T.R.

GEBZE TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

UNCERTAINITY QUANTIFICATION AND PERFORMANCE COMPARISON FOR VARIOUS RF CIRCUIT TOPOLOGIES

ŞABAN ŞENLİ

A THESIS SUBMITTED FOR THE DEGREE OF

MASTER OF SCIENCE

DEPERTMENT OF ELECTRONICS ENGINEERING

GEBZE

2022

T.R.

GEBZE TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

UNCERTAINITY QUANTIFICATION AND PERFORMANCE COMPARISON FOR VARIOUS RF CIRCUIT TOPOLOGIES

ŞABAN ŞENLİ

A THESIS SUBMITTED FOR THE DEGREE OF MASTER OF SCIENCE

DEPERTMENT OF ELECTRONICS ENGINEERING

THESIS SUPERVISOR

ASSIST. PROF. DR. ÖNDER ŞUVAK

GEBZE

2022

T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ÇEŞİTLİ RF DEVRE TOPOLOJİLERİ İÇİN BELİRSİZLİK HESAPLAMA VE PERFORMANS KARŞILAŞTIRMASI

ŞABAN ŞENLİ
YÜKSEK LİSANS TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

DANIŞMANI
DR. ÖĞR. ÜYESİ ÖNDER ŞUVAK

GEBZE
2022

GEBZE TEKNİK ÜNİVERSİTESİ	**YÜKSEK LİSANS JÜRİ ONAY FORMU**

GTÜ Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 23/06/2022 tarih ve 2022/31 sayılı kararıyla oluşturulan jüri tarafından 01/07/2022 tarihinde tez savunma sınavı yapılan Şaban ŞENLİ'nin tez çalışması Elektronik Mühendisliği Anabilim Dalında YÜKSEK LİSANS tezi olarak kabul edilmiştir.

JÜRİ

ÜYE (TEZ DANIŞMANI)	:Dr. Öğr. Üyesi Önder ŞUVAK
ÜYE	:Doç. Dr. Serdar Süer ERDEM
ÜYE	:Doç. Dr. Mustafa Berke YELTEN

ONAY

Gebze Teknik Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ...…/……/…… tarih ve ……./…..... sayılı kararı.

İMZA/MÜHÜR

SUMMARY

A number of simulation tools can be used in RF/Analog circuit designs. In these simulation tools like Keysight ADS, CST Design Studio, Cadence AWR etc. a number of targets are set for the desired circuit purpose. Circuits are tried to be optimized according to these goals. If one considers a filter example which is used a transmission line filter for this study the simulation tool is told which frequencies and how much to suppress for this filter, the simulation tool tries to find the closest results to the characteristics we demand by making thousands of trials. In this study, it has been shown that correct results can be obtained with machine learning methods for similar optimization processes. In this way, it will not be necessary to carry out heavy operations and evolutionary electromagnetic computations for each time for optimization. In this study, a number of machine learning techniques have been applied for a filter that can operate from 1GHz to 10GHz. Finally, it has been shown comparatively that the filter parameters required for a desired filter characteristic can be obtained as a result of this machine learning application. In order to train machine learning applications mentioned filter modeled in MATLAB and data collected for characteristic impedance(Z0) ranges from 30Ohm to 90 Ohm for each of transmission lines.

Key Words: Machine learning, RF/Microwave filter optimization, polynomial regression, Prediction Algorithms.

ÖZET

RF/Analog devre tasarımlarında bir takım simülasyon araçları kullanılmaktadır. Keysight ADS, CST Design Studio, Cadence AWR vb. bu simülasyon araçlarında istenilen devre amacı için bir takım hedefler belirlenir. Devreler bu amaçlara göre optimize edilmeye çalışılır. Bu çalışma için iletim hattı filtresi kullanılan bir filtre örneğini ele alırsak, simülasyon aracına bu filtre için hangi frekansları ne kadar bastıracağı söylenir, simülasyon aracı binlerce işlem yaparak istediğimiz özelliklere en yakın sonuçları bulmaya çalışır.Bu çalışmada, benzer optimizasyon süreçleri için makine öğrenmesi yöntemleri ile doğru sonuçların alınabileceği gösterilmiştir. Bu sayede optimizasyon için her seferinde ağır işlemler ve bir takım elektromanyetik hesaplamalar yapılmasına gerek kalmayacaktır. Bu çalışmada, 1GHz'den 10GHz'e kadar çalışabilen bir filtre için bir takım makine öğrenmesi teknikleri uygulanmış ve son olarak bunun sonucunda istenilen bir filtre özelliği için gerekli filtre parametrelerinin elde edilebileceği karşılaştırmalı olarak gösterilmiştir.Makine öğrenimi uygulamalarını eğitmek için, MATLAB'da modellenen filtrenin her bir iletim hattının karaktersitik empedansı 30 Ohm ile 90Ohm arasında birer Ohm farkla tekrar tekrar simule edilerek veri seti elde edilmiştir.

Anahtar Kelimeler: Makine öğrenmesi, filtre optimizasyonu, polinomiyal regresyon,Tahmin Algoritmaları.

ACKNOWLEDGEMENTS

It would be pleasure for me to declare lots of thanks of gratitude to my supervisor Dr. Önder ŞUVAK who strongly support me from very beginning of this study until end of the studies on the topic High Frequency Filter Optimizations with Machine Learning Techniques helping me to concentrate on very detailed research. Eventually I learned about so many interesting and beneficial things in short period of time.

Secondly, it would be pleasure for me to declare lots of thanks of to my beloved mother, my father and friends who helped me a lot in finalizing this study within very short range of time.

TABLE of CONTENTS

LIST OF ABBREVIATIONS AND ACRONYMS

Abbreviations and Acronyms		**Explanations**
W	:	Width
l	:	Length
h	:	Height
ε	:	Relative Dielectric Constant
Hz	:	Hertz
F/f	:	Frequency
V	:	Volt
γ	:	Gamma
Z_0	:	Characteristic Impedance
I	:	Current
R^2	:	R-squared
DT	:	Decision Tree
ML	:	Machine Learning
OLS	:	Ordinary Least Squares
RF	:	Random Forest
SVR	:	Support Vector Regression

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

In circuit simulation tools, especially RF/Microwave design and Keysight ADS, CST Design Studio, Cadence AWR etc. some optimization targets can be set to optimize the design in the simulation tools, then the programming functions start thousands of traces. Optimizations are very important tasks for designers, otherwise it is very difficult and time consuming for designers to achieve their goals. The working principle of optimizations is very simple, but it consumes time and energy. As soon as some goals are achieved, experiments continue, with some parameters preserved. This method is a good solution most of the time, but it creates a lot of processing overhead for the hardware. Also, this method may not work for some special and difficult electromagnetic function solutions.

In this study, an alternative way for optimization problems has been developed by developing some machine learning applications. One of the most important conditions for a machine learning application to run successfully is to train it with sufficient datasets. In this study, to create this data set, a filter circuit was simulated in MATLAB and simulated thousands of times by sweeping the Z0 (characteristic impedance) parameters for each transmission line between 30 Ohm and 90 Ohm. As a result, the necessary data set was obtained. The obtained data were preprocessed in a way that machine learning algorithms could understand and machine learning algorithms were trained. Although the machine learning algorithms that should be used are prediction algorithms, since the prediction algorithms are specialized for some purposes in themselves, many different machine learning methods have been tried and finally these algorithms have been compared.

1.1. Comparison with Previous Deep Learning Based Methods

Artificial Neural Networks have been used for some RF and Microwave modeling and simulations recently. Thus, the processor load and simulation processes required for more complex electromagnetic simulations and combinations can be reduced compared to traditional approaches. In Zhang, Gupta and Devabhaktuni's[6] studies, it is explained theoretically and practically how Artificial Neural Networks can be evaluated for RF/Microwave modeling and design. With this study, the authors claim that new circuits can be operated more easily and with wider parameter ranges than traditional approaches in design and simulations, that is, in solving complex electromagnetic simulations. On the other hand, in Watson and Gupta's studies [7], coplanar waveguide circuit elements are modeled with an ANN algorithm with the data obtained from electromagnetic simulations and the obtained models are then modeled as double stub filters and power dividers. Thus, there is no need for complex electromagnetic solutions.

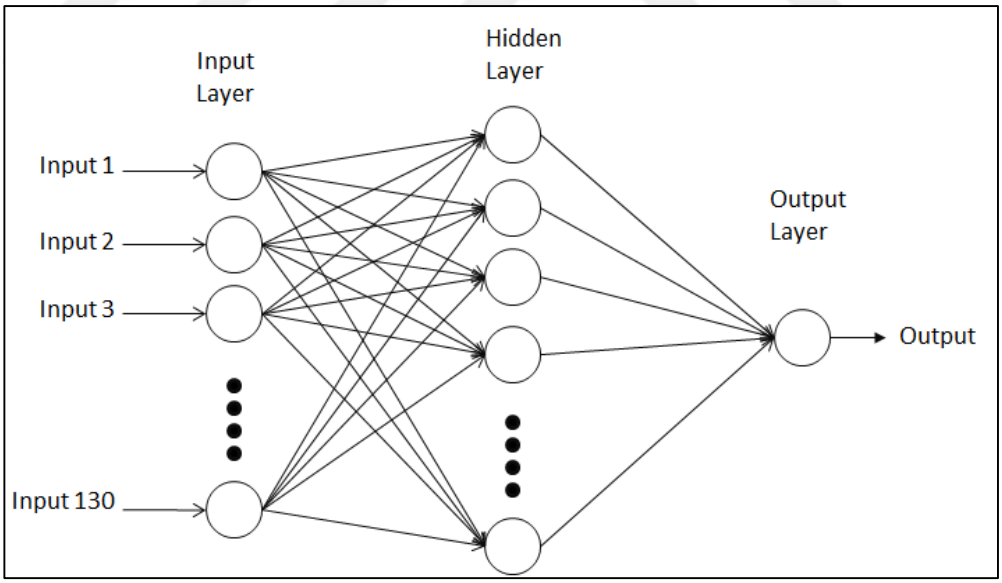


Figure 1.1: ANN algorithm structure.

However, the deep learning methods used in both studies already contain a significant amount of processing overhead during the training phase. In addition, the filter topologies that can operate successfully in a certain frequency range for RF/Microwave filters are completely changing. For example, while it is necessary to design with stacked elements up to 1GHz, it is difficult to design at frequencies above 1GHz. Now it is necessary to design with distributed components. Also Helical filters, coaxial filters and perhaps interdigital filters should be used at much higher frequencies. As a result, designers cannot use all the frequencies they want by changing the parameters of a filter. For this reason, although solutions for slightly wider frequency ranges have been produced with deep learning methods in order to design some passive/active high frequency circuit components like power dividers and double stub filter etc., design of certain components of certain frequency range results in much more accurate results as we have studied.

