

**REPORT ON**  
**ELECTRICAL STEEL PROPERTY PREDICTION**  
**MODEL USING MACHINE LEARNING**

By

NAME	ID
Amish Gupta	2019B5AA1386H
Saarth Jhaveri	2019A8PS0669G
Swapnil Pokale	2019A8PS0510H

AT

**JSW Steel, Vijayanagar**



**A Practice School –I Station of**



**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI**  
**(Rajasthan)**  
**June 2021**

**A  
REPORT  
ON**

**ELECTRICAL STEEL PROPERTY PREDICTION**  
**MODEL USING MACHINE LEARNING**

**By**

<b>Name</b>	<b>ID No</b>	<b>Discipline of Student</b>
<b>Amish Gupta</b>	2019B5AA1386H	Physics + Electronics and Communication engineer
<b>Saarth Jhaveri</b>	2019A8PS0669G	Electronics and Instrumentation Engineering
<b>Swapnil Pokale</b>	2019A8PS0510H	Electronics and Instrumentation Engineering

**Prepared in partial fulfillment of the  
Practice School-I Course No.  
BITS C221/BITS C231/BITS C241**

**AT**

**JSW Steel, Vijayanagar**



**A Practice School –I Station of**



**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI  
(Rajasthan)  
July 2021**

## **ACKNOWLEDGMENTS**

We wish to express our gratitude to everyone who supported us throughout the course of the project including our family and friends. We are thankful for their aspiring guidance, invaluable constructive criticism and friendly advice during these difficult pandemic affected days. We would like to thank our Project Mentor **Dr. Satish** and Co-mentor **Mr. Sunal**, who provided us with all the guidance and conducive conditions for progress of this project.

We would also like to extend our heartfelt gratitude to the entire team at JSW involved with the Practice School. They were kind enough to help provide a smooth transition to the Steel Industry through their informative lectures in the first week of this project.

We would also like to thank our PS Faculty **Prof. Tanmay Tulsidas Verlekar** for his advice and thoughts on our progress in this project so far, we thank him for all the improvements suggested by him during Practice School.

- Amish, Saarth and Swapnil.

**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE,  
PILANI-(Rajasthan)**

**Practice School Division**

**Station : JSW Steel**

**Centre: Vijayanagar, Karnataka**

**Duration: 8 weeks**

**Date of Start: 31 May 2021**

**Date of Report Submission: 16 July 2021**

**Title of the Project: “Electrical steel property prediction model using Machine learning ”**

<b>Name</b>	<b>ID No</b>	<b>Discipline of Student</b>
<b>Amish Gupta</b>	2019B5AA1386H	Physics + Electronics and Communication engineer
<b>Saarth Jhaveri</b>	2019A8PS0669G	Electronics and Instrumentation Engineering
<b>Swapnil Pokale</b>	2019A8PS0510H	Electronics and Instrumentation Engineering

**Name and Designation of the Expert:**

- 1. Dr Satish (Mentor)**  
DGM , Research And Development,  
JSW Steel
- 2. Mr. Sunal (Co- Mentor)**  
Research And Development Engineer  
JSW Steel

**Name of the PS Faculty:** Prof. Tanmay Tulsidas Verlekar

**Key words:** Data Science, Machine Learning, Steel Making

**Project Areas:** Data Analytics, Machine Learning

**Abstract:**

The electrical steel manufacturing process includes procedures such as annealing, hot rolling, cold rolling, and insulation coating, all of which have several control factors. The quality of the steel produced is heavily influenced by the process and operational factors. It is difficult to manage and maintain the appropriate steel grade due to the dynamic nature of the process. As a result, modelling such processes is critical in order to reduce the cost and time required for process optimization.

The important strength properties of electrical steel, B50, Coreloss After Aging, Core Loss Before Aging, Permeability, and Hardness, are key to the improvement and optimization of the production process.

As a result, determining the mapping functions between steel qualities and their affecting factors is critical. The goal of this project is to create a statistical model that uses machine learning to optimise process parameters and forecast steel strength qualities. This model will not only be capable of predicting properties but also addresses the process optimization problem which will ultimately help in improving the quality of produced steel.

**Signature of Student**

**Signature of PS Faculty**

**Date: 15 July 2021**

**Date**

# **TABLE OF CONTENTS**

<b>Chapter 1. Introduction</b>	<b>7</b>
<b>Chapter 2. Literature Review</b>	<b>10</b>
<b>Chapter 3. Model Building</b>	<b>12</b>
<b>Chapter 4. Result</b>	<b>23</b>
<b>Chapter 5. Reference</b>	<b>35</b>
<b>Chapter 6. Glossary</b>	<b>37</b>

# **Chapter 1. Introduction**

## **JSW Steel :**

JSW Steel Ltd. is an Indian multinational steel making company based in Mumbai, Maharashtra. It is a subsidiary of JSW Group. It is one of the fastest growing companies in India with a global footprint in over 140 countries. After the merger of Ispat steel, JSW Steel has become India's second largest private sector steel company. The current installed capacity of the company stands at 18 MTPA. A \$13 billion conglomerate, with presence across India, USA, South America & Africa, the JSW Group is a part of the O.P. Jindal Group with strong footprints across core economic sectors, namely, Steel, Energy, Infrastructure, Cement, Ventures and Sports. JSW's history can be traced back to 1982, when the Jindal Group acquired Piramal Steel Limited, which operated a mini steel mill at Tarapur in Maharashtra and renamed it as Jindal Iron and Steel Company (JISCO).

## **Vijayanagar Works :**

India's first 12 MTPA steel plant at single location, "the fastest growing steel plant in India". The JSW Steel Vijayanagar plant is the first integrated steel plant to reach 12 MTPA capacity in a single location. It is the first in India to use the Corex technology for hot metal production. Now other steel plants are coping the same.

Located at a remote village, Toranagallu part of under-developed North Karnataka in the Bellary-Hospet iron ore belt of Karnataka, the fully integrated steel plant is well-connected with both the Goa, Krishnapatnam, Mangalore and Chennai ports.



(Fig 1)

## Electrical Steel and its properties :

Electrical steel (also known as lamination steel, silicon electrical steel, silicon steel, relay steel, and transformer steel) is an iron alloy with specialised magnetic properties such as a short hysteresis area, minimal core loss, and high permeability. Electrical steel is typically produced as cold-rolled strips with a thickness of less than 2 mm.

The laminated cores of transformers, as well as the stator and rotor of electric motors, are made from these strips, which are cut to shape and stacked together. Laminations can be punched and die-cut to their final shape, or laser-cut or wire-embedded in lesser numbers.

Important Properties of Electrical Steel :-

- ***B50*** : B50 indicates the magnetic flux density
- ***Core Loss before aging*** : energy wasted by hysteresis and eddy currents in a magnetic core (as of an armature or transformer) before aging
- ***Core Loss after aging*** : energy wasted by hysteresis and eddy currents in a magnetic core (as of an armature or transformer) after aging
- ***Hardness*** : Metal hardness is a characteristic that determines the surface wear and abrasive resistance.



- **Permeability** : In electromagnetism, permeability is the measure of magnetization that a material obtains in response to an applied magnetic field.

