200
WARNING:tensorflow:Reduce LR on plateau conditioned on metric `val_accuracy` which is not available. Available me
trics are: loss,acc,val_loss,val_acc,lr
7500/7500 - 1s - loss: 0.9772 - acc: 0.5636 - val_loss: 1.0164 - val_acc: 0.5680
Epoch 10/200
WARNING:tensorflow:Reduce LR on plateau conditioned on metric `val_accuracy` which is not available. Available me
trics are: loss,acc,val_loss,val_acc,lr
Restoring model weights from the end of the best epoch.
7500/7500 - 3s - loss: 0.9776 - acc: 0.5641 - val_loss: 1.0245 - val_acc: 0.5632
Epoch 00010: early stopping
10000/10000 [==============================] - 2s 173us/sample - loss: 0.9776 - acc: 0.5707
```

Figure 3.10 Model accuracy

After training the dataset, we move on to testing the model with a test dataset. After fitting , the distance values are predicted based on the RSSI values and the resulting plot is given in Fig 3.11. It can be seen that the actual and predicted data almost coincide save for a few outliers. So even if our accuracy is not so high, our model was still able to predict properly. For a better prediction, we need a proper dataset.

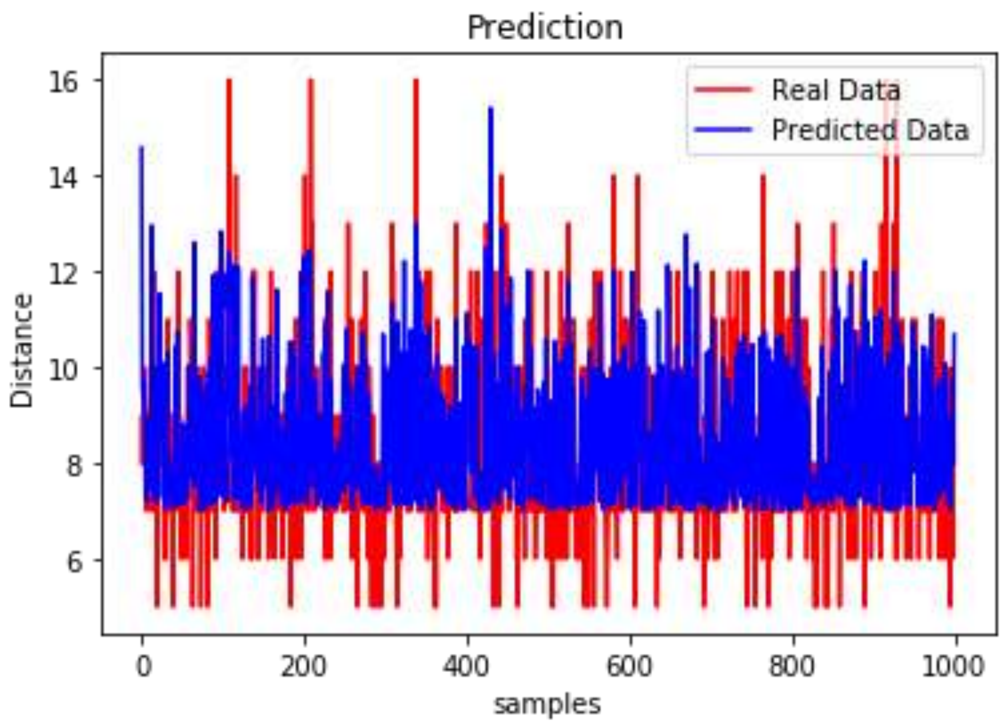

Figure 3.11 Plot of actual and predicted value using ANN

4.1. Code

The programming is done in python using various machine learning libraries like pandas, NumPy, TensorFlow, matplotlib, seaborn, etc.,