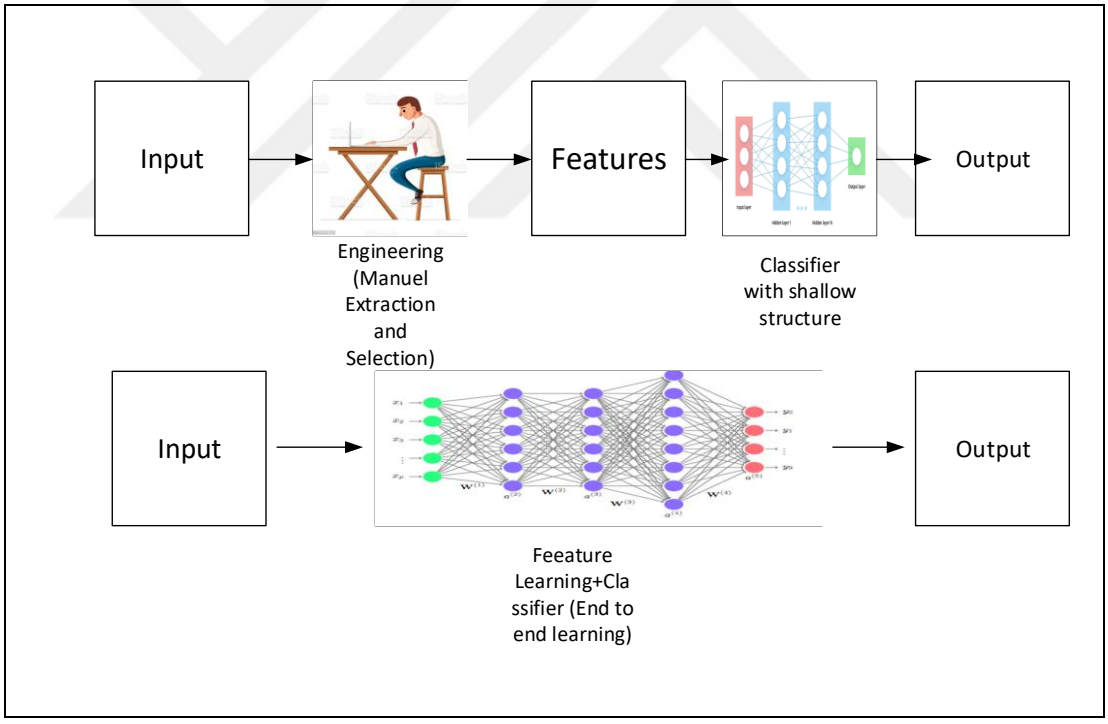


Figure 1.2: Comparison of machine learning and deep learning.

In this study, a design has been made with distributed elements that are guaranteed to work successfully in the 1 GHz-10 GHz range. It has been observed that successful results can be obtained with Prediction algorithms, which is one of the classical machine learning methods, and a more application-specific and highly accurate solution has been obtained with less processing load.

2. FILTER MODELLING AND OVERVIEW OF THE APPROACH

Instead of conventional lumped elements, transmission/micro-strip line or other applications that can simulate desired impedance values are used to design filters as higher frequencies are reached. At very high frequencies, the lumped elements exhibit very high parasitic properties, making the design difficult. Note that for this purpose 'Microwave and RF Design of Wireless Systems' book which is written by Pozar is followed [1].

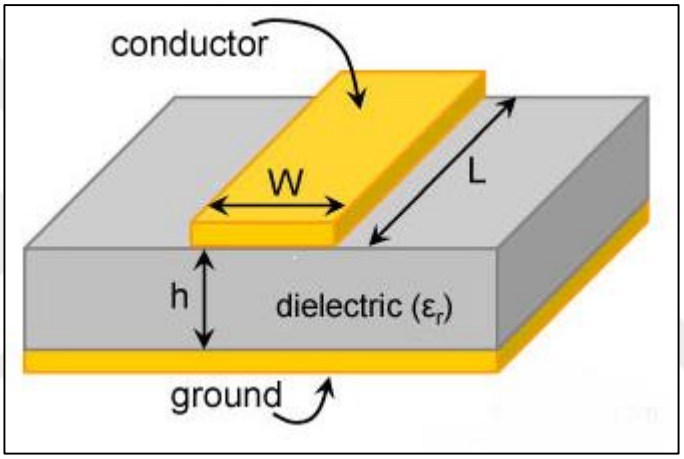


Figure 2.1: Sample transmission line.

In this study, the design made using the transmission line as seen in Figure 2.1. In order to simulate these circuits in the MATLAB environment, some microwave functions must be applied. For example, two transmission lines can be connected in parallel or in series, or grounded or open-ended. Some formulas are used to simulate this circuit. The prepared dataset is an s-parameter file. The following are the command functions most critically used to achieve S-parameter results.

- getZ0
- abcd2s
- txlineMicrostrip

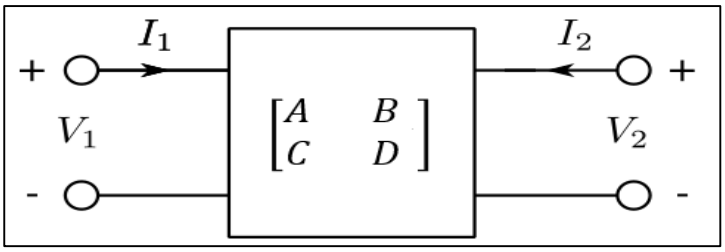


Figure 2.2: ABCD matrix for a system.

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \tag{2.1}$$

ABCD parameter values of a system are calculated as in Figure 2.2. Since there is more than one component in the given circuit and these components are connected in series or parallel to each other, the ABCD matrix of each component must be calculated to obtain ABCD. By multiplying these ABCD matrices, the ABCD matrix of the whole system can be obtained. [1] Finally, these ABCD parameters must be converted to the S-parameter matrix, since system-related operations cannot be calculated over the S parameters.

$$\begin{bmatrix} V_{in} \\ I_{in} \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} V_L \\ I_L \end{bmatrix} \tag{2.2}$$

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} cosh(\gamma l) & Z_0 sinh(\gamma l) \\ sinh(\gamma l)\,/Z0 & cosh(\gamma l) \end{bmatrix} \tag{2.3}$$

ABCD parameter calculation matrix of transmission line is given above. When the matrix calculations shown in equation above are calculated on serial and parallel transmission lines, the equations given below are obtained. [Appendix A]

$$Transmission\ Line = \begin{bmatrix} cosh(\gamma l) & Z_0 sinh(\gamma l) \\ sinh(\gamma l)\,/Z0 & cosh(\gamma l) \end{bmatrix} \tag{2.4}$$

$$Serial\ Impedance = \begin{bmatrix} 1 & Z \\ 0 & 1 \end{bmatrix} \tag{2.5}$$

$$Shunt\ Impedance = \begin{bmatrix} 1 & 0 \\ 1/Z & 1 \end{bmatrix} \tag{2.6}$$

The filter topology that used in this study consists of a transmission line and a ground-tipped transmission line connected in parallel, as shown in Figure 2.3.

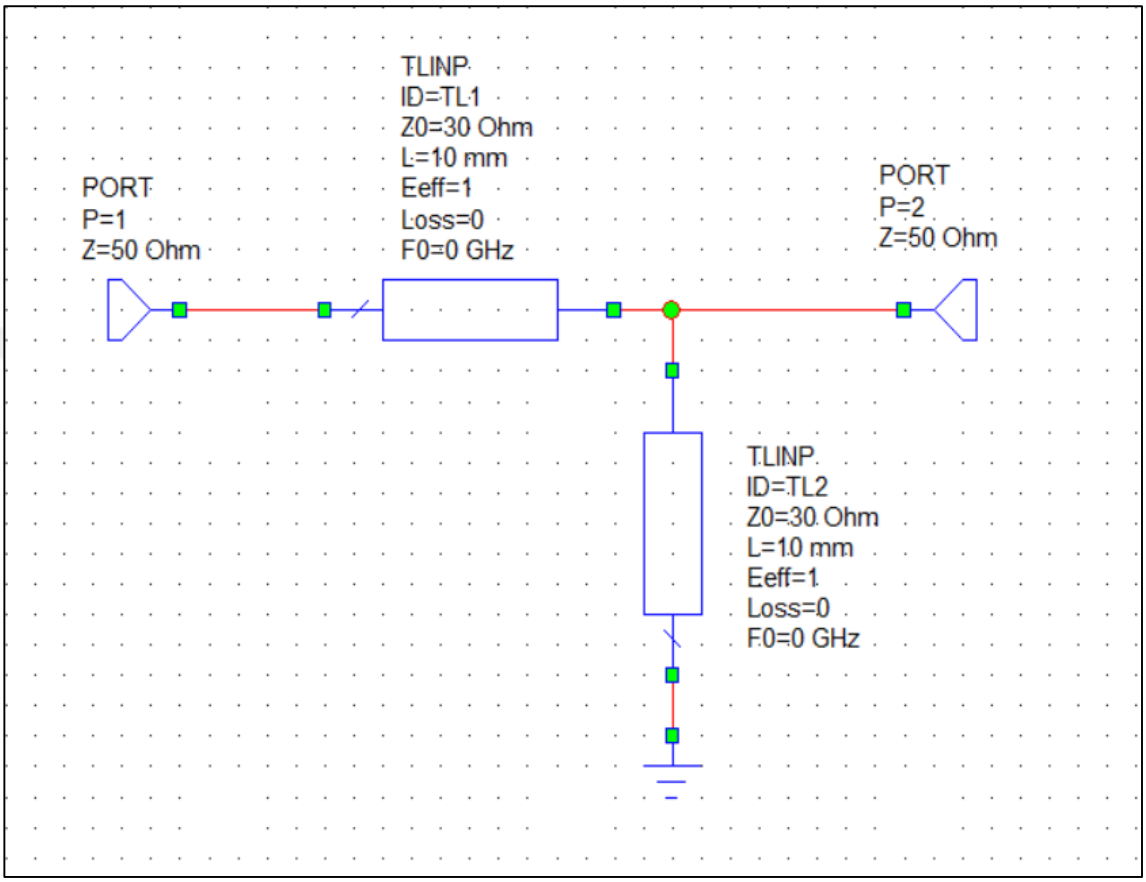


Figure 2.3: Filter topology.

The resultant ABCD matrix of these transmission lines, which can be considered as connected in a row, can be calculated as given in Figure 2.4. [2]

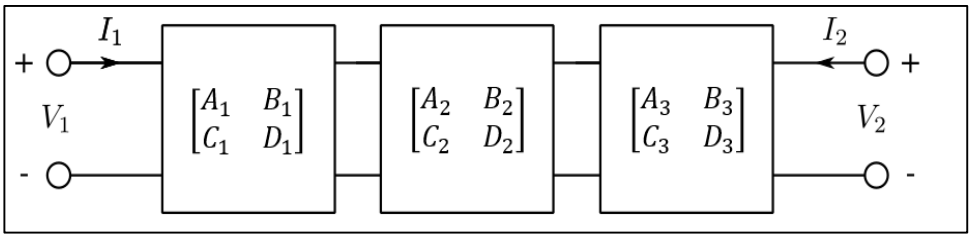


Figure 2.4: Cascaded ABCD parameters.

$$\begin{bmatrix} V_1 \\ I_1 \end{bmatrix} = \begin{bmatrix} \acute{A} & \acute{B} \\ \acute{C} & \acute{D} \end{bmatrix} \begin{bmatrix} V_2 \\ I_2 \end{bmatrix} \; where \; \begin{bmatrix} \acute{A} & \acute{B} \\ \acute{C} & \acute{D} \end{bmatrix} = \begin{bmatrix} A_1 & B_1 \\ C_1 & D_1 \end{bmatrix} \begin{bmatrix} A_2 & B_2 \\ C_2 & D_2 \end{bmatrix} \begin{bmatrix} A_3 & B_3 \\ C_3 & D_3 \end{bmatrix} \quad (2.7)$$

The next thing to do is to obtain the S-parameter matrix from the obtained ABCD parameters matrix by applying computational conversion formulas [Appendix B] and repeat this for the characteristic impedance values in a certain range and keep these data.