## **Problem Statement :**

*“Machine Learning model to establish a relationship between electrical steel properties and operating parameters to optimize the production process.”*

The electrical steel manufacturing process consists of production processes including annealing, hot rolling, cold rolling, and insulation coating; these steps include lots of controlling parameters. The process and operating parameters strongly influence the quality of the produced steel. Because of the dynamic nature of the process it is not easy to control and maintain the desired steel quality. Therefore, modeling of such processes is very much needed to reduce the cost and time required for optimization of the process. B50, Coreloss After Aging, Core Loss Before Aging, Permeability, Hardness being the important strength properties of electrical steel, are key to the improvement and optimization of the manufacturing process.

Therefore, it is important to determine the mapping functions between the steel properties and their influencing factors. In this work, a statistical model using machine learning will be developed to optimize the process parameters and predict the strength properties of steel. This model will not only be capable of predicting properties but also addresses the process optimization problem which will ultimately help in improving the quality of produced steel.

## **Chapter 2. Literature Review**

- **Comprehensive Research and Analysis**

We went through a lot of research papers on prediction algorithms, in order to check the outputs, results obtained and the parameters used. The primary objective was to look for the necessary and vital parameters, and types of algorithms being used in similar industries for similar work, and we noted down all the important details and summary for which was compiled in an excel sheet for a concise reference([Link to the research papers Summary](#)).

After reading through research papers we were able to conclude that the physical properties of produced electrical steel was directly related to some predefined parameters and hence according to availability of data we can build a model to work accordingly to predict physical properties of electrical steel thus produced.

Of all the papers we read one which was close to our work was titled as- ‘Online prediction of mechanical properties of hot rolled steel plate using machine learning’. Description of which can be found below:

In this paper, based on the composition and production process parameters of hot-rolled steel sheets, a DNN model is developed to predict the mechanical properties of hot-rolled steel sheets, which is suitable for advanced steel manufacturing plants in the real world. The factory data used in this study includes 27 input features, 4 target variables, and 11,101 data points from the “hot rolling process”, which is the focus of the production process in this study. In the process of model development, the parameters that affect the machine learning process, such as preprocessing, network structure, learning rate, optimization algorithm, etc., were adjusted and analyzed in detail. The DNN model with a topological structure ( $27 \times 200 \times 200 \times 4$ ), combined with the Z Score preprocessing method, Adam optimizer

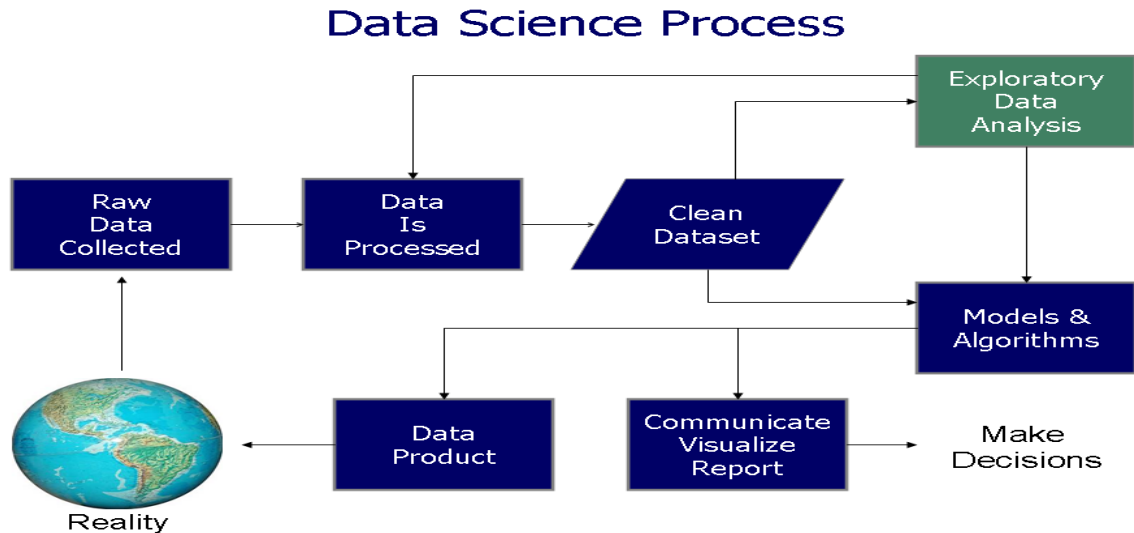
and a learning rate of 0.0001, achieved the highest prediction accuracy in the test data set,  $R^2 = 0.907$ . For each of the 4 target variables, the DNN model produced RMSE of 23.08 MPa, 17.72 MPa, 2.35%, and 40.14 J for YS, UTS, and EL, and RMSPE and 16.2% of 4.7%, 2.9%, 7.7%, and 16.2%.

Akv. These results show that the DNN model developed in this study can fit the complex relationship between the mechanical properties and process parameters of the steel plate and the composition of the steel grade. The prediction accuracy of this model is also compared to other classic machine learning algorithms, among which DNN performs better than other algorithms. Based on LIME, several local linear interpretation models of the ANN model are analyzed. For YS and UTS,  $\text{tfce}$  is one of the most important parameters. For the prediction of EL, the composition characteristics, that is, the elements Nb, Cr and V are the three most influential parameters, while for Akv, it is observed that the content of C plays the most important role in the four degrees of steel. The interpretation results of the DNN model conform to the laws of physical metallurgy, which further implies that the DNN model can be used to guide and guide metallurgical research. The developed model was successfully applied online to Steel Plant, and data was obtained from it. The system is connected to the manufacturing execution system (MES) of the factory, and retrain the neural network based on new product data, and regularly assists in online monitoring and control of production.

## Chapter 3. Model Building

- **Data Analysis and Cleaning**

Studying and learning the process of data analysis was crucial as the data amount and parameters were very high and the raw data required cleaning. The below image briefly explains the process



(Fig 2)

Deleting the data causing abnormalities is one of the easiest ways to proceed forward but is only feasible when there is abundant availability of data but in cases where we can't delete data, imputation is the process of replacing missing data with substituted values.

The Data Imputation methods are:

1. **Mean substitution**

This imputation technique involves replacing any missing value with the mean of that variable for all other cases, which has the benefit of not changing the sample mean for that variable. However, mean imputation attenuates any correlations involving the variable(s) that are imputed. This is because, in cases with imputation, there is

guaranteed to be no relationship between the imputed variable and any other measured variables. Thus, mean imputation has some attractive properties for univariate analysis but becomes problematic for multivariate analysis.