```
In [17]:
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Embedding, LSTM, GRU
from sklearn.linear_model import LinearRegression
train = pd.read_csv(r'C:\Users\Desktop\miniproject\RSSI.csv')
test = pd.read_csv(r'C:\Users\Downloads\test.csv')
sns.lmplot(x='RSSI',y='Angles',data=train)
plt.figure(figsize=(10,6))
sns.heatmap(train.corr(), annot=True, linewidth=0.2)
fig=plt.gcf()
fig.set_size_inches(10,10)
plt.show()
print(train.describe())
print(train.corr())
train.corr()['Angles'].sort_values()
lr = LinearRegression()
a = np.array([-56.9,-39,-54.1,-52,-43,-55.4,-55,-56.9,-57.1,-61.1,-57.1,-62.3,-74.7,-58,-75,-73.7,-79.9,-75,-88,]).r
b = np.array([12.1,4.1,13,6,4.8,15.4,10.2,8.7,18.8,14.6,14.3,19.3,19.1,23.9,26.8,24,28.8,31.2,29.5])
train=pd.read_csv(r'C:\Users\Desktop\miniproject\RSSI.csv')
#x = train.iloc[:,:-1].values
#y = train.iloc[:,3].values
x = train[['RSSI', 'Angles','log X']]
y = train['Distance']
model = LinearRegression()
model.fit(x, y)
model = LinearRegression().fit(x, y)
print('intercept', model.intercept_)
print('slope',model.coef_)
print(regressor.coef_)
r_sq = model.score(x, y)
print('coefficient of determination:', r_sq)
y_pred = model.predict(x)
lr.fit(x,y)
```

```
df = pd.DataFrame({'Actual': y, 'Predicted': y_pred})
print(df)
plt.scatter(a,b, color = "red")
plt.plot(a, lr.predict(a), color = "green")
plt.title("power")
plt.xlabel("RSSI")
plt.ylabel("distance")
plt.show



#train.plot(x='RSSI', y='log X')
#plt.title('Angles vs distance')
#plt.xlabel('power')
#plt.ylabel('Angles')
#plt.show()
```

Fig 4.1 and 4.2 Code snippets of Linear Regression

CHAPTER 5

RESULT and CONCLUSIONS

5.1 RESULT

1. Curve fitting done in linear regression shows the relationship of distance and received power and predicts the value with an accuracy of 81%
2. We see that the ANN predicts the output and that it almost coincides with the actual value. The accuracy of prediction would be much better with a better dataset.
3. There is a correlation between the AOA and RSSI
4. The stronger the signal strength the more reliable the connections and higher speeds are possible.

5.2 Conclusion

Hence, we trained a model using supervised machine learning algorithm to predict the value successfully. The application of this project includes object localisations in military and other sectors. For example, the location of a mobile phone could be easily found if the distance and the AOA value is known. Thus, it is useful in predicting the geolocation of cell phones. This serves as an important tool for the defence and police force.

CHAPTER 6

6.1. REFERENCES

1. Mathematical Modelling in Object Localization Based on Signal Strength by VAN HECKE Tanja, Ghent University, Faculty of Engineering Sciences, Belgium
2. Pandey, S. and Agrawal, P. “A Survey on Localization Techniques for Wireless Networks”,Journal of the Chinese Institute of Engineers 29(7), 2007, p. 1125-1148
3. ‘A Novel Millimeter-Wave Channel Simulator and Applications for 5G Wireless Communications’, Shu Sun, George R. MacCartney Jr., and Theodore S. Rappaport, 2017 IEEE International Conference on Communications (ICC), doi:10.1109/icc.2017.7996792
4. https://stackabuse.com/linear-regression-in-python-with-scikit-learn/
5. http://gmacc.me/slides/ICC2017_Slides_NYUSIM_vFinal.pdf
6. T. S. Rappaport et al., “Statistical channel impulse response models for factory and open plan building radio communicate system design,” IEEE Transactions on Communications, vol. 39, no. 5, pp. 794–807,May 1991.
7. https://stackabuse.com/linear-regression-in-python-with-scikit-learn/
8. Katsuyuki Haneda, Lei Tian et al, “Indoor 5G 3GPP-like Channel Models for Office and Shopping Mall Environments”, IEEE ICC2016-Workshops: W09-Workshop on 5G RAN Design
9. https://en.wikipedia.org/wiki/Angle_of_arrival

P3_1

ORIGINALITY REPORT

8%	6%	2%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	stackabuse.com Internet Source	2%
2	Gang Liu, Ming Zhang, Yaming Bo. "A Statistical Channel Model for Stochastic Antenna Inclination Angles", International Journal of Antennas and Propagation, 2019 Publication	2%
3	hdl.handle.net Internet Source	1%
4	msatechnosoft.in Internet Source	1%
5	towardsdatascience.com Internet Source	1%
6	open.library.ubc.ca Internet Source	1%
7	www.ijcaonline.org Internet Source	<1%
8	conference.pixel-online.net Internet Source	<1%

9	biblio.ugent.be Internet Source	<1 %
10	gmacc.me Internet Source	<1 %

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		