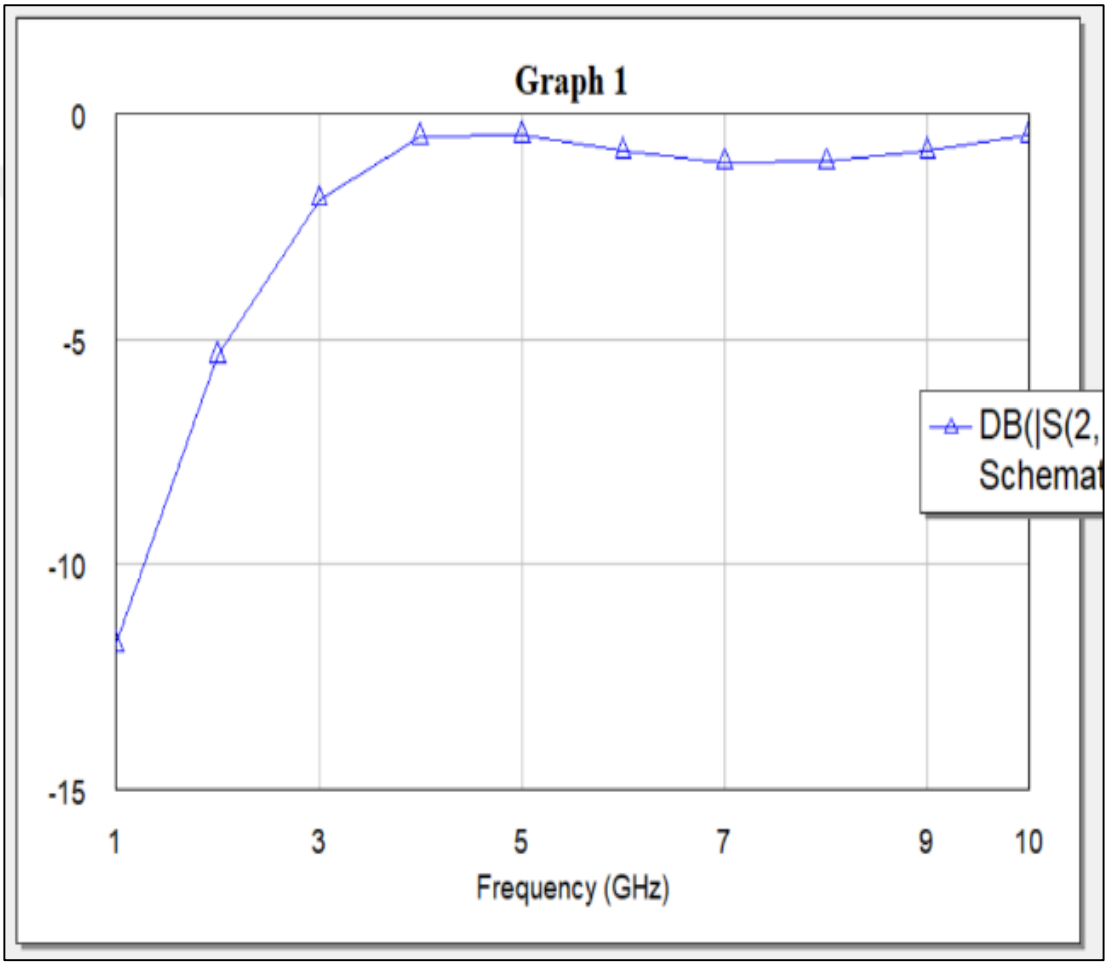


Figure 2.5: A sample of S21 graph for figure 2.3.

In this study, values between 30 Ohm and 90 Ohm were tried for each transmission line and these results were taken as samples from 1GHz to 10Ghz at 1GHz intervals. Thus,3600 of S21 values obtained.

For a system where power or energy injected from one port and measured from any other port, S-parameter calculations are more measurable and more easy to calculate than voltage and current calculations especially for RF (Radio Frequency)

networks. Therefore, S-parameter matrix indicates system characteristics most commonly for high frequencies. Each element of matrix symbolize how system behaves for different ports. Note that for this purpose ‘ Micro-strip filters for RF/Microwave Applications’ book which is written by Hong and Lancaster is utilized [2].

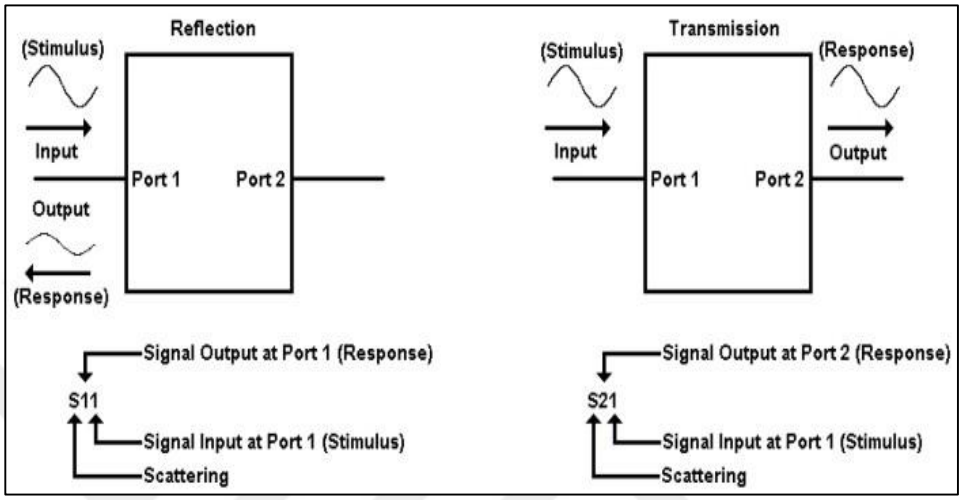


Figure 2.6: Scattering parameters for a system.

S-parameters are commonly represented in matrix form and obey the rules of matrix algebra. In this study S21 values are used to train machine learning algorithms. In the same way S11, S22 and S12 values can be achieved also. However, since S21 indicates how much of power absorbed during power pass through port 1 to port 2 S21 is most important parameter. Therefore, S21 is chosen for all calculations in this study.

$$S_{11} = \frac{b_1}{a_1}\ (where\ a2 = 0) = \frac{reflected\ power\ wave\ at\ port1}{incident\ power\ wave\ at\ port1} \tag{2.8}$$

$$S_{21} = \frac{b_2}{a_1}\ (where\ a2 = 0) = \frac{transmitted\ power\ wave\ at\ port2}{incident\ power\ wave\ at\ port1} \tag{2.9}$$

$$S_{22} = \frac{b_2}{a_2}\ (where\ a1 = 0) = \frac{reflected\ power\ wave\ at\ port2}{incident\ power\ wave\ at\ port2} \tag{2.10}$$

$$S_{12} = \frac{b_1}{a_2}\ (where\ a1 = 0) = \frac{transmitted\ power\ wave\ at\ port1}{incident\ power\ wave\ at\ port2} \tag{2.11}$$

As can be seen from above equations s-parameters are a measure of power calculations from one port to another. In this study aim of designer is to find how much of injected power absorbed and how much of injected power observed from output port. Therefore, S21 parameter used for calculations in this study.

3. MACHINE LEARNING-BASED CIRCUIT PERFORMANCE PREDICTION

Regression (Prediction) methods can be used as a machine learning technique since it is desired to finally estimate the S21 parameter values with all the data obtained. [3] However, since there is more than one regression method, the CRISP-DM model given in Figure 3.1 as a roadmap should be used. CRISP-DM is a standard processing model that is known for step-by-step data processing and model development. As a result, data scientists take action and make progress to improve the model.

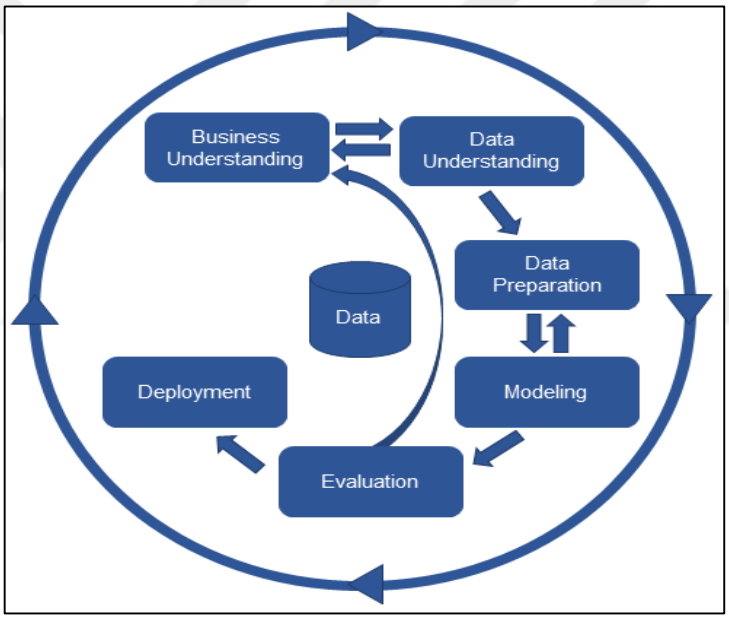


Figure 3.1: CRISP-DM flow diagram.

In this study, the first stage (Business Understanding) has been explained in detail in the previous titles. In the second step (Data Understanding), it is necessary to understand the data, the data obtained is a numerical data and is the data obtained as a result of a number of electromagnetic equations. In this respect, there is a polynomial relationship between the data rather than a linearity.

The third stage is the data preparation stage. At this stage, the data was cleaned by using the Pandas Library and turned into ready-to-use, outlier-free data frames.

In the next stages, algorithms have been developed by using various regression methods and mainly scikit-learn, numpy and matplotlib libraries. The results obtained as a result of these methods were tried to be optimized with the methods of backward elimination and forward elimination of p-values. In addition, success rates were compared in terms of R2 values and success percentages.

R-squared is parameter that measures how successfully results fit into model for regression models. R-squared measure success of model by measuring how much independent variables can explain dependent variables of output which means it shows the strength of relationship between independent and dependent variables. This parameter may have a value in the range of 0-100% scale.

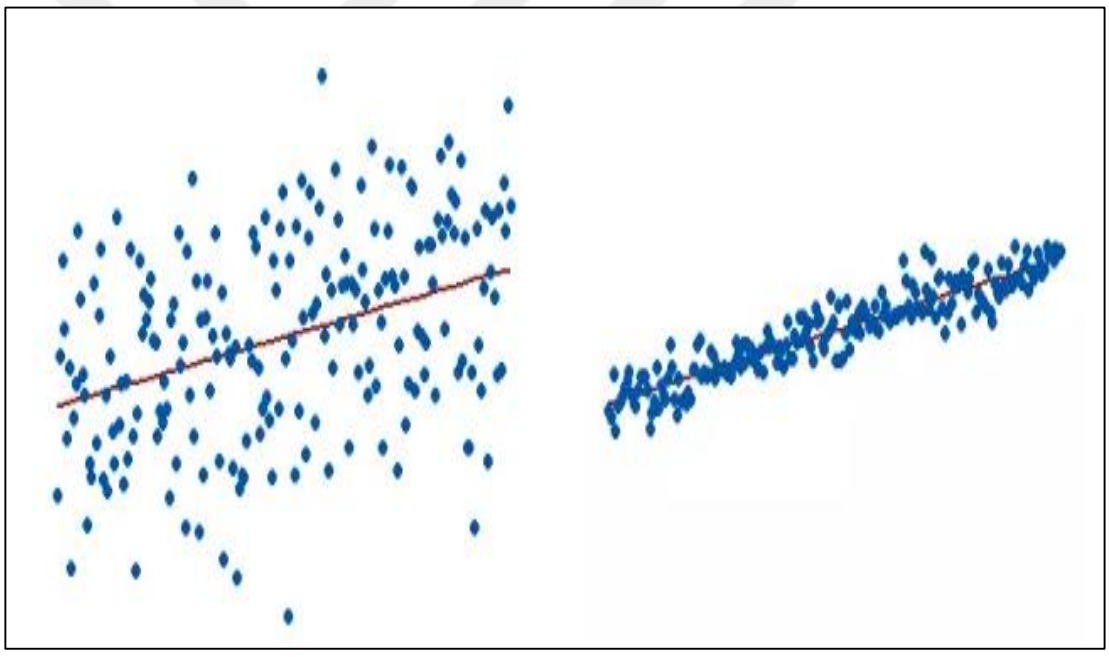


Figure 3.2: Successful and unsuccessful methods.