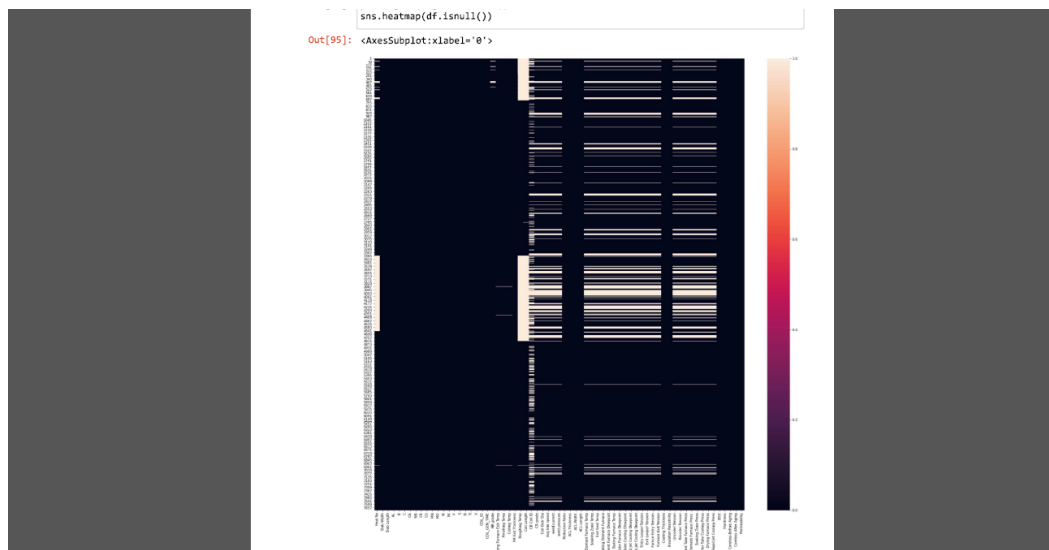
## 2. Multiple Imputation by Chained Equations (MICE)

MICE is a multiple imputation method used to replace missing data values in a data set under certain assumptions about the data missingness mechanism. This fills in the missing values of a feature by treating it as a regression problem involving all the other features.

### Data Preprocessing:

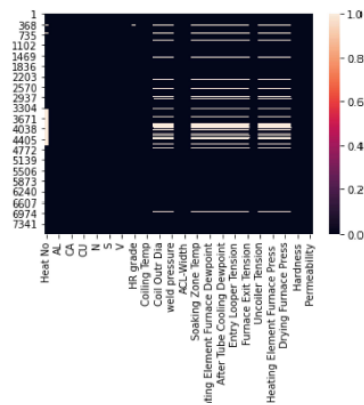
Data was collected at the Steel Plant of JSW Steel Vijaynagar. Data included the values of 67 important parameters involved in the production of high quality electrical steel. The regression task aims at the prediction of 5 important features namely **B50**, **Hardness**, **Permeability**, **Core Loss After Aging** and **Core Loss Before Aging** from the other 62 independent features.

The Dataset was far from an ideal dataset and hence Pre-Processing was of pivotal importance. Multiple features were removed due to a high frequency of missing values ( refer to Fig 3 and Fig 4).



(Fig 3. Raw Dataset)

```
Out[100]: <AxesSubplot: xlabel= '0'>
```

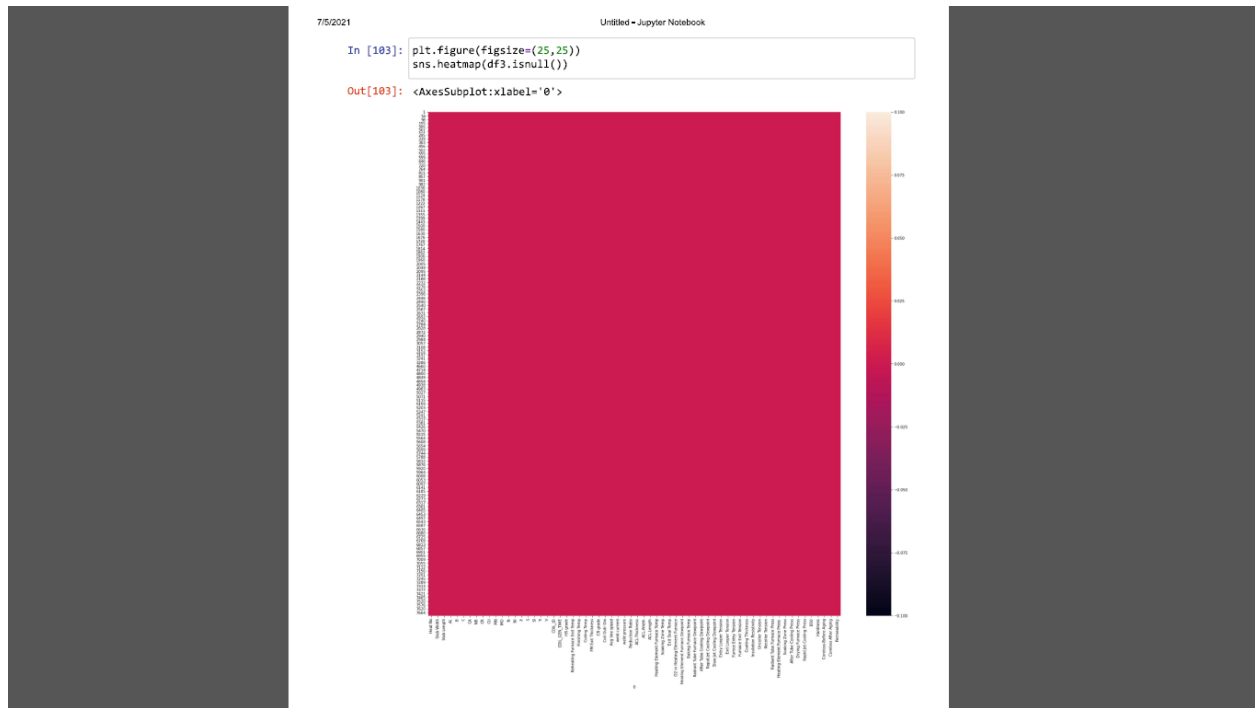


(Fig 4. Dataset during cleaning)

## Data removal

In statistics, imputation is the process of replacing missing data with substituted values. Also when dealing with data that is missing at random, related data can be deleted to reduce bias. There are three main problems that missing data causes: missing data can introduce a substantial amount of bias, make the handling and analysis of the data more arduous, and create reductions in efficiency.

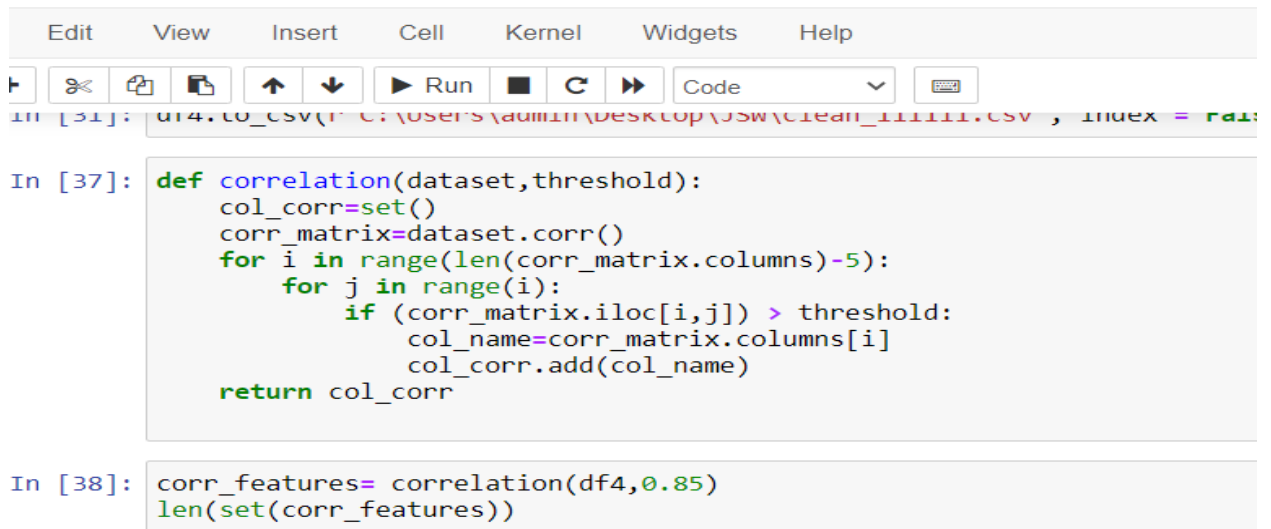
To handle the missing values in dataset, all the features in dataset having more than 20% of its values as null were dropped(3 features) in order to ensure good efficiency of the model, on the same lines the rows containing any missing values or some out of bound values were dropped as they accounted for only 18% of the original data. Also few features like Coil Id were dropped as they were not responsible for properties of electrical steel.



