Predicting antenna behavior for curved surface obstacles using Machine Learning

BTP Report

by

Varshini Rama S20190020245

Kavitha Menti S20190020230

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY SRICITY

20/12/2022

Final Report

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY SRICITY

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the BTP entitled "**Predicting antenna behavior for curved surface obstacles using ML**" in the partial fulfillment of the requirements for the award of the degree of B. Tech and submitted in the Indian Institute of Information Technology SriCity, is an authentic record of my own work carried out during the time period from January 2022 to December 2022 under the supervision of Dr.Divyabramham Kandimalla, Indian Institute of Information Technology SriCity, India.

The matter presented in this report has not been submitted by me for the award of any other degree of this or any other institute.

Signature of the student with date

R Varshini(20/12/2022)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Signature of BTP Supervisor with date

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY
SRICITY

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the BTP entitled "**Predicting antenna behavior for curved surface obstacles using ML**" in the partial fulfillment of the requirements for the award of the degree of B. Tech and submitted in the Indian Institute of Information Technology SriCity, is an authentic record of my own work carried out during the time period from January 2022 to December 2022 under the supervision of Dr.Divyabramham Kandimalla, Indian Institute of Information Technology SriCity, India.

The matter presented in this report has not been submitted by me for the award of any other degree of this or any other institute.

Signature of the student with date

M Kavitha(20/12/2022)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Signature of BTP Supervisor with date

ABSTRACT

We review machine learning and its applications in a wide range of electromagnetic problems, including radar, communication, imaging and sensing. We extensively discuss some recent progress in development and use of intelligent algorithms for antenna design, synthesis, and characterization. Machine learning techniques are currently taking a major part of the ongoing research, and expected to be the key player in today's technologies. The motive of the study is to find a model which better suits our problem statement which is the application of machine learning in antenna design. It covers the major aspects of machine learning, including its basic concept, differentiation with artificial intelligence and deep learning, learning algorithms, its wide applications in various technologies, with a main focus on its usage in antenna design. The review also includes a comparison of the results using machine learning in antenna design, compared to the conventional design methods.

CONTENTS

INTRODUCTION

In current generation technology, the requirement of faster communication is becoming more and more relevant. With 5G using millimeter technology, its power decays pretty quickly. Although we may install many cell towers to tackle this situation, it's best to optimize the antennas or the surrounding environment for producing the best results. So, we intend to study the pattern of antenna characteristics under the influence of different objects placed at various positions.

The range of 5G connectivity is not great as the frequency waves are only able to travel a short distance. Added to this setback is the fact that 5G frequency is interrupted by physical obstructions such as trees, towers, walls and buildings. The obtrusions will block, disrupt or absorb the high-frequency signals. To counter this setback, the telecom industry is extending existing cell towers to increase the broadcast distance.

We intend to employ Machine-Learning techniques to determine/predict the antenna characteristics that vary. The characteristics **Maximum gain**, **Directivity of that maximum gain** and the **Pattern of reflection** caused due to various obstacle scenarios are desired to be predicted. Instead of simulating various situations each time, we can build a Machine-Learning model to predict the desired characteristics once we train the model with sufficient data set.

LITERATURE SURVEY

[1] MACHINE LEARNING IN ELECTROMAGNETICS: A REVIEW AND SOME PERSPECTIVES FOR FUTURE RESEARCH: This research paper is a discussion on machine learning and its applications in a wide range of electromagnetic problems, including radar, communication, imaging and sensing. In the past few years, there has been growing interest in machine learning (ML), more specifically artificial neural networks (ANN) and one part of its broader family - deep learning (DL). To date, ML and DL have been demonstrated to be effective in image classification, speech processing, and other information processing tasks. Very recently, ML and DL have been further extended to complex electromagnetic problems, such as structural design and optimal parameter extraction for antennas, beamforming algorithms for adaptive antenna arrays, and data interpretation for radar and MIMO systems, provides many such insights on how ML techniques can be employed in the field of antennas.

[2] MACHINE LEARNING IN ANTENNA DESIGN: AN OVERVIEW ON MACHINE LEARNING CONCEPT AND ALGORITHMS: This paper introduces and investigates the applications of machine learning in antenna design. It covers the major aspects of machine learning, including its basic concept, differentiation with artificial intelligence and deep learning, learning algorithms, its wide applications in various technologies, with a main focus on its usage in antenna design. The review also includes a comparison of the results using machine learning in antenna design, compared to the conventional design methods. With the growth and wide variety of available data, advanced processing, and affordable data storage, machine learning is witnessing great attention in finding optimized solutions in various fields. Machine learning techniques are currently taking a major part of the ongoing research, and expected to be the key player in today's technologies.