A high R2 indicates good regression model fit. As mentioned above, if all points were on the regression line, there would be a perfect model. Since the Sum of Squares (SSS) of the residuals will be zero when all points are on the line, R2 will also be equal to 1 and take the highest value it can get. [4]

3.1. Model and Method Selection

Model selection is core part of a machine learning studies where all different models like support vector regression, k-nearest neighbor, logistic regression, random forests, decision trees, etc. and their different range of hyper parameters (e.g. Radial Basis Function for kernel in Support Vector Machines, k value for k-nearest neighbor) are used and compared with respect to their goodness of fit and accuracy. [8]

3.1.1. Simple/Multiple/Polynomial Linear Regression Method

$$Simple\ \ Linear\ Regression\ \ y = b_0 + b_1 x_1 \tag{3.1}$$

$$Multiple\ \ Linear\ Regression\ \ y = b_0 + b_1 x_1 + b_2 x_2 \ldots \ \ \ldots + b_n x_n \tag{3.2}$$

$$Polynomial\ \ Linear\ Regression\ \ y = b_0 + b_1 b_1 x_1 + b_2 x_1^2 \ldots \ \ \ldots + b_n x_n^n \tag{3.3}$$

As can be seen from above, Simple Linear Regression is expressed with only one independent variable and one coefficient, while multiple linear regression is expressed with more than one independent variable and one constant number, and polynomial linear regression is expressed with a constant number and exponential values of more than one independent variable. In this study, it is reasonable to express the behavior of the data using the polynomial linear regression method. [8]

3.1.2. Support Vector Regression Method

The support vector regression method can be modeled as in Figure 3.3. The line drawn in for obtained data may be linear, but it is possible to draw curves using different methods. In other words, linear SVR exists as well as non-linear SVR. For this, a non-linear interval drawn when the SVR model is applied together with the Radial Basis Function (RBF) method. [5] In this respect, successful results can be expected with the Support Vector Machine method in this study.

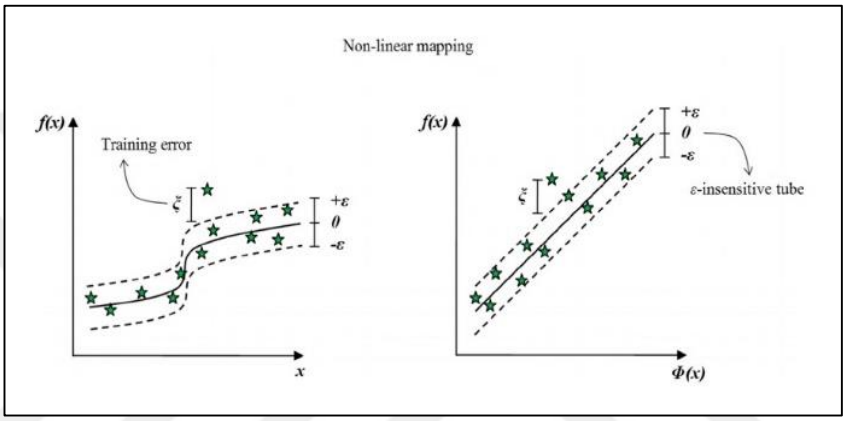


Figure 3.3: Boundary graph for support vector regression.

3.1.3. Decision Tree Method

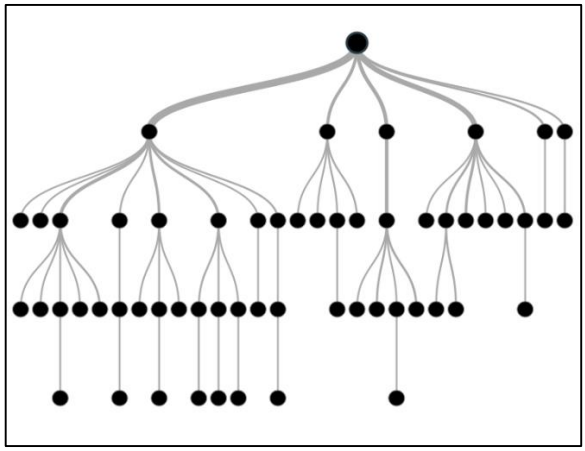


Figure 3.4: Decision tree diagram.

Decision tree as can be understood from its name it is a predictive model used in machine learning applications data mining that looks like a tree structure. In decision tree structure most outer branches or leaves represent the output values which may be called class label, inner branch represent nodes of classification of values or features. In decision tree model which is known for its simplicity of algorithms both of the prediction and classification problems can be solved. For prediction problems results should be taken as real values of certain output values. However, for classification problems results achieved as one of the class given into model which is best fit into solution. To conclude, results are either a decided class or decided approximated value.

Number of nodes that branch symbolizes are critical point since there is a trade-off between number of nodes and accuracy. If number of nodes choose very big, results will have very high accuracy percentage however there will be also too much processing load and processing time. Also there may be over-fit for given dataset which means model do not learn how to solve problem it just memorizes values. As a result, during the test of new dataset there will be very low accuracy. On the other hand, if number of the nodes choose very small, results will have very low accuracy percentage. All mentioned theoretical information provided by the textbook named as ‘Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent system’. [8]

3.1.4. Random Forest Method

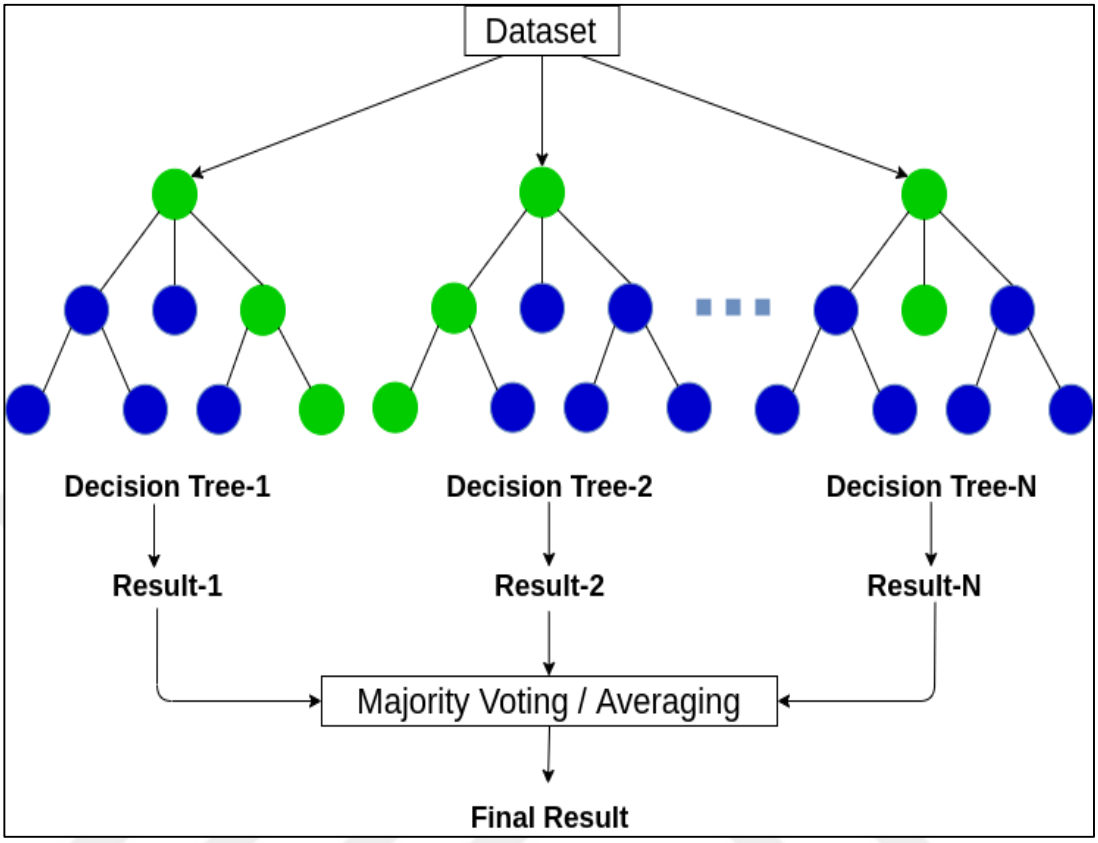


Figure 3.5: Working diagram of random forests.

Random forests are simply compound of decision trees that have different behavior for same dataset input. Random forest method can be applied to both classification and prediction problems. While for classification problems results should be obtained from vote of each decision tree and result is the one which has maximum rate. On the other hand, for prediction problems results are average of output of each decision tree. [8] For most of the time random forests are more successful than decision trees however gradient boosted trees have better accuracy. Nevertheless, accuracy generally depends on data characteristics.

The working principle of the Random Forest method for regression creates decision trees by dividing the data set into small pieces as given in Figure 3.5. When an estimation is made later, the estimations in these decision trees are averaged.

3.2. Optimization of Model

There are also some important stages after model selection and implementation in data science. One of these stages is model optimization. Optimization is a mandatory step because many of the default hyper parameters may not fit the model with the best values. Besides model selection step, the hyper parameters within the selected model for optimizations need to be adjusted according to the available data properties. Optimization results can be measured by several parameters. Since data science is a multidisciplinary science, there are some parameters and concepts taken from different sciences. One of them is the P value. The probability value, the P value, is an indicator to understand whether the null hypothesis (H0) is true. P values take different values in the range of 0 to 1. In this study, as in most machine learning studies, 0.5 was accepted as the critical point to see the importance of each different input parameter. A feature with a p-value greater than 0.5 means that this feature value does not help the model fit the solution when processing the dataset. However, input parameters with a p value less than 0.5 can have a strong effect on the output values, so these parameters should not be neglected.

There are many more models for solving problems in machine learning, but only some were used for this study. Ordinary Least Squares Regression is a regression model that can successfully solve unstructured data types for input. This model gives results in reports containing very important parameters to understand how successful the applied method is. Therefore, in this study, OLS Reports were used to compare the success rates of the previously mentioned models. The Ordinary Least Squares (OLS) regression method is in the family of Supervised Learning. This model measures the Euclidean distance between the actual output values and the estimated distance. After measuring, it compares with other trials and tries to reduce it in each trial. As a result, the model can reach the optimum point where all R-squared values are minimized. Equation in below shows how R-squared values calculated.

$$R^2 = \frac{SSR}{SST} = \frac{\sum(\hat{y}i - \bar{y})^2}{\sum(yi - \bar{y})^2} \tag{3.4}$$

$$SSR = SUM\ OF\ SQUARED\ REGRESSİON \tag{3.5}$$

$$SST = SUM\ OF\ TOTAL\ ERROR \tag{3.6}$$

$$\hat{y}i = predicted\ value\ of\ y\ dependant\ variables\ of\ i \tag{3.7}$$

$$yi = dependant\ y\ \ values\ of \tag{3.8}$$

$$\bar{y} = average\ of\ y\ dependant\ variables \tag{3.9}$$

OLS Regression method may work very well for different types of data sets included single independent variable as input and single or multi dependent variables as output. Explanations of the terms that mentioned in OLS Regression Report:

- R-squared: This value indicates percentage of how well model fit into solution.
- Adj. R-squared: This value means how importantly additional input variables affects the results.
- F-statistic : High values prefered.This value is the ratio between mean regression sum of and the mean error sum of squares.
- AIC :it measure whether model is relatively better or not with given dataset
- BIC : it compares success for set of models.coef :the coefficients of contant terms and independent variables.
- std err : standard error estimation for coefficients.
- t : it shows importance of coefficients.
- P > |t| : this equations means that nul hypotesis is correct.