(Fig 5. Final Dataset used)

## Principal Component Analysis

Feature Engineering reduced the number of features in the data to 62, however, this is of a very high dimensionality. Therefore, PCA was implemented on the improvised dataset. The 35 Principle components were selected since they were able to express 85 percent of the variance of the dataset. Hence the total dimensionality was successfully reduced to 35.



```

In [31]: df4.to_csv('C:\Users\admin\Desktop\JSW\Clean_111111.CSV', index = False)

In [37]: def correlation(dataset, threshold):
          col_corr=set()
          corr_matrix=dataset.corr()
          for i in range(len(corr_matrix.columns)-5):
              for j in range(i):
                  if (corr_matrix.iloc[i,j]) > threshold:
                      col_name=corr_matrix.columns[i]
                      col_corr.add(col_name)
          return col_corr

In [38]: corr_features= correlation(df4,0.85)
          len(set(corr_features))

```

(Fig 6. code snippet for finding correlated features)

- **Building Machine Learning:**

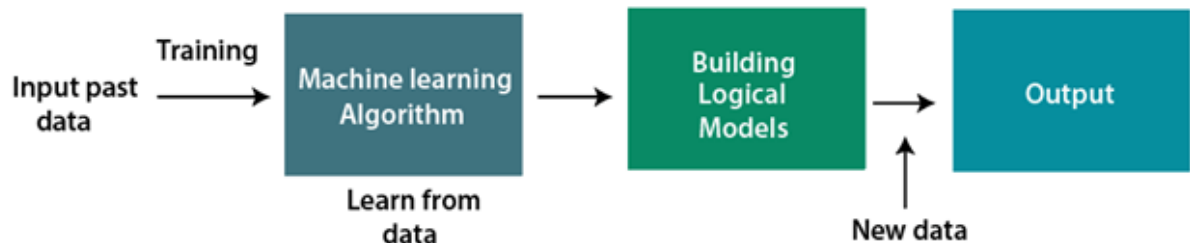
Machine learning, as defined by Arthur Samuel, a computer scientist who pioneered the science of artificial intelligence in 1959, is "the study of giving computers the ability to learn without being explicitly taught."

A more technical definition given by Tom M. Mitchell (1997) : “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.”

Machine learning is one of the most popular areas of computer science, with several applications. Over the previous few decades, it has become a common approach in almost every task that requires information extraction from large data sets. Anti-spam software learns how to filter our email messages, and credit card transactions are protected by software that learns how to detect frauds. Smartphones with intelligent personal assistance software learn to recognise voice commands and digital cameras. Learn to recognise people's faces.



A common feature of all of these applications is that, unlike more typical uses of computers, a human programmer cannot provide an explicit, fine-tuned specification of how such operations should be carried out in these circumstances due to the intricacy of the patterns that must be recognised. As intelligent beings have demonstrated, many of our skills are gained or strengthened by learning from our experiences (rather than following explicit instructions given to us). Machine learning technologies are designed to enable programmes to "learn" and adapt.



(Fig 7)

Machine Learning, Deep Learning, and Artificial Intelligence are all terms for an area of automation and computational statistics study in which the goal is to enable machines to make decisions based on prior experience, or data in this case. A computer can utilise "Sample Data" to uncover patterns in the inputs it receives and predict/classify/detect specific traits.

There are 3 approaches to machine learning:

- **Supervised learning:** The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.
- **Unsupervised learning:** No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).
- **Reinforcement learning:** A computer program interacts with a dynamic

environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.

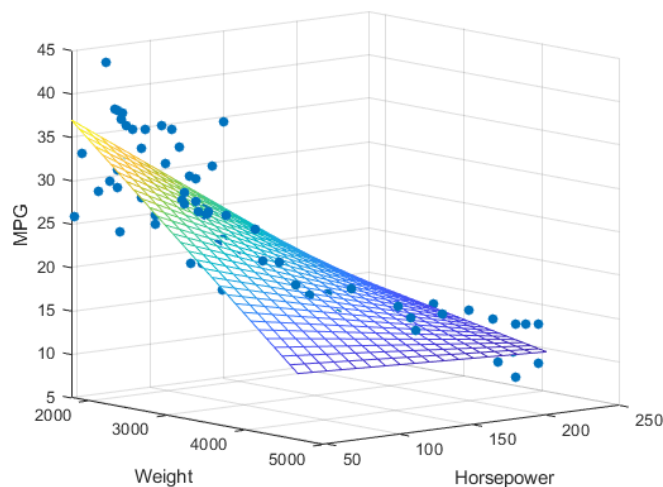
## Machine Learning Algorithms:

There are many sorts of machine learning algorithms for different business goals and data sets, so model creation isn't a one-size-fits-all operation. The comparatively simple linear regression approach, for example, is easier to train and execute than other machine learning algorithms, but it may not contribute value to a model that requires sophisticated predictions.

The models which we used for the project are -

- **Multiple Linear Regression**

Multiple linear regression (MLR), often known as multiple regression, is a statistical technique that predicts the result of a response variable by combining numerous explanatory variables. Multiple linear regression (MLR) attempts to represent the linear relationship between explanatory (independent) and response (dependent) variables.



(Fig 8. Hyperplane for two input MLR)

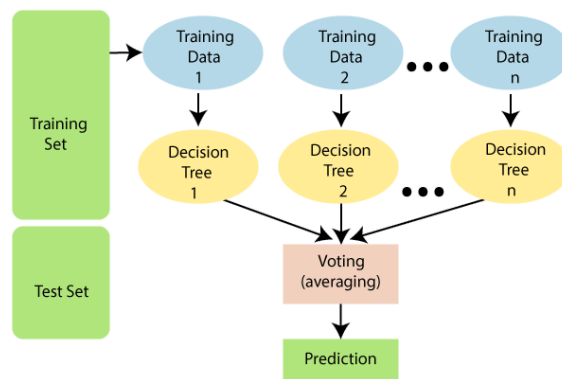
- **Random Forest**

Random Forest is a well-known machine learning algorithm that uses the supervised learning method. In machine learning, it can be utilised for both classification and regression problems. It is based on ensemble learning, which is a method of integrating several classifiers to solve a complex problem and increase the model's performance.

As the name suggests, "Random Forest is a classifier that contains a number of decision trees on various subsets of a given dataset and takes the average to enhance the predicted accuracy of that dataset". Instead of relying on a single decision tree, the random forest collects the forecasts from each tree and predicts the final output based on the majority votes of predictions.