METHODOLOGY

Workflow

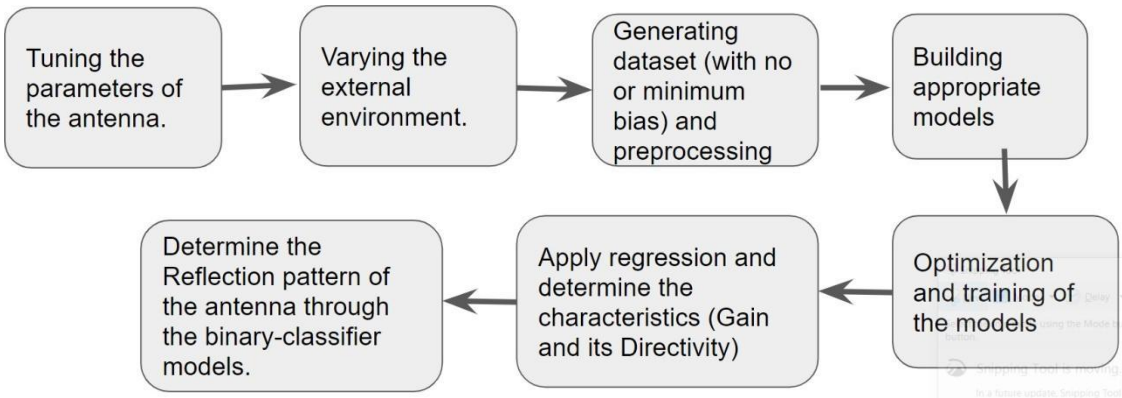

Figure 3.1: Workflow

Dataset Generation

We intend to study the variation of characteristics in a half wave dipole antenna. Since there is no readily available dataset for our problem, we have simulated the required dataset using FEKO simulation software. We input the parameters of the dipole antenna and observe the pattern of radiation with respect to various object position scenarios. We have chosen the object/obstacle to be a 3-Dimensional nurbs surface of a fully conducting material (Perfect Electric Conductor, **PEC**) and proceeded with our study on observing the pattern of reflection of the antenna.

3.2. DATASET GENERATION

We tune in the following parameters:

- Frequency of operation (GHz)
- Length of the dipole antenna (m)
- Antenna impedance (Ω)
- Radius of the wire (mm)
- Radius of curvature (m)
- Height of the curved surface (m)
- Angle (degree)
- Relative Permittivity
- Distance between the curved surface and the antenna (m)

Once we tune in the parameters of the antenna, we then define the **Nurbs surface dimensions** and **position** with respect to the Antenna.

Varying these parameters appropriately, we simulate various scenarios to determine the **Maximum gain**, the **Directivity of the maximum gain** and the **Pattern of reflection** from the polar plots obtained.

Based on the type of material used for the object, there will be slight variation in the pattern of radiation.

A **Perfect Electric Conductor** fully reflects back the EM wave from the point of con- tact either in a unidirectional or multidirectional fashion.

For each simulation, We observe the pattern of radiation and note down the maximum gain and direction of the maximum gain and pattern of reflection from the polar plot of the far-field.

3.2. DATASET GENERATION

Sample scenarios for each of the 2 types of material considered and pattern of reflection exhibited :

1. Perfect Electric Conductor material and Uni-directional reflection observed:

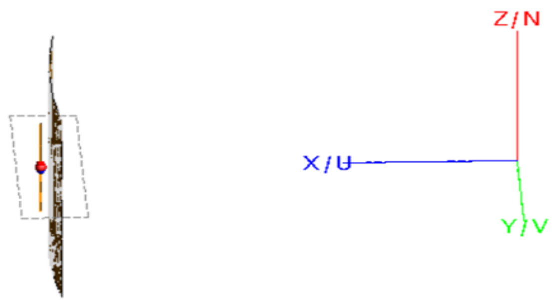

Figure 3.2: Sample simulation for a PEC obstacle, where Uni-directional pattern is observed.