The following page has an OLS report for linear regression. Some conclusions can be drawn from this report. P-value and R-square value, which are the most important parameters determining the quality of the method, are given in the report. These parameters are given and interpreted for each method in the following sections.

Table 3.1: OLS report.

Dependent Variable:	y	R-squared (uncentered):	0.921
Model:	OLS	Adj. R-squared (uncentered):	0.920
Method:	Least Squares	F-statistic:	4305.
Date:	Tue, 05 Apr 2022	Prob (F-statistic):	0.00
Time:	17:34:10	Log-Likelihood:	-15969.
Number of Observations:	3721	AIC:	3.196e+04
Df Residuals:	3711	BIC:	3.202e+04
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
x1	-12.1796	0.215	-56.664	0.000	-12.601	-11.758
x2	16.4630	0.709	23.220	0.000	15.073	17.853
x3	-5.3018	0.802	-6.614	0.000	-6.873	-3.730
x4	-6.6054	0.865	-7.638	0.000	-8.301	-4.910
x5	-4.5427	3.774	-1.204	0.229	-11.941	2.856
x6	-11.4269	4.838	-2.362	0.018	-20.912	-1.942
x7	-5.3406	7.953	-0.672	0.502	-20.933	10.252
x8	0.5613	7.978	0.070	0.944	-15.081	16.204
x9	-5.3297	4.798	-1.111	0.267	-14.736	4.077
x10	-2.0276	3.753	-0.540	0.589	-9.386	5.331

Omnibus:	17.162	Durbin-Watson:	0.391
Prob(Omnibus):	0.000	Jarque-Bera (JB):	14.357
Skew:	-0.084	Prob(JB):	0.000763
Kurtosis:	2.747	Cond. No.	323.

3.3. Stepwise Regression

With advances in the computational capabilities of processors today, step-by-step regression becomes very easy. Stepwise regression is a method for improving the success rates of models. Stepwise regression is simply the process of removing unnecessary input variables from input datasets. After each of the input values are subtracted, the processing and evaluation of the data begins again and the result is compared with the previous results. These steps should be performed several times to achieve the best and optimized results. However, deciding which input variables to eliminate is an important point. Therefore, there are different types of stepwise regression. Note that for this purpose 'Mathematics for machine learning' book which is written by Faisal and Ong is followed [5].

3.3.1. Types of Stepwise Regression

Basically, there is three types of stepwise regression that explained below:

- Forward selection: While at the beginning optimization of model there is no variable, at each step variables added and tested step by step. If added variable significantly affect model positively to optimize it, it should be kept and process should go on until sufficient variable added and model reached most optimized version.
- Backward elimination: While at the beginning optimization of model there is set of independent variable, at each step an independent variable eliminated and model tested step by step. If eliminated variable significantly affect model positively to optimize it, process should go on until all unnecessary variable eliminated and model reached most optimized version.
- Bidirectional elimination is a combination of forward selection and backward elimination method. This method may result in most successful optimized model sometimes.

In this study, backward elimination technique was applied at the end of the execution of the algorithms. Using the backward elimination technique, some values with p-values greater than 0.5 are sequentially removed from the input data set.

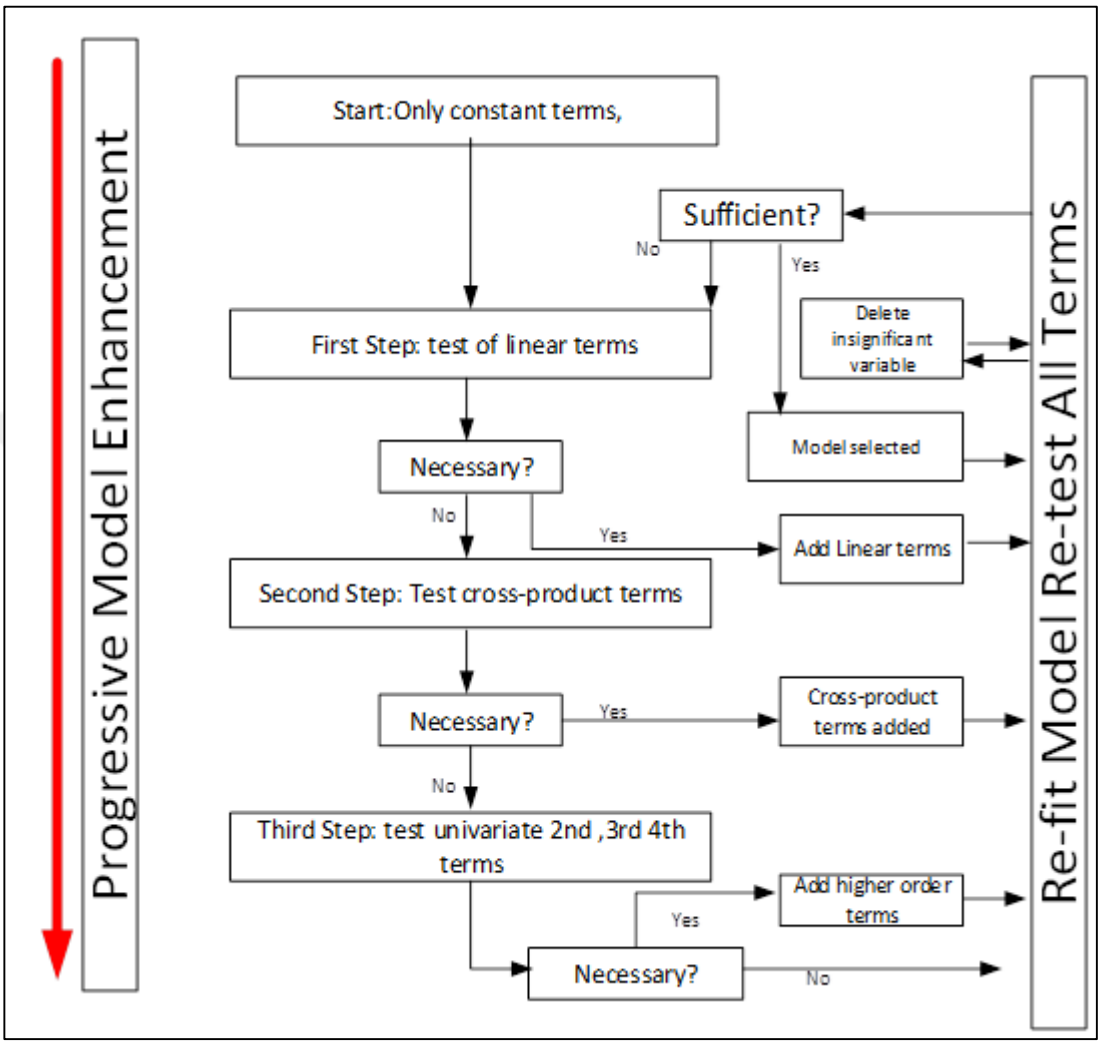


Figure 3.6: Flow diagram of stepwise regression.

It seems that the eliminated input parameters for all methods do not affect or decrease the R2-squared values. This means that removing some input parameters does not affect the accuracy of the results. Therefore, removing some parameters can lighten the processing load and reduce the running time of the algorithms.

4. CONCLUSION AND NUMERICAL RESULTS

4.1. Simple/Multiple/Polynomial Linear Regression Method

The success rates of the results obtained according to these methods are interpreted from R-squared values. Linear R-squared value is 0.840026 and polynomial R-squared value is 0.950922.

As can be understood from the results given above, the problem handled in this study is 16% error margin with linear regression method, while polynomial regression gives only 5% margin of error. In this case, it can be said that the polynomial regression method can give successful results.

Table 4.1: Linear regression OLS table.

	Coef.	Std err	t	P>|t|	[0.025	0.975]
f1	-12.1796	0.215	-56.664	0.000	-12.601	-11.758
f2	16.4630	0.709	23.220	0.000	15.073	17.853
f3	-5.3018	0.802	-6.614	0.000	-6.873	-3.730
f4	-6.6054	0.865	-7.638	0.000	-8.301	-4.910
f5	-4.5427	3.774	-1.204	0.229	-11.941	2.856
f6	-11.4269	4.838	-2.362	0.018	-20.912	-1.942
f7	-5.3406	7.953	-0.672	0.502	-20.933	10.252
f8	0.5613	7.978	0.070	0.944	-15.081	16.204
f9	-5.3297	4.798	-1.111	0.267	-14.736	4.077
f10	-2.0276	3.753	-0.540	0.589	-9.386	5.331

As shown in Table 4.1 for linear regression method:

$$f7 = 0.502(Frequency = 7GHZ \quad (4.1)$$

$$f8 = 0.944(Frequency = 8GHZ \quad (4.2)$$

$$f10 = 0.589(Frequency = 10\ GHZ \tag{4.3}$$

There can be understood that f7, f8 and f10 are greater than 0.5 therefore they can be omitted. After omitting of mentioned parameters training and running of algorithms again may result in better results for linear regression method.

Table 4.2: Polynomial regression OLS results.

	coef	std err	t	P>|t|	[0.025	0.975]
f1	-12.1802	0.228	-53.509	0.000	-12.626	-11.734
f2	16.4853	0.751	21.956	0.000	15.013	17.957
f3	-5.3635	0.849	-6.318	0.000	-7.028	-3.699
f4	-6.5959	0.916	-7.202	0 .000	-8.392	-4.800
f5	-4.6764	3.996	-1.170	0.242	-12.512	3.159
f6	-11.4973	5.124	-2.244	0.025	-21.542	-1.452
f7	-5.8561	8.422	-0.695	0.487	-22.369	10.657
f8	1.0944	8.449	0.130	0.897	-15.471	17.660
f9	-5.2412	5.081	-1.032	0.302	-15.203	4.720
f10	-1.8668	3.974	-0.470	0.639	-9.659	5.926

As shown in Table 4.2 for polynomial regression method:

$$f7 = 0.487(Frequency = 7GHZ \tag{4.4}$$

$$f8 = 0.897(Frequency = 8GHZ \tag{4.5}$$

$$f10 = 0.639(Frequency = 10\ GHZ \tag{4.6}$$

There can be understood that f7, f8 and f10 are greater than 0.5 therefore they can be omitted. After elimination of mentioned parameters training and running of algorithms again may result in better results for linear regression method.

4.2. Support Vector Regression/Machines Method

The success rates of the results obtained according to these methods are interpreted from R-squared values. SVR R-Squared value is 0.9519.