The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

The below diagram explains the working of the Random Forest algorithm:



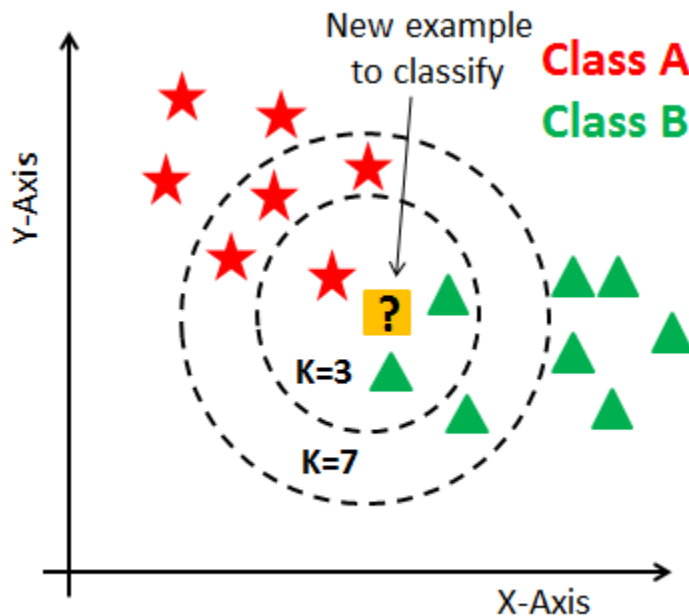
(Fig 9. Random forest)

- **K-Nearest Neighbours**

The function is only approximated locally in k-NN classification, and all computation is postponed until the function is evaluated. Because this method relies on distance for classification, normalising the training data can

greatly increase its performance if the features represent various physical units or come in wildly different scales.

Both classification and regression are possible with this method.



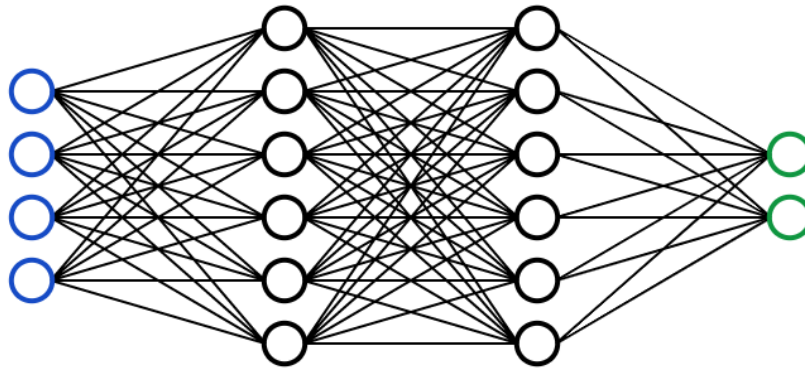
(Fig 9. KNN with 3 and 7 neighbours)

- **Artificial Neural Networks**

A Neural Network is essentially a network of mathematical equations. It takes one or more input variables, and by going through a network of equations, results in one or more output variables.

An ML algorithm that can work within any learning approach. It is designed to imitate the processes of the human brain. ANNs are built using a single layer (perceptron) or several layers of neurons that can pass the same input to various weights and biases, thus naturally tending towards a required

solution. Subsequent sections contain an in-depth explanation about the working of a Neural Network. Previous Applications: ANNs are very commonly used in control systems of vehicles and aircraft, pattern recognition in radar and surveillance systems and have a wide range of applications in medicine, chemistry and deeper research in computer science. It is by far one of the most commonly used algorithms.



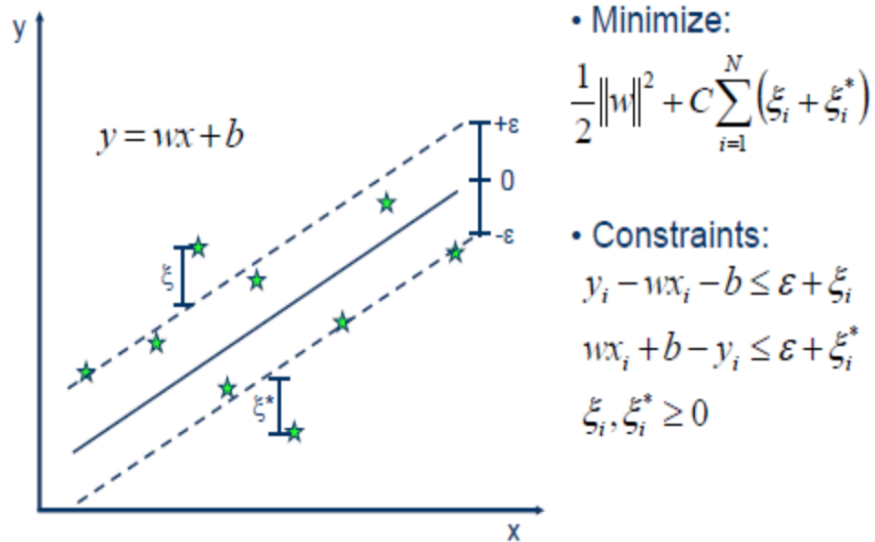
(Fig 10.ANN with 2 hidden layers)

The input layer is represented by blue circles, the hidden levels are represented by black circles, and the output layer is represented by green circles. Each node in the hidden levels represents a linear function as well as an activation function that the nodes in the preceding layer pass through on their way to the green circle output.

- **Support vector regression**

Support Vector Machine can also be used as a regression method, maintaining all the main features that characterize the algorithm (maximal margin). The Support Vector Regression (SVR) uses the same principles as the SVM for classification, with only a few minor differences. First of all, because output is a real number it becomes very difficult to predict the information at hand, which has infinite possibilities. In the case of regression, a margin of tolerance (epsilon) is set in approximation to the SVM which would have already been requested from the problem. But

besides this fact, there is also a more complicated reason, the algorithm is more complicated therefore to be taken in consideration. However, the main idea is always the same: to minimize error, individualizing the hyperplane which maximizes the margin, keeping in mind that part of the error is tolerated.



(Fig 11.SVR algorithm)