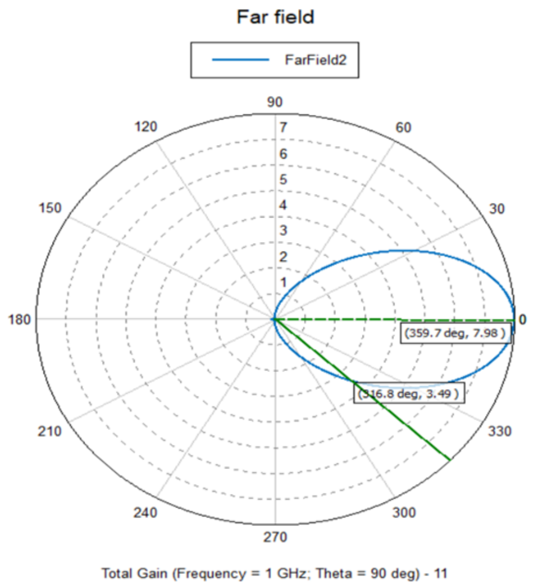

Figure 3.3: Polar plot showing the Gain and its directivity for a Uni-directional reflection pattern caused by a PEC obstacle.

Observations: Maximum gain of 7.98 is observed at an angle of 359 degrees and Uni-directional pattern of reflection is noted to occur.

2. Perfect Electric Conductor material and Multi-directional reflection observed:

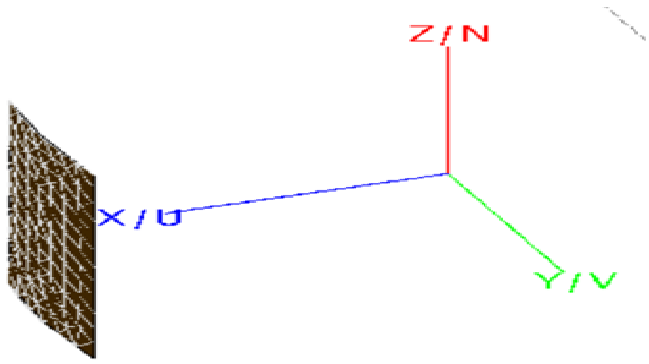

Figure 3.4: Sample simulation for a PEC obstacle, where Multi-directional pattern is observed.

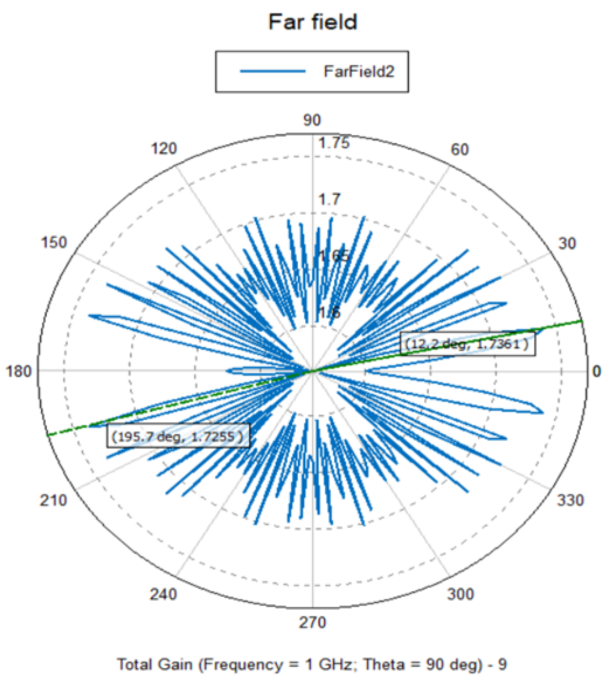

Figure 3.5: Polar plot showing the Gain and its directivity for a Multi-directional reflection pattern caused by a PEC obstacle.

Observations: Maximum gain of 1.74 is observed at an angle of 348 degrees and

Multi-directional pattern of reflection is noted to occur.

3.2. DATASET GENERATION

We have performed such simulations making sure that we have covered most of the scenarios, hence generating minimum or unbiased dataset.

We have generated 362 data samples.