Support Vector Regression gives approximately 5% margin of error in the results obtained with Decision Tree and Random Forest methods. Therefore, it would not be wrong if one said that these methods can also be successful. However, since the output values have to be 1-dimensional in these methods, the multidimensional output values must be expressed in a single vector, then the results must be converted into a one-dimensional vector again. This method can add some extra margin of error. Considering all these and considering that the probability of overfitting (memorization) increases when the machine learning application is trained with a lot of data in the decision tree method, and if we consider that there is no need to reduce the output values to a single vector, it can be said that the most efficient method is polynomial regression. In addition, margins of error can be eliminated with backward elimination and forward elimination methods by comparing the p-values ratios, where some features may be less effective on the results. As a result, margins of error can be reduced to 1% or 2%.

Table 4.1: Support vector regression OLS report.

	coef	std err	t	P>|t|	[0.025	0.975]
f1	1.0587	0.017	61.373	0.000	1.025	1.093
f2	-0.6495	0.026	-25.375	0.000	-0.700	-0.599
f3	-0.5933	0.020	-30.175	0.000	-0.632	-0.555
f4	-0.2050	0.016	-13.166	0.000	-0.236	-0.174
f5	-0.0577	0.047	-1.238	0.216	-0.149	0.034
f6	-0.1406	0.041	-3.469	0.001	-0.220	-0.061
f7	0.0226	0.058	0.392	0.695	-0.090	0.136
f8	-0.0625	0.058	-1.078	0.281	-0.176	0.051
f9	0.0238	0.041	0.586	0.558	-0.056	0.103
f10	-0.0036	0.047	-0.077	0.939	-0.096	0.088

As shown in Table 4.3 for Support Vector Regression method:

$$f7 = 0.695(Frequency = 7GHZ) \tag{4.8}$$

$$f9 = 0.558(Frequency = 9GHZ) \tag{4.9}$$

$$f10 = 0.939(Frequency = 10\ GHZ) \tag{4.10}$$

There can be understood that f7, f9 and f10 are greater than 0.5 therefore they can be omitted. After omitting of mentioned parameters training and running of algorithms again may result in better results for linear regression method.

4.3. Decision Tree Method

The success rates of the results obtained according to these methods are interpreted from R-squared values. Decision tree R-Squared value is 0.9543.

Table 2.4: Decision tree OLS report.

	coef	std err	t	P>|t|	[0.025	0.975]
f1	-12.1796	0.227	-53.587	0.000	-12.625	-11.734
f2	16.4630	0.750	21.959	0.000	14.993	17.933
f3	-5.3018	0.848	-6.254	0.000	-6.964	-3.640
f4	-6.6054	0.914	-7.223	0.000	-8.398	-4.812
f5	-4.5427	3.990	-1.138	0.255	-12.366	3.281
f6	-11.4269	5.116	-2.234	0.026	-21.457	-1.397
f7	-5.3406	8.410	-0.635	0.525	-21.828	11.147
f8	0.5613	8.436	0.067	0.947	-15.979	17.102
f9	-5.3297	5.073	-1.051	0.294	-15.276	4.617
f10	-2.0276	3.968	-0.511	0.609	-9.808	5.753

As can be seen from Table 4.4 for decision tree method:

$$f7 = 0.525(Frequency = 7GHZ) \quad (4.11)$$

$$f8 = 0.947(Frequency = 8GHZ) \quad (4.12)$$

$$f10 = 0.609(Frequency = 10\ GHZ) \quad (4.13)$$

There can be understood that f7, f8 and f10 are greater than 0.5 therefore they can be omitted. After elimination of mentioned parameters training and running of algorithms again may result in better results for linear regression method.

4.4. Random Forest Method

The success rates of the results obtained according to these methods are interpreted from R-squared values. Random forest R-Squared value is 0.9541.

Table 4.5: Random forest OLS report.

	coef	std err	t	P>|t|	[0.025	0.975]
f1	-12.1861	0.228	-53.525	0.000	-12.633	-11.740
f2	16.4503	0.751	21.905	0.000	14.978	17.923
f3	-5.2348	0.849	-6.165	0.000	-6.900	-3.570
f4	-6.5847	0.916	-7.188	0.000	-8.381	-4.789
f5	-4.4394	3.997	-1.111	0.267	-12.276	3.398
f6	-11.5093	5.124	-2.246	0.025	-21.556	-1.462
f7	-4.9978	8.424	-0.593	0.553	-21.514	11.518
f8	0.2227	8.451	0.026	0.979	-16.346	16.791
f9	-5.2688	5.082	-1.037	0.300	-15.232	4.694
f10	-2.1808	3.975	-0.549	0.583	-9.975	5.613

As shown in Table 4.5 for linear regression method:

$$f7 = 0.553(Frequency = 7GHZ), \quad (4.14)$$

$$f8 = 0.979(Frequency = 8GHZ) \quad (4.15)$$

$$f10 = 0.583(Frequency = 10\ GHZ \quad (4.16)$$

There can be understood that f7, f8 and f10 are greater than 0.5 therefore they can be eliminated. After omitting of mentioned parameters training and running of algorithms again may result in better results for linear regression method.

Table 4.6: R2 value of methods.

Method	R2 Value
Linear Regression	0.8400264072160829
Polynomial regression	0.9509224628219423
Support Vector	0.9521927515801217
Decision Tree	0.9543175405863307
Random Forest	0.9541842322545392

From the comparison table given in Table 4.6, it is seen that it is not a good choice for the linear regression method since the R2 values are the lowest. It is also seen that the R2 value for all other methods is approximately the same and the error rate is close to 5%. However, decision tree algorithms can sometimes over fit, meaning that the algorithms don't learn but memorize the training dataset. As a result, sometimes the decision tree method can result in very bad solutions. Also, the degree of polynomial functions is a critical point for polynomial regression. Very large degrees can cause huge processing overhead, but a low rating can reduce accuracy, which means there is a trade-off. In this study, degree 4 is chosen for polynomial functions. The random forest method splits the data into several trees, and the solutions finally average the different trees. Therefore, random forest may be a safer method than the decision tree method.

4.5. Learning Curves and Detailed Comparison of Accuracy

4.5.1. Model Features and Purposes

To select the best machine learning model for a given dataset, it is crucial to consider the model's features or parameters. Parameters and model objectives help measure model flexibility, assumptions, and learning style.

For example, when comparing two linear regression models, one may aim to reduce the mean squared error but the other to reduce the mean absolute error through objective functions. To understand whether the second model is more appropriate, we need to understand whether outliers in the data are affecting the results. If anomalies or outliers need to be accounted for, then using the second model with the objective function as the mean absolute error would be the right choice.

4.5.2. Learning Curves

There is an assumption value in pre-training machine learning models to facilitate calculations. This point is the bias value. Also, as soon as the training phase begins, the estimated target will change at the rate of change in training data. This range of variation is variance. What we need to do is to reach the intersection of variance and bias in a minimal and stable position. This process is illustrated in the figure below. As can be seen in Figure 4.5, there is an important trade-off between bias and variance. While variance is proportional to model complexity, bias is inversely proportional to model complexity.

Learning curves can help data scientists observe what happens inside the model during the model's learning process. It indicates how well the optimum bias-variance point has been reached. Learning curves include training score and validation score during trials. Below are the learning curves for each machine learning method used in this study.

These curves help find optimal combinations of hyper parameters and reduce the effort of finding the best model for solution and model evaluation. Basically, a learning

curve is a way of observing learning or improvement in model scores on the y-axis and duration or trials on the x-axis.

Two main elements of learning curves are:

Training Learning Curve: It plots the evaluation metric score over time during a training process, as a result helping to observe the learning or progress of the model during training.

Validation Learning Curve: In this curve, the evaluation score is drawn in real time on the validation set.

Sometimes it can be observed that the training curve has a good score and is better, but the validation curve shows a low score ratio. This indicates that the model is overfitting, i.e. it just memorizes the data it doesn't learn and needs to be restarted from previous iterations with different models or hyper parameters within the model. In other words, the validation learning curve defines how well the model fits the solution.

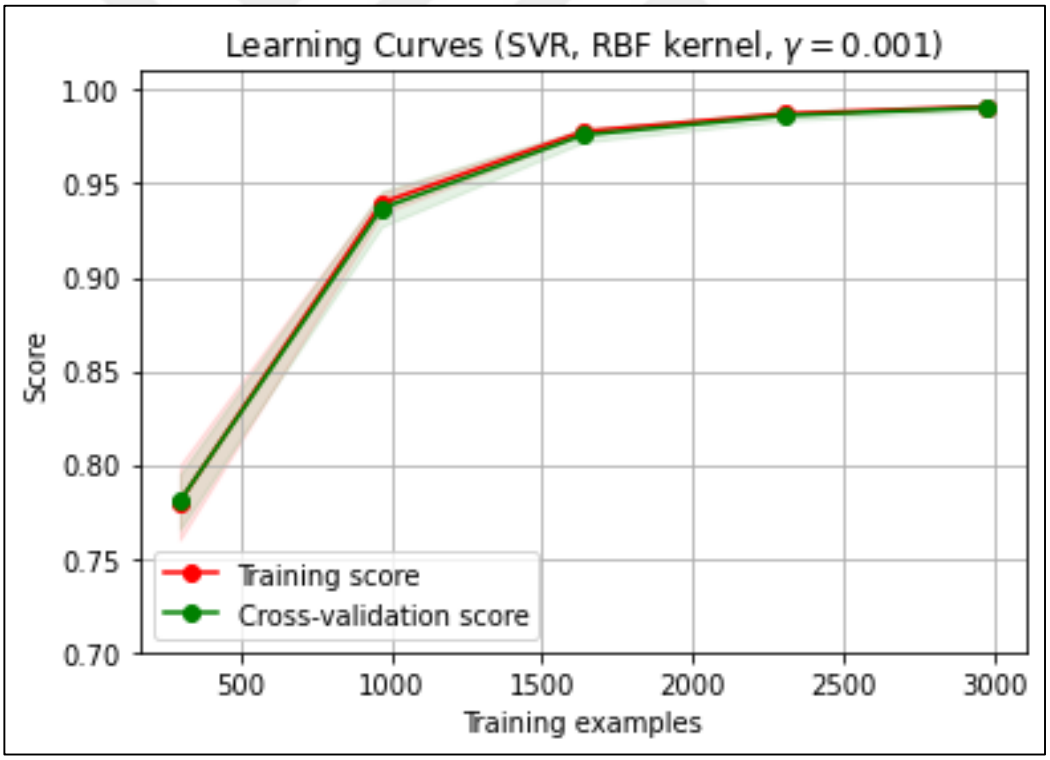


Figure 4.1: Learning curves for support vector regression.

As seen in Figure 4.1 above, the learning curve for the support vector regression method is plotted on the graph. It is understood that after 500 training examples, the support vectors start to fit into the solution. It eventually reaches a maximum of about 0.99 in 3000 training samples. As a result, it can be said that the training score and cross validation score for the support vector regression algorithms can reach their optimal points in a sufficient time interval. No more training examples needed.