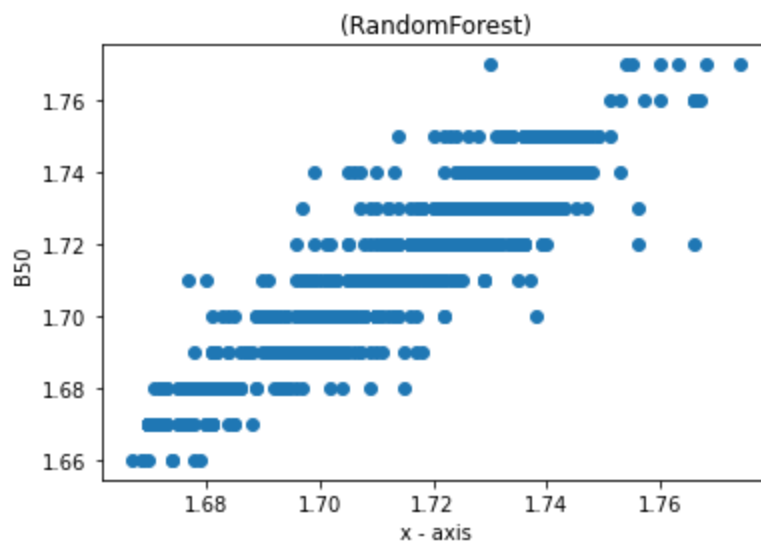
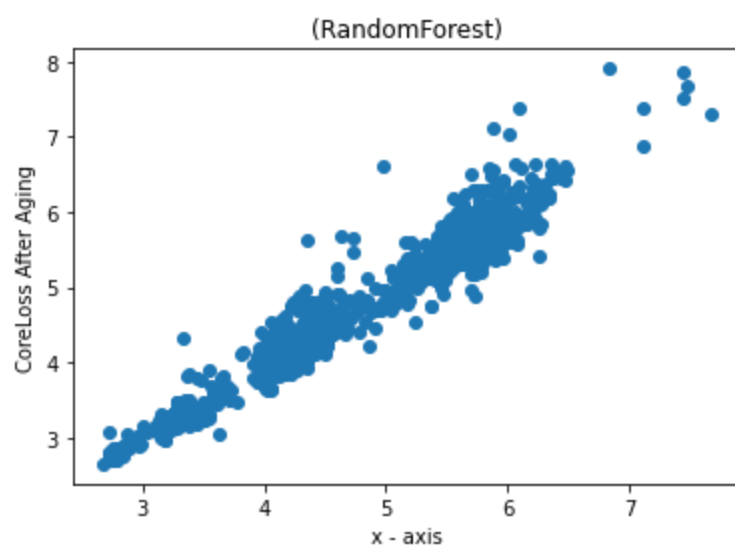
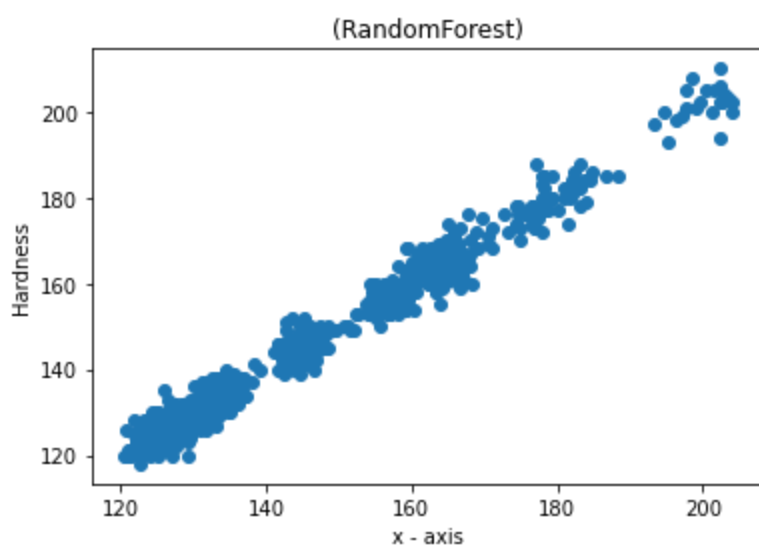
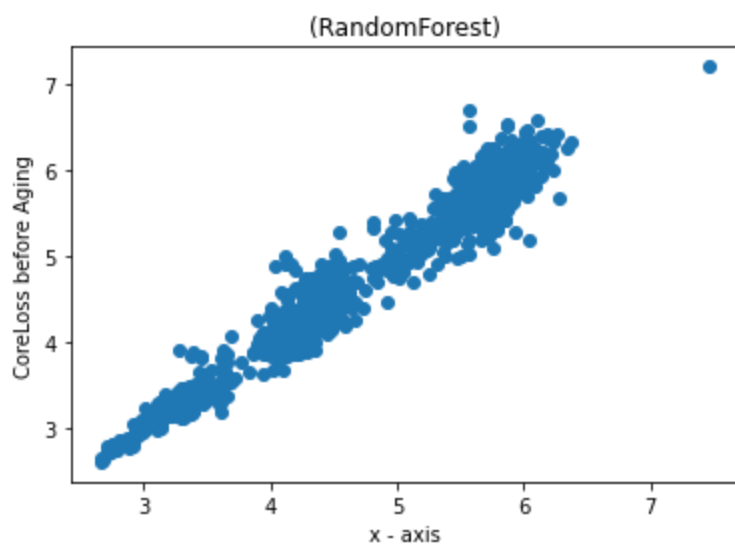
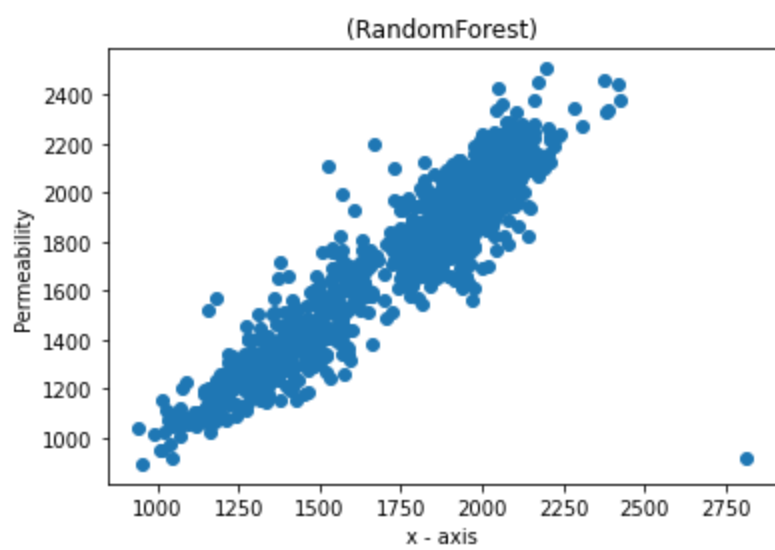
## **Chapter 4. Result**

### **Random Forest**

- Random Forest was found to be comparatively the best learner for this task.
- The effects of different numbers of trees (n\_estimator ) in the Forest was studied for a particular train and test set.
- It was observed that for multiple test-train splits of the dataset, n\_estimator=10, and random\_state=0; achieved the maximum accuracy.
- Hence, n\_estimator=10, was studied in detail, and the following are the obtained results -

(Table 1. Random Forest)

<b>Property</b>	<b>R_sq Value</b>	<b>MSE</b>
Hardness	0.9755	2.576
Core Loss Before Aging	0.9459	0.212
Core Loss After Aging	0.9314	0.2312
Permeability	0.8612	127.6016
B50	0.8359	0.008453