	A	B	C	D	E	F	G	H	I	J	K	L
1	Frequency(GHz) (f	Dipole_length (L	Impedance (z	Wire Radius(mm) (rho	Radius of curvature (m) (a	Curved surface height(m) (h	Angle(theta)	Distance(m) (d	Relative_Permittivit	Max_gain	Direction_max_gain	Reflection_Pattern
2	1	0.0749	50	1	0.2997	0.1498	15	0.2997	infinite	2.763	310	Multi-Directional
3	1	0.0749	50	1	0.2997	0.1498	15	0.5995	infinite	2.119	332	Multi-Directional
4	1	0.0749	50	1	0.2997	0.1498	15	0.8993	infinite	1.943	338.10	Multi-Directional
5	1	0.0749	50	1	0.2997	0.1498	15	1.1991	infinite	1.865	342	Multi-Directional
6	1	0.0749	50	1	0.2997	0.1498	15	1.4989	infinite	1.819	344	Multi-Directional
7	1	0.0749	50	1	0.2997	0.1498	15	1.7987	infinite	1.789	346	Multi-Directional
8	1	0.0749	50	1	0.2997	0.1498	15	2.0985	infinite	1.764	346	Multi-Directional
9	1	0.0749	50	1	0.2997	0.1498	15	2.3983	infinite	1.753	348	Multi-Directional
10	1	0.0749	50	1	0.2997	0.1498	15	2.6981	infinite	1.74	348	Multi-Directional
11	1	0.0749	50	1	0.2997	0.1498	15	2.9979	infinite	1.727	350	Multi-Directional
12	1	0.0749	50	1	0.5995	0.1498	15	0.2997	infinite	7.98	359.9	Uni-Directional

Figure 3.6: Snippet of our Dataset

The columns A-I are the features/attributes:

- Frequency of Operation(GHz)
- Dipole length(m)
- Impedance of the circuit(Ω)
- Wire Radius(mm)
- Radius of Curvature(m)
- Height of Curved Surface(m)
- Angle
- Distance(m)
- Relative Permittivity

The columns K, L and M are our target variables:

- Max_gain
- Direction_max_gain
- Reflection_Pattern

3.2. DATASET GENERATION

Since the **Maximum gain, the direction of the maximum gain and the reflection pattern** are to be determined as independent target variables, we build Machine-Learning models for each of them.

Note:

- Frequency of operation values is taken to be 1 GHz.
- Impedance of the circuit belongs to the range of $50 \pm 5\%\Omega$.
- The distance(d) between the half wave dipole antenna and the obstacle is taken to be much greater than the operating wavelength(λ).
- The relative permittivity of a Perfect electric conductor is Infinite.

Taking all of these into account, a dataset of 362 samples has been generated.

Data Pre-processing

1. **Getting Features and Target variables:**

All the **features**(Columns A-I from the **image 3.10**) are stored in the variable **X**. The **target variables** (Columns J, K and L from the **image 3.10**) are stored in **y_1**, **y_2** and **y_3** respectively (i.e
X holds all the features,
y_1 is the target variable 'MAX GAIN',
y_2 is the target variable 'Direction of the Maximum Gain' and y_3 is
the target variable 'Reflection Pattern')

2. **Processing of the feature 'Relative Permittivity':**

The feature 'Relative Permittivity' is valued as INFINITE for Perfect Electric Conductor material. Since we only input numbers to our Machine Learning models and to find a way to input the value of Infinity, we take the reciprocal of the feature 'Relative Permittivity' for all the Data samples.

Material	Relative_Permittivity
Conductor	Infinity
Dielectric	5
Conductor	Infinity
Dielectric	5
Dielectric	4.8

Material	Relative_Permittivity
Conductor	0.000000
Dielectric	0.200000
Conductor	0.000000
Dielectric	0.200000
Dielectric	0.208333

Figure 3.7: Reciprocal conversion

3. Data Scaling:

All the features and targets are of different ranges, so in order to make sure that all of the features contribute in similar proportions, we need to scale the data.

The scale function from scikit-learn scales each of the attributes/features and targets such that they follow normal or gaussian distribution with mean=0 and variance =1.

Frequency(GHz)		
2.4		-0.938806
2.4		-0.938806
2.4	⇨	-0.938806
5.0		1.065183
2.4		-0.938806

Figure 3.8: Scaling the data

Now we split the data into 70% for training and 30% for validation purposes.

Building ML Models

Since we intend to predict the **values** of the maximum gain and its directivity, we need to employ a **regression** technique.
Whereas for the reflection pattern, we intend to **classify** it into being **Uni-directional** or **Multi-directional** reflection, so we need to employ **binary-classification** techniques.

The data we deal with, has labels associated with each of the values. So, we employ a **supervised classification/regression** technique.

We have developed various Classification/Regression models to compare and choose the better model for our problem statement.

1. **SVM (Support Vector Machine) both for classification and regression:**

- A support vector (data points closest to the hyperplane) determine/influence the position and orientation of the hyperplane.
- A SVR/SVC(Support Vector Regression/Support Vector Classifier) performs higher dimension computations when the data can't really be separated in the original given dimensions.
- This higher dimension computations work on the principle of kernel trick.
- Based on the kernel we choose, it accepts inputs in the original lower dimensional space and returns the dot product of the transformed vectors in the higher dimensional space.