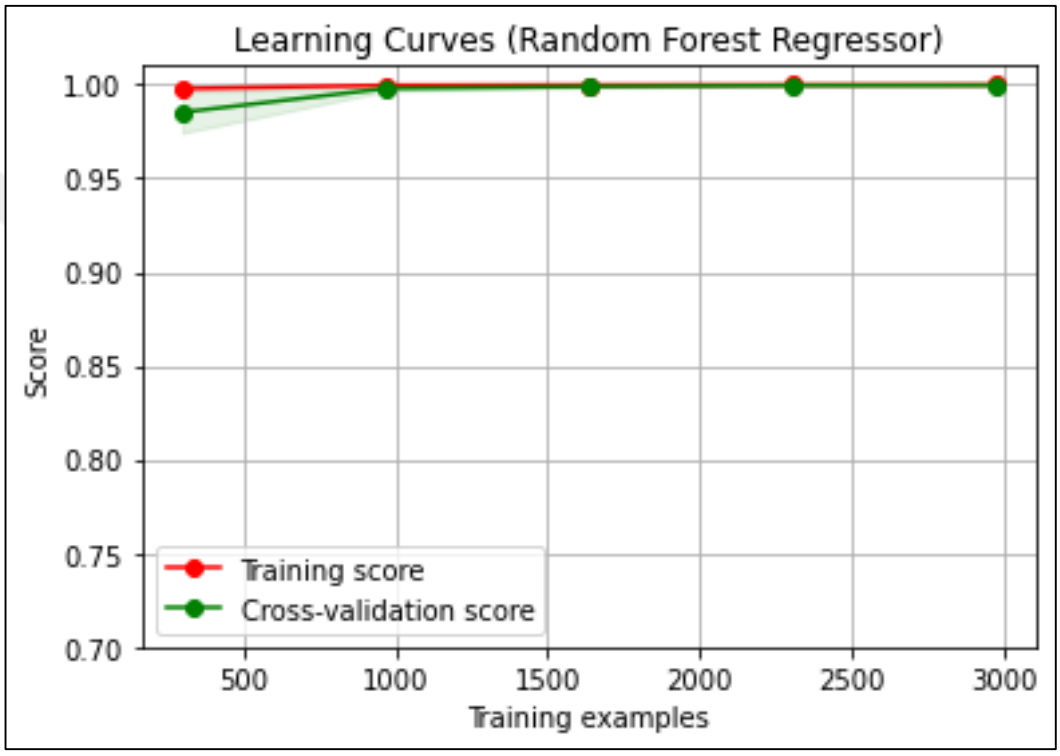


Figure 4.2: Learning curves for random forest regression.

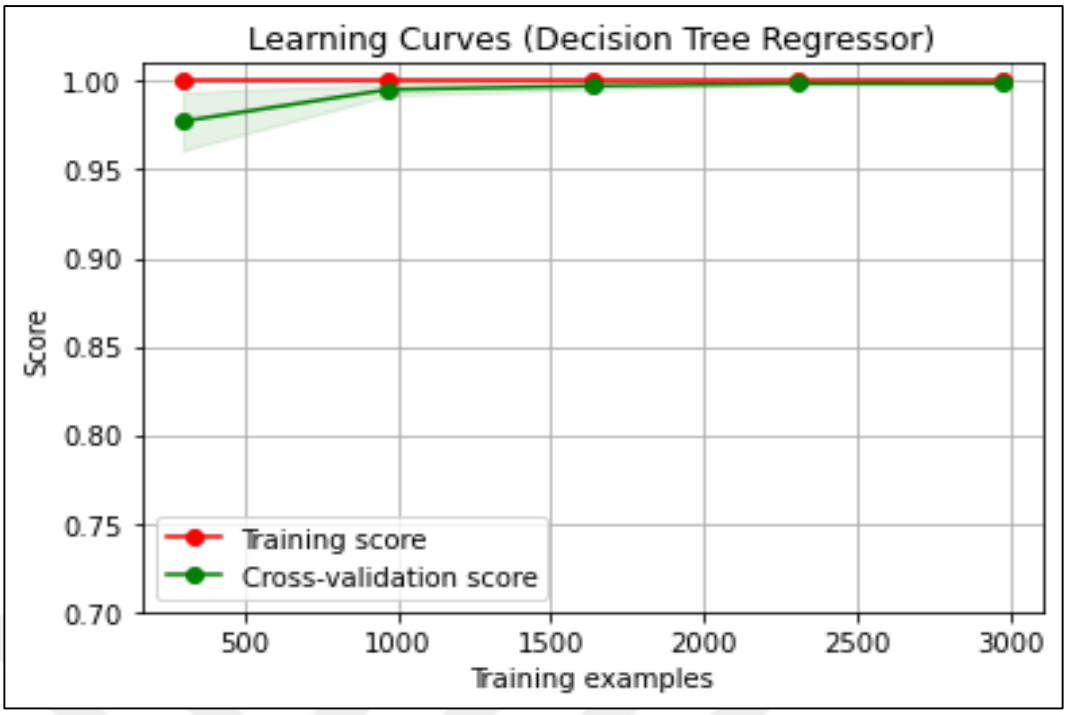


Figure 4.3: Learning curves for decision tree regression.

In Figure 4.2 and 4.3 it can be observed that both the random forest regression and the decision tree regression algorithms behave roughly the same since there are same classification functions. In addition, they can reach the optimum point faster than other methods.

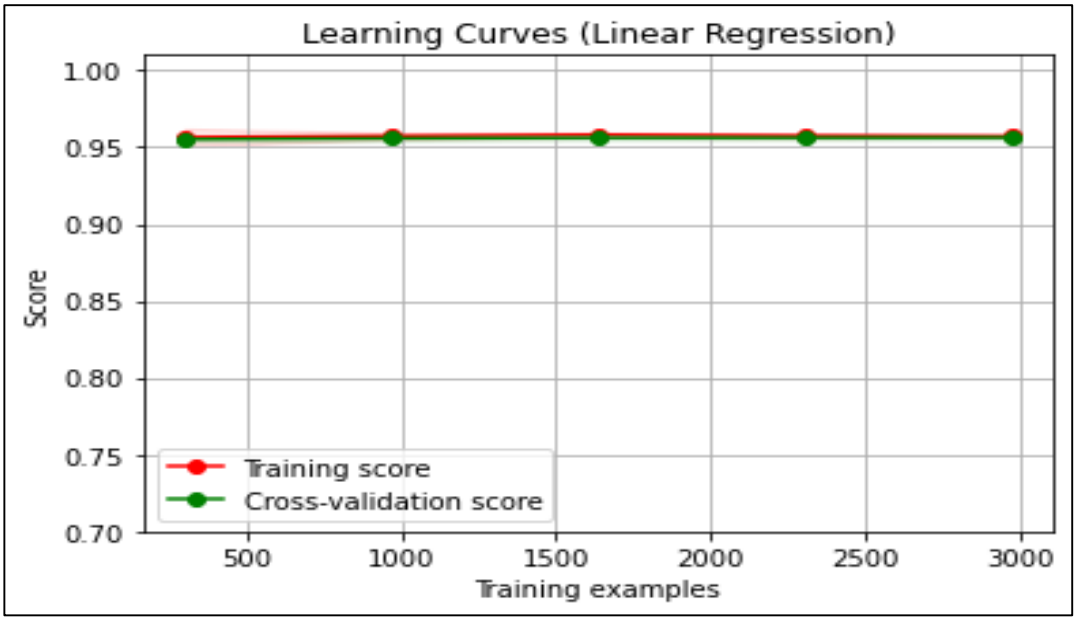


Figure 4.4: Learning curves for linear regression.

Figure 4.4 above shows the learning curves for linear regression. The learning curve includes the 0.95 value training score and the cross validation score. In addition, the training score and cross validation score do not change over time and are fixed at 0.95. This score means that the linear regression functions do not fit the solution exactly.

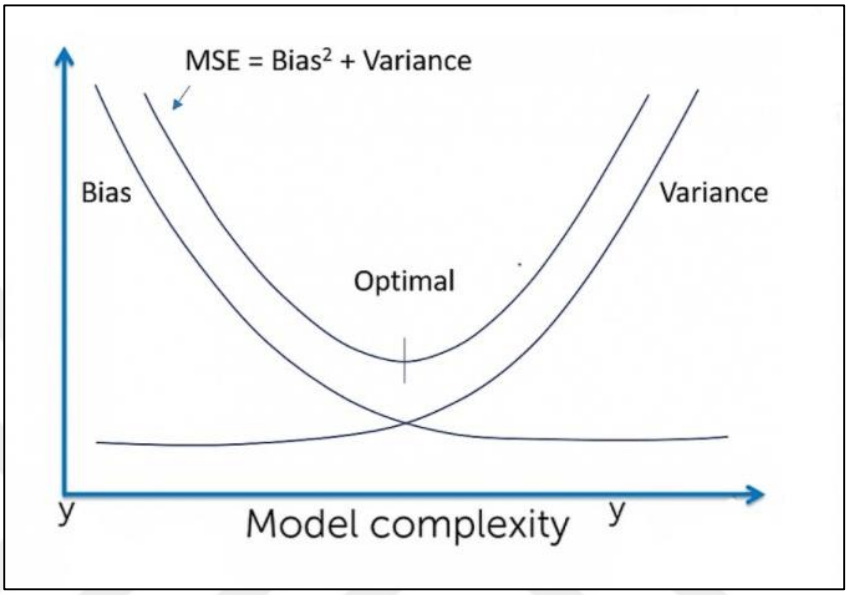


Figure 4.5: Bias variance graph.

4.5.3. Loss Functions and Metrics

There is a difference between loss functions and metric functions. Loss functions are used for model optimization or model tuning, but metric functions are used for model evaluation and selection. However, the same metrics can be used to evaluate performance and model error for optimization.

Loss functions can be thought of as arguments to models so that models can be adjusted to reduce the loss function. If the model does not fit the correct expression, a negative feedback is generated by the loss function. There are three main metrics used to evaluate models.

- Mean Absolute Error (MAE): Represents the mean error and is the easiest to understand.
- Mean squared Error (MSE). It's like MAE but the noise is exaggerated and bigger mistakes are avoided. It is more difficult to interpret than MAE because it is not in base units, but is generally more popular.
- Root Mean Squared Error (RMSE): Most popular metric similar to MSE, but result is square root for more interpretability than base units. It is recommended that data scientists to use RMSE as the primary metric for interpreting their models

Table 4.7: MAE, MSE and RMSE values for each methods.

	Linear Regression Method	Polynomial Regression Method	Support Vector Regression Method	Decision Tree Method	Random Forest Method
MAE	5.638	3.010	0.162	2.805	2.820
MSE	51.718	15.866	0.0478	14.768	14.811
RMSE	7.191	3.983	0.218	3.843	3.848

As seen in Table 4.7, linear regression has the highest error value in accordance with the R-square value. In contrast, support vector regression has the lowest error values. In addition, the polynomial regression method, the random forest method and the decision tree method have approximately the same error value. While the R-squared value of the support vector regression model is approximately the same as the other methods, the error values are lowest.

REFERENCES

[1] Pozar D., (2001), “Microwave and RF design of wireless systems”, John Wiley & sons.

[2] Hong J., Lancaster M. J., (2001), “Microstrip filters for RF/Microwave Applications”, 2nd Edition John Wiley & sons.

[3] Li Y., Lin Y., Madhusudan M., Sharma A., Xu W., Sapatnekar S., Hu J. ,(2020), “Exploring a machine learning approach to performance driven analog IC placement”, In 2020 IEEE computer society annual symposium on VLSI (ISVLSI), 24-29.

[4] Zennaro E., Servadei L., Devarajegowda K., Ecker W., (2018, August), “A machine learning approach for area prediction of hardware designs from abstract specifications”, 21st Euromicro Conference on Digital System Design (DSD), 413-420.

[5] Deisenroth M. P., Faisal A. A., Ong C.S., (2020), “Mathematics for machine learning” ,Cambridge University Press.

[6] Zhang Q. J., Gupta K. C., Devabhaktuni V. K., (2003), “Artificial neural networks for RF and microwave design-from theory to practice”, IEEE transactions on microwave theory and techniques, 51(4), 1339-1350.

[7] Watson P. M., Gupta K.C., (1997), “Design and optimization of CPW circuits using EM-ANN models for CPW components”, IEEE Transactions on Microwave Theory and Techniques, 45(12), 2515-2523.