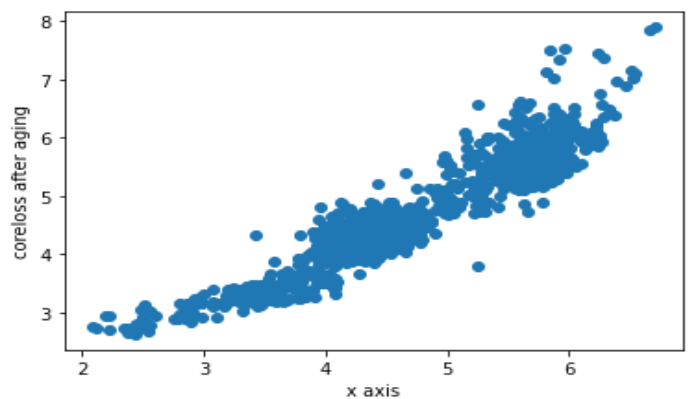
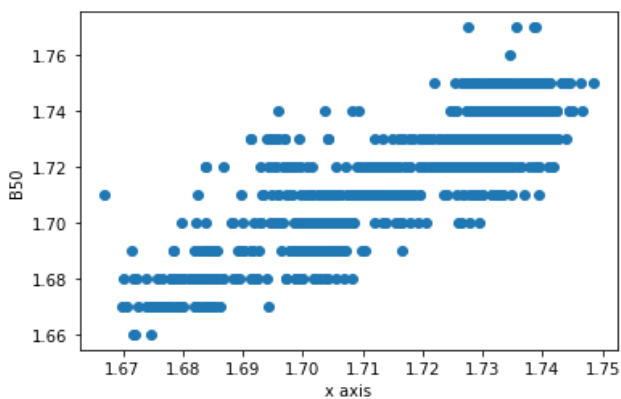
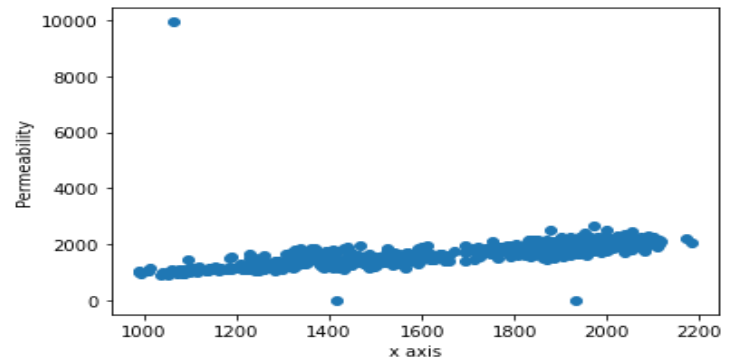
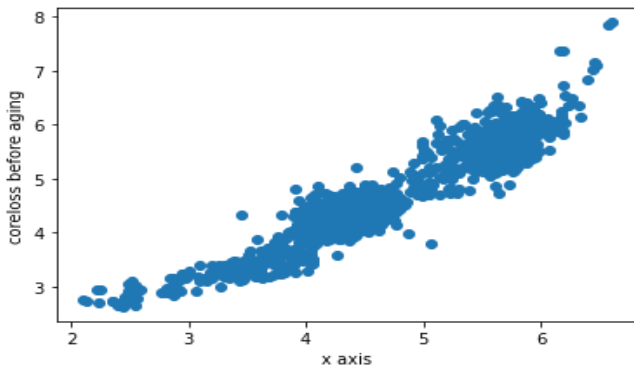


## Multiple Regression

- As expected, MLR proved to be a better learner for the task as compared to Support Vector Machine(SVM). But, it was still a comparatively weak learner as compared to Random Forest which was explored in the later part of this report.
- Fig 4, shows the estimation of R2 score for different values of physical properties of steel .

(Table 2. Multiple Regression)

Property	R_sq Value	MSE
Hardness	0.592	2.89602
Core Loss Before Aging	0.8832	0.2943
Core Loss After Aging	0.8906	0.3194527
Permeability	0.753	304.52119
B50	0.7208	0.0109

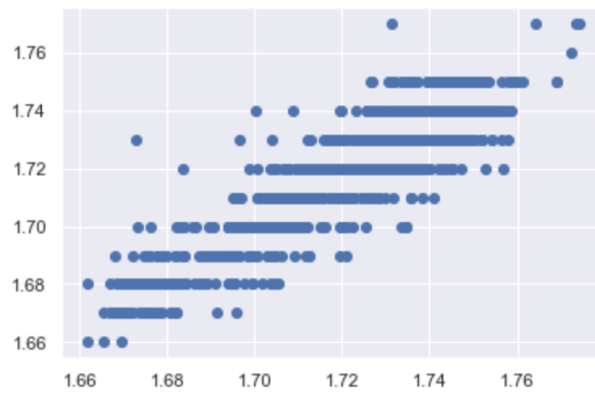


## Artificial Neural Network

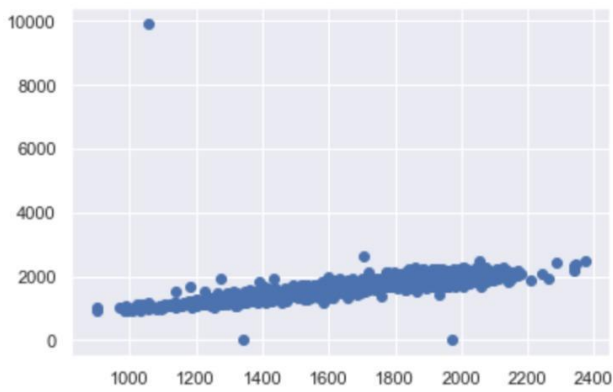
- Towards the end fourth week of the project, an attempt at testing the accuracy of a simple ANN on the task was done.
- A simple ANN with 3 hidden layers consisting of 100, 50 and 25 neurons with the ReLu activation function implemented in Keras gave promising results.

(Table 3.ANN)

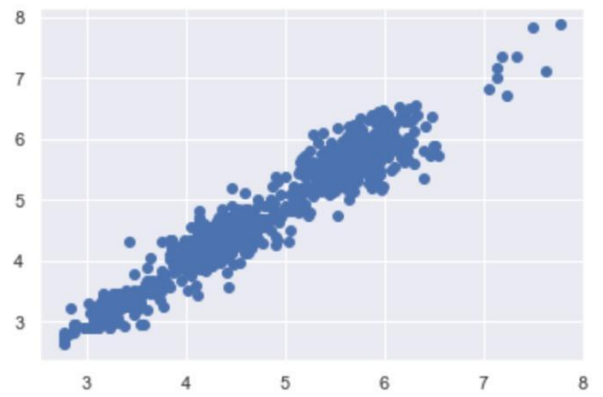
Property	R_sq Value	MSE
Hardness	0.968	3.565277
Core Loss Before Aging	0.939	0.2367
Core Loss After Aging	0.9037	0.26322
Permeability	-0.002	299.89238
B50	0.768	0.010091