We have built SVR/SVC with the following kernels:

(a) Linear.

(b) Radial Basis Function with default parameters C and γ.

(c) Radial Basis Function with optimized parameters C and γ.

Note:

- C is a hyperparameter to control the error.It emphasizes on how lenient our model can be towards errors while building the hyperplane.
- γ is a hyperparameter that determines the curvature of the hyperplane.

2. K-nearest neighbor (KNN) both for classification and regression:

- KNN is a supervised technique based on similarity measures(distance function).
- The new input data is matched with 'k' nearest neighbor data points.
- Based on the k value chosen, the new input determines its k nearest neighbor data points and
  - averages out their values for a regression model.
  - ranks the number of data points falling under each of the classes and classifies into the class which is most frequent among those k data points.

3. Logistic Regression for binary classification:

- Primarily used for binary classification.
- Logistic regression gives the likelihood of the event occurring. Its range is bounded between 0 and 1, based on the value being higher than or lower than 0.5, it classifies into 0 or 1.

We have built each of these appropriate regression/classification models to predict y_1(Maximum gain), y_2(Direction of maximum gain) and y_3(Reflection Pattern).

RESULTS

Performance Analysis

1. **Regression Models:**

Once we develop the **regression** models to predict the target variables **y_1(Max Gain)** and **y_2(Directivity of maximum gain)**, we need to evaluate them on the test dataset.

Since we have built **regression** models, we evaluate them based on **error functions**. The error functions that we have chosen to determine the performance of our model are :

- Mean-Squared-Error (MSE): It is the summation of (square of(true value - predicted value)) and averaged to the no of samples N.

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

MSE = mean squared error
n = number of data points
Y_i = observed values
$\hat{Y}_i$ = predicted values

Note: Since we square the difference, MSE punishes more when there is a large difference between the predicted value and the true value.

4.1. Performance Analysis

- Mean-Absolute-Error (MAE): It is the summation of (modulus of(the true value - predicted value)) and averaged to the no of samples N.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

MAE = mean absolute error

y_i = prediction

x_i = true value

n = total number of data points

Note: These error functions range between 0 and infinity .The closer the error function value is to 0, the better the model fits our data.

2. **Classification models:** The performance of binary-classifier models are judged based on their accuracy.

Accuracy is determined by the **number of correct predictions /total samples on which the model is being tested**.

To determine which model performs the best, it is understood that instead of just considering a random split(70% training and 30% test), k-fold cross validation is a better way to understand which model does the best for various chunks of data samples considered.

In K-fold cross validation, based on the value of k chosen, the total data samples are partitioned into k equal parts and each group out of the k groups is taken to be a testing set and the rest of the k-1 groups are taken as training data. Each time this process occurs, the accuracy of the model is determined,noted and the model is discarded.

Once this process is repeated for all of the k groups, the average of the accuracies is considered to better understand the performance of the model.

Model Evaluation

- **Regression Models to predict the Max gain and Direction of Max gain:** Evaluating the regression models based on the mse and mae scores obtained for each of the models.

1. SVR with Linear Kernel:

   a) Prediction of Maximum Gain:

```
MSE for predicting the MAXIMUM GAIN using linear SVR is 0.32481
MAE for predicting the MAXIMUM GAIN using linear SVR is 0.44155
```

   b) Prediction of Directivity of maximum Gain:

```
MSE for predicting the DIRECTION OF MAXIMUM GAIN using linear SVR is 0.68196
MAE for predicting the DIRECTION OF MAXIMUM GAIN using linear SVR is 0.44207
```

2. SVR with RBF Kernel (default parameters):

   a) Prediction of Maximum Gain:

```
MSE for predicting the MAXIMUM GAIN using SVR with rbf kernel and default parameters is 0.19081
MAE for predicting the MAXIMUM GAIN using SVR with rbf kernel and default parameters is  0.21213
```

   b) Prediction of Directivity of maximum Gain:

```
MSE for predicting the DIRECTION OF MAXIMUM GAIN using SVR with rbf kernel and default parameters is 0.15723
MAE for predicting the DIRECTION OF MAXIMUM GAIN using SVR with rbf kernel and default parameters is 0.21496
```

3. SVR with RBF (optimized parameters):

   a) Prediction of Maximum Gain when the optimal parameters are: C=10, gamma=0.25

```
MSE for predicting the MAXIMUM GAIN using SVR with rbf kernel and optimal parameters is 0.19414
MAE for predicting the MAXIMUM GAIN using SVR with rbf kernel and optimal parameters is 0.18496
```

   b) Prediction of Directivity of maximum Gain when the optimal parameters are: C=100, gamma=0.25

```
MSE for predicting the DIRECTION_MAX_GAIN using SVR with rbf kernel and optimal parameters is 0.10052
MAE for predicting the DIRECTION_MAX_GAIN using SVR with rbf kernel and optimal parameters is 0.21743
```

4.2. MODEL EVALUATION

Note: It is observed that optimizing the parameters made the msa and mae values further closer to 0.