[8] Géron A., (2019), “ Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems”, O'Reilly Media Inc.

BIOGRAPHY

Şaban ŞENLİ graduated from Middle East Technical University, Department of Electrical and Electronics Engineering in 2018 with a BS degree. In 2019, he started his master's degree at Gebze Technical University, Institute of Natural and Applied Science, Department of Electronics Engineering. He has been working as an RF Hardware Design Engineer at TÜBİTAK BİLGEM since 2019.

APPENDICES

Appendix A:

Using the formulas given in Figure A1, ABCD matrix of different types of circuit elements can be obtained.

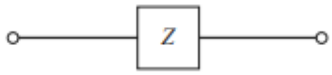
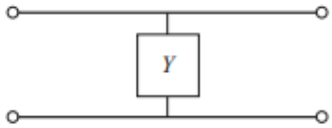
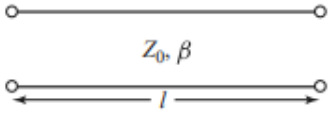
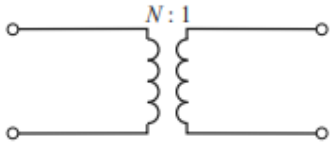
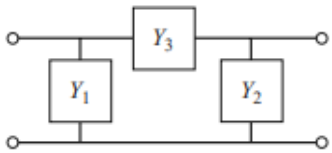
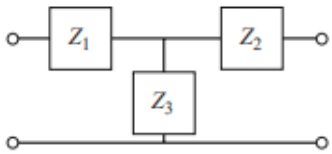
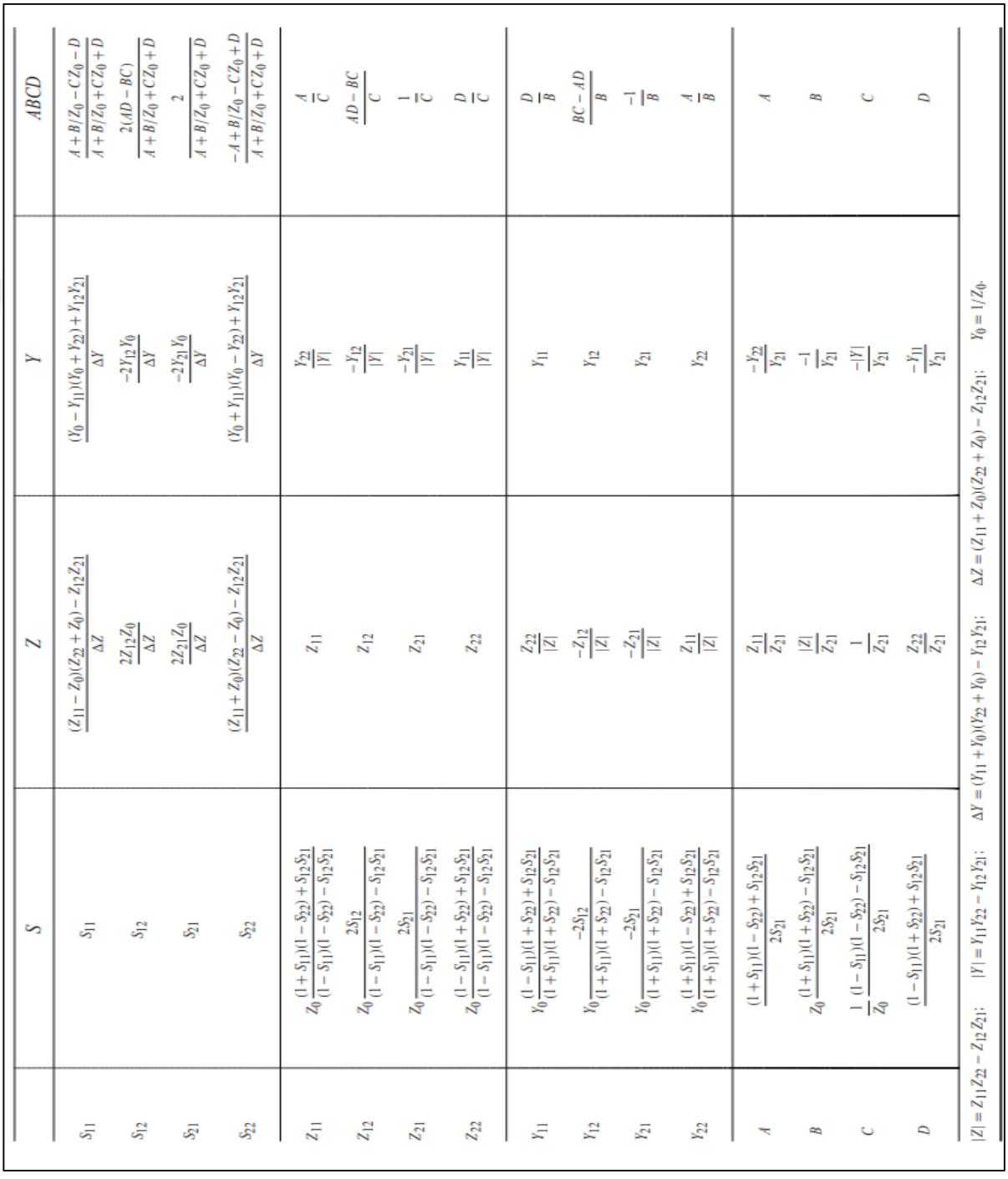
Circuit	*ABCD* Parameters	
	$A = 1$ $C = 0$	$B = Z$ $D = 1$
	$A = 1$ $C = Y$	$B = 0$ $D = 1$
	$A = \cos\beta\ell$ $C = jY_0 \sin\beta\ell$	$B = jZ_0 \sin\beta\ell$ $D = \cos\beta\ell$
	$A = N$ $C = 0$	$B = 0$ $D = \frac{1}{N}$
	$A = 1 + \frac{Y_2}{Y_3}$ $C = Y_1 + Y_2 + \frac{Y_1 Y_2}{Y_3}$	$B = \frac{1}{Y_3}$ $D = 1 + \frac{Y_1}{Y_3}$
	$A = 1 + \frac{Z_1}{Z_3}$ $C = \frac{1}{Z_3}$	$B = Z_1 + Z_2 + \frac{Z_1 Z_2}{Z_3}$ $D = 1 + \frac{Z_2}{Z_3}$

Figure A1.1: ABCD parameters for some known two-port circuits.

Appendix B:

The formulas given in Figure B1 are used to convert ABCD parameters to S parameters for all matrix elements.



	S	Z	Y	ABCD
S_{11}	S_{11}	$\frac{(Z_{11}-Z_0)(Z_{22}+Z_0)-Z_{12}Z_{21}}{\Delta Z}$	$\frac{(Y_0-Y_{11})(Y_0+Y_{22})+Y_{12}Y_{21}}{\Delta Y}$	$\frac{A+B/Z_0-CZ_0-D}{A+B/Z_0+CZ_0+D}$
S_{12}	S_{12}	$\frac{2Z_{12}Z_0}{\Delta Z}$	$\frac{-2Y_{12}Y_0}{\Delta Y}$	$\frac{2(AD-BC)}{A+B/Z_0+CZ_0+D}$
S_{21}	S_{21}	$\frac{2Z_{21}Z_0}{\Delta Z}$	$\frac{-2Y_{21}Y_0}{\Delta Y}$	$\frac{2}{A+B/Z_0+CZ_0+D}$
S_{22}	S_{22}	$\frac{(Z_{11}+Z_0)(Z_{22}-Z_0)-Z_{12}Z_{21}}{\Delta Z}$	$\frac{(Y_0+Y_{11})(Y_0-Y_{22})+Y_{12}Y_{21}}{\Delta Y}$	$\frac{-A+B/Z_0-CZ_0+D}{A+B/Z_0+CZ_0+D}$
Z_{11}	$\frac{(1+S_{11})(1-S_{22})+S_{12}S_{21}}{Z_0\frac{(1-S_{11})(1-S_{22})-S_{12}S_{21}}}$	Z_{11}	$\frac{Y_{22}}{\lvert Y\rvert}$	$\frac{A}{C}$
Z_{12}	$\frac{2S_{12}}{Z_0\frac{(1-S_{11})(1-S_{22})-S_{12}S_{21}}}$	Z_{12}	$\frac{-Y_{12}}{\lvert Y\rvert}$	$\frac{AD-BC}{C}$
Z_{21}	$\frac{2S_{21}}{Z_0\frac{(1-S_{11})(1-S_{22})-S_{12}S_{21}}}$	Z_{21}	$\frac{-Y_{21}}{\lvert Y\rvert}$	$\frac{1}{C}$
Z_{22}	$\frac{(1-S_{11})(1+S_{22})+S_{12}S_{21}}{Z_0\frac{(1-S_{11})(1-S_{22})-S_{12}S_{21}}}$	Z_{22}	$\frac{Y_{11}}{\lvert Y\rvert}$	$\frac{D}{C}$
Y_{11}	$\frac{(1-S_{11})(1+S_{22})+S_{12}S_{21}}{Y_0\frac{(1+S_{11})(1+S_{22})-S_{12}S_{21}}}$	$\frac{Z_{22}}{\lvert Z\rvert}$	Y_{11}	$\frac{D}{B}$
Y_{12}	$\frac{-2S_{12}}{Y_0\frac{(1+S_{11})(1+S_{22})-S_{12}S_{21}}}$	$\frac{-Z_{12}}{\lvert Z\rvert}$	Y_{12}	$\frac{BC-AD}{B}$
Y_{21}	$\frac{-2S_{21}}{Y_0\frac{(1+S_{11})(1+S_{22})-S_{12}S_{21}}}$	$\frac{-Z_{21}}{\lvert Z\rvert}$	Y_{21}	$\frac{-1}{B}$
Y_{22}	$\frac{(1+S_{11})(1-S_{22})+S_{12}S_{21}}{Y_0\frac{(1+S_{11})(1+S_{22})-S_{12}S_{21}}}$	$\frac{Z_{11}}{\lvert Z\rvert}$	Y_{22}	$\frac{A}{B}$
A	$\frac{(1+S_{11})(1-S_{22})+S_{12}S_{21}}{2S_{21}}$	$\frac{Z_{11}}{Z_{21}}$	$\frac{-Y_{22}}{Y_{21}}$	A
B	$Z_0\frac{(1+S_{11})(1+S_{22})-S_{12}S_{21}}{2S_{21}}$	$\frac{\lvert Z\rvert}{Z_{21}}$	$\frac{-1}{Y_{21}}$	B
C	$\frac{1}{Z_0}\frac{(1-S_{11})(1-S_{22})-S_{12}S_{21}}{2S_{21}}$	$\frac{1}{Z_{21}}$	$\frac{-\lvert Y\rvert}{Y_{21}}$	C
D	$\frac{(1-S_{11})(1+S_{22})+S_{12}S_{21}}{2S_{21}}$	$\frac{Z_{22}}{Z_{21}}$	$\frac{-Y_{11}}{Y_{21}}$	D

$|Z| = Z_{11}Z_{22} - Z_{12}Z_{21}$; $|Y| = Y_{11}Y_{22} - Y_{12}Y_{21}$; $\Delta Y = (Y_{11}+Y_0)(Y_{22}+Y_0) - Y_{12}Y_{21}$; $\Delta Z = (Z_{11}+Z_0)(Z_{22}+Z_0) - Z_{12}Z_{21}$; $Y_0 = 1/Z_0$.

Figure B1.1: Conversion between two-port network parameters.