4. K-nearest neighbor :

   a) Prediction of Maximum Gain(better results were achieved for k=3):

```
MSE for predicting the MAXIMUM GAIN using knn when k= 3 is 0.07530
MSE for predicting the MAXIMUM GAIN using knn when k= 3 is 0.08737
```

   b) Prediction of Directivity of maximum Gain(better results were achieved for k=5):

```
MSE for predicting the DIRECTION OF MAXIMUM GAIN using knn when k= 5 is 0.11660
MSE for predicting the DIRECTION OF MAXIMUM GAIN using knn when k= 5 is 0.17737
```

Note: In K nearest neighbor algorithm, when k=3, our model is subjective to any outliers in the dataset.(But since the dataset is generated from simulations, k=3 can be allowed.)

5. Random Forest Model:

   a) Prediction of Maximum Gain:

```
MSE for predicting the MAXIMUM GAIN using Random Forest is 0.05104
MAE for predicting the MAXIMUM GAIN using Random Forest is 0.08117
```

   b) Prediction of Directivity of maximum Gain:

```
MSE for predicting the DIRECTION OF MAXIMUM GAIN using Random Forest is 0.02864
MAE for predicting the DIRECTION OF MAXIMUM GAIN using Random Forest is 0.08776
```

- **Binary-Classifiers to predict the Reflection Pattern:**

1. Logistic Regression:

   Accuracy of Logistic Regression model in predicting the 'Reflection Pattern' is observed as 86.813%

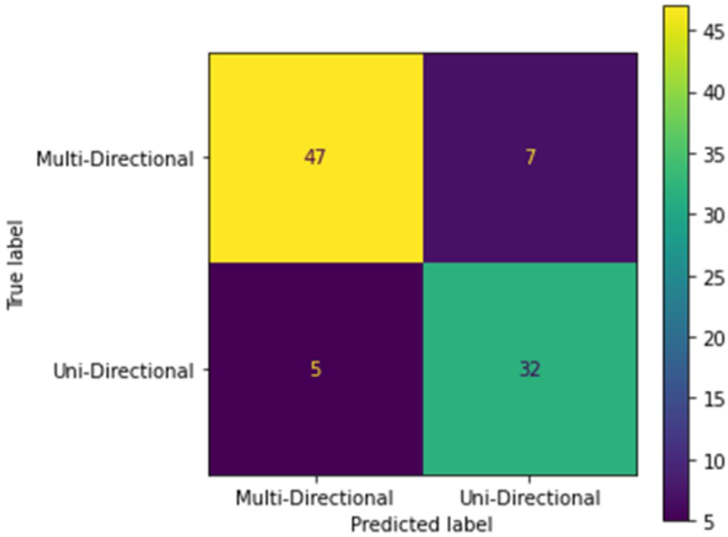

Figure 4.1: Confusion matrix for Logistic regression

2. SVC with linear kernel:

   Accuracy of SVC with linear kernel in predicting the 'Reflection Pattern' is observed as 85.714%

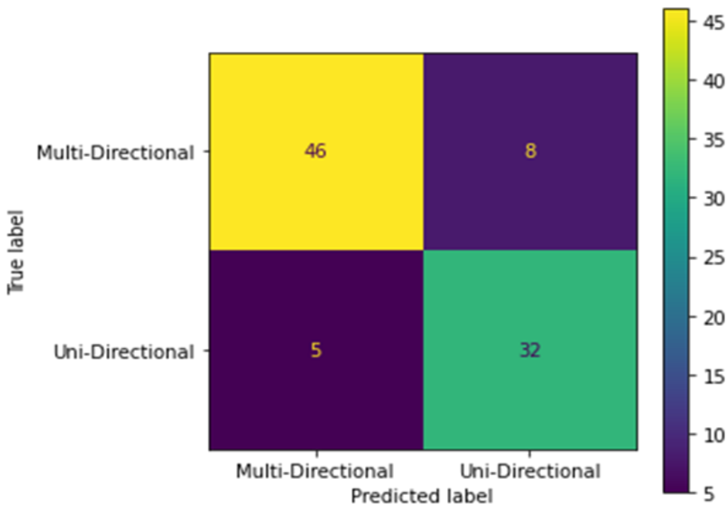

Figure 4.2: Confusion matrix for SVC with linear kernel

3. SVC with rbf kernel and default parameters:

   Accuracy of SVC with rbf kernel and default parameters in predicting the 'Re- flection Pattern' is observed as 89.011%

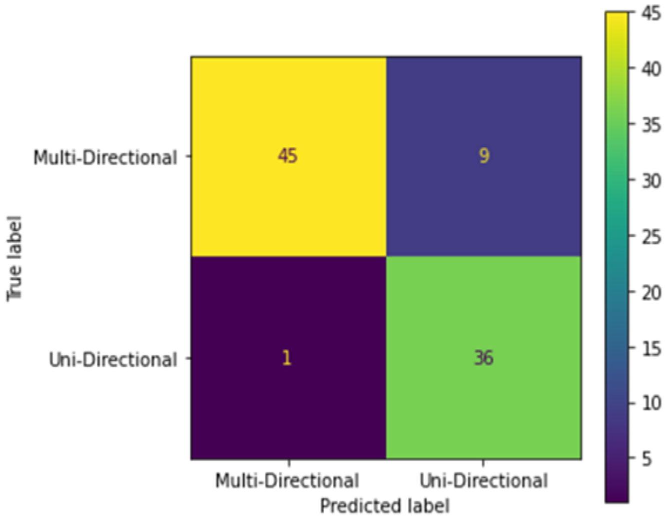

Figure 4.3: Confusion matrix for SVC with rbf kernel and default parameters

4. SVC with rbf kernel and optimized parameters:

   Accuracy of SVC with rbf kernel and optimized parameters(C=1000,γ=0.1) in predicting the 'Reflection Pattern' is observed as 95.604%

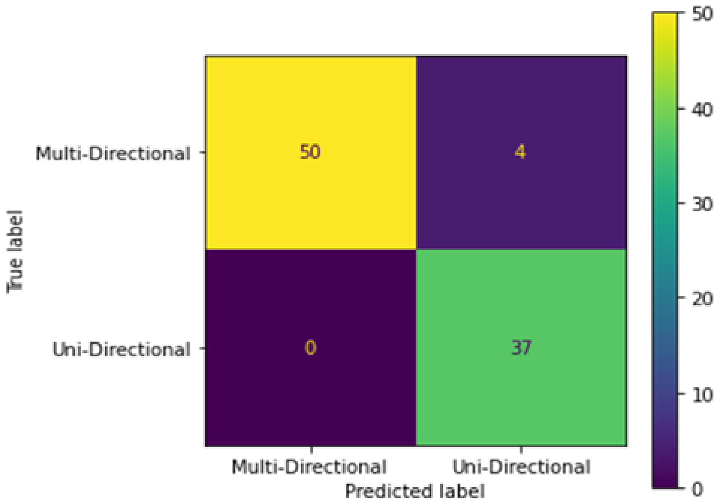

Figure 4.4: Confusion matrix for SVC with rbf kernel and optimized parameters

5. K-Nearest Neighbour Model:

   Accuracy of K-Nearest Neighbour Model for k=2 in predicting the 'Reflection Pattern' is observed as 94.505%

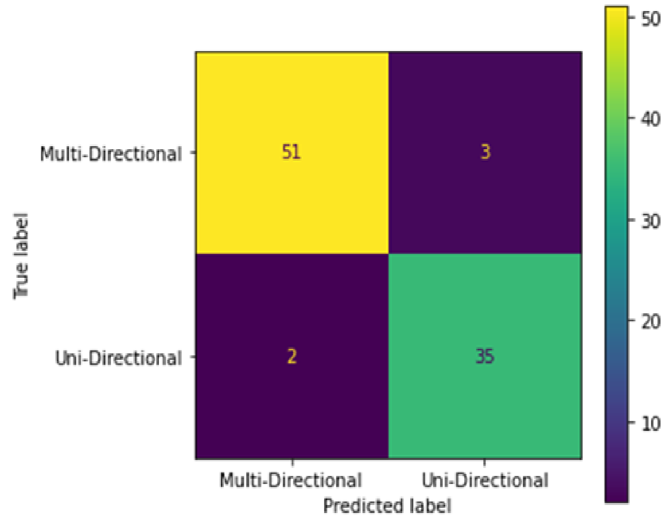

Figure 4.5: Confusion matrix for K-Nearest Neighbour Model

6. Random Forest Model:

   Accuracy of Random Forest Classifier Model in predicting the 'Reflection Pattern' is observed as 98.901%

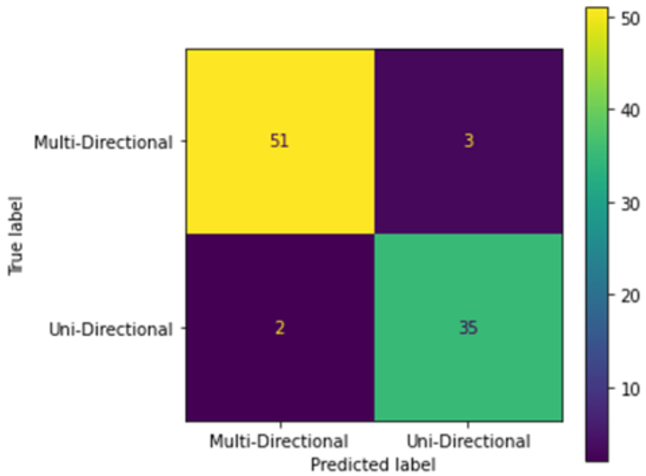

Figure 4.6: Confusion matrix for Random Forest Model

Observations and conclusion

For the regression models built to predict the ’Max gain’ and ’direction of max gain’:

Regression Model	Mean-Squared-Error (MSE)	Mean-Absolute-Error (MAE)
Linear SVR	0.324	0.441
SVR with RBF Kernel (default parameters)	0.190	0.212
SVR with RBF (optimized parameters)	0.194	0.184
K-nearest neighbor	0.075	0.087
Random Forest	0.051	0.0811

Table 4.1: Evaluation metrics for Maximum gain prediction

Regression Model	Mean-Squared-Error (MSE)	Mean-Absolute-Error (MAE)
Linear SVR	0.681	0.442
SVR with RBF Kernel (default parameters)	0.157	0.214
SVR with RBF (optimized parameters)	0.100	0.217
K-nearest neighbor	0.116	0.177
Random Forest	0.028	0.087

Table 4.2: Evaluation metrics for prediction of directivity of maximum gain

Since the metrics **MSE** and **MAE** are best observed(closer to 0) for the **Random Forest model** in prediction of **maximum gain** and for the **direction of the maximum gain**, these models best fit our data.

For the Classification models built to predict the 'Reflection Pattern':

With the help of k-fold cross validation technique, we have observed the average of accuracy value of the model for each partition of the accuracy scores observed for all the partitions.

Classifier	Accuracy	K-fold cross validation(Avg)
Logistic Regression	86.813	84.5
Linear SVM	85.714	84.1
SVC with RBF Kernel (default parameters)	89.011	90.0
SVC with RBF (optimized parameters)	95.604	88.6
K-nearest neighbor	94.505	95.6
Random Forest	98.901	96.7

Table 4.3: Evaluation metrics for Binary-Classifiers designed to predict the Reflection Pattern

It is observed that the **Random Forest Model** better fits the data to predict the Reflection Pattern

List of Figures

List of Tables

ABBREVATIONS

ML	Machine Learning
DL	Deep Learning
ANN	Artificial Neural Networks
MIMO	Multiple Input Multiple Output
PEC	Perfect Electric Conductor
SVC	Support Vector Classification
SVR	Support Vector Regression
KNN	K-nearest neighbor
MSE	Mean-Squared-Error
MAE	Mean-Absolute-Error
RBF	Radial-Basis-Function
SVC	Support Vector Classification
PEC	Perfect Electric Conductor

REFERENCES

[1] D. Erricolo et al., “Machine Learning in Electromagnetics:A Review and Some Perspectives for Future Research,”2019 International Conference on Electromagnetics in Advanced Applications(ICEAA),2019,pp.1377-1380, doi: 10.1109/ICEAA.2019.8879110

[2] H. M. El Misilmani., “Machine Learning in Antenna Design: An Overview on Machine Learning Concept and Algorithms,”2019 International Conference on High Performance Computing & Simulation (HPCS),2019,DOI: 10.1109/HPCS48598.2019.9188224

[3] Hilal M. El Misilmani., “A Review on the Design and Optimization of Antennas Using Machine Learning Algorithms and Techniques,”2020 International Journal of RF and Microwave Computer-Aided Engineering, 2020,DOI:10.1002/mmce.22356