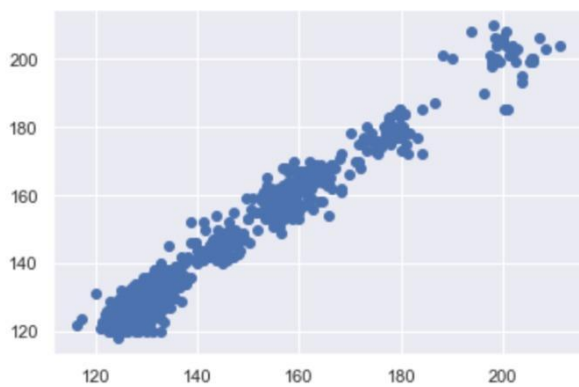
**B50: Predicted v/s Actual**



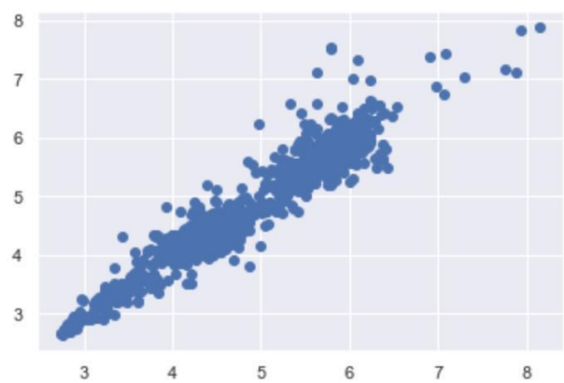
**Permeability: Predicted v/s Actual**



**CBA: Predicted v/s Actual**



**Hardness: Predicted v/s Actual**



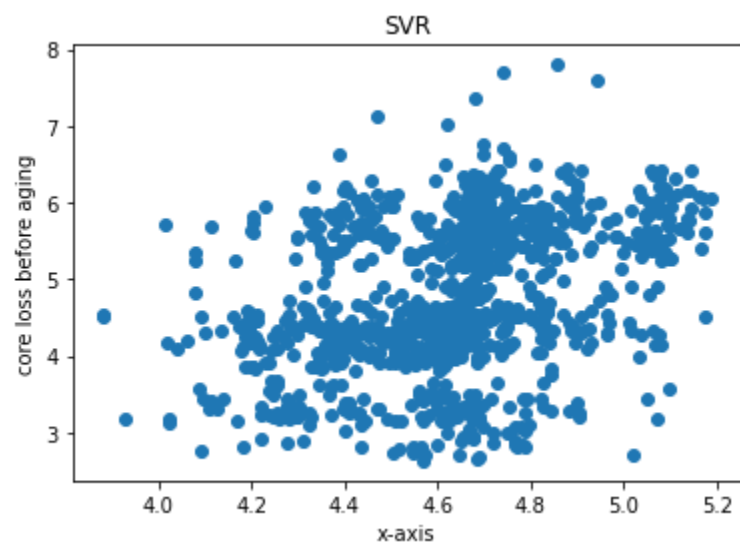
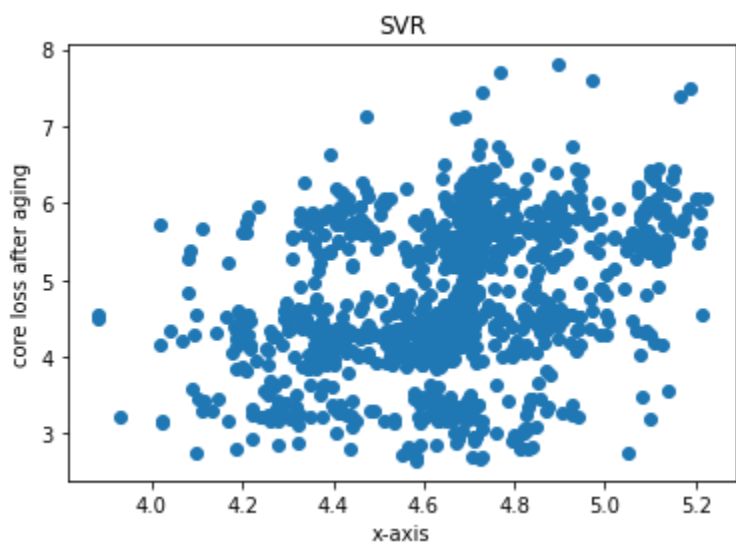
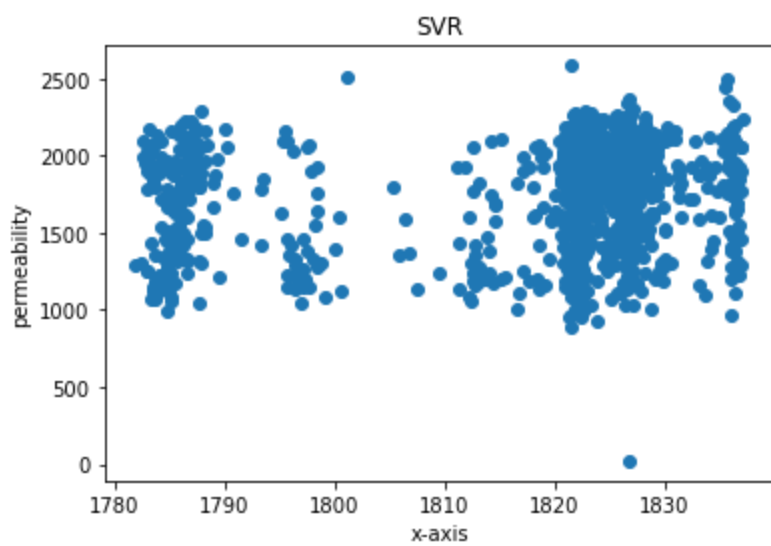
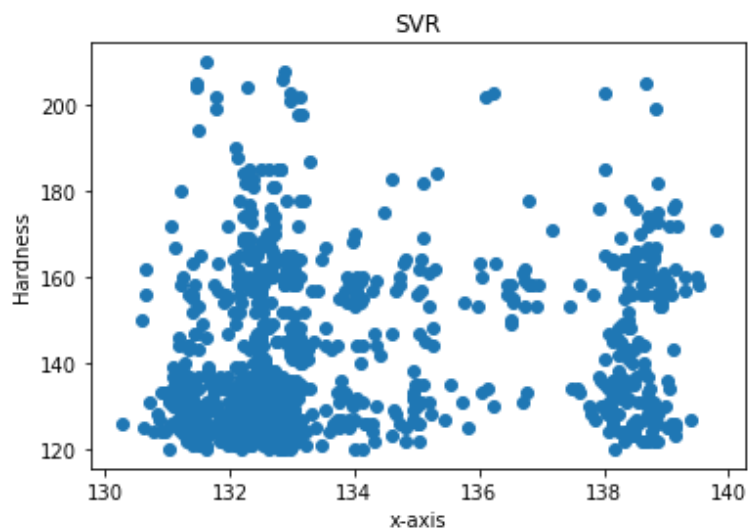
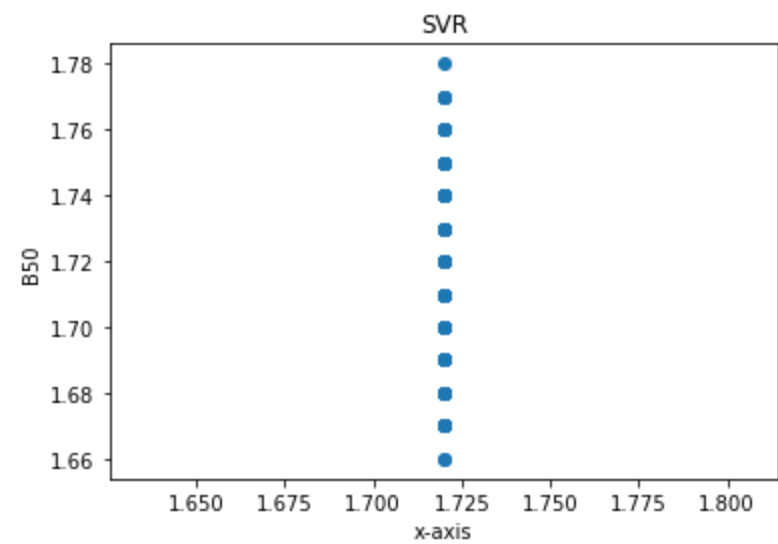
**CAA: Predicted v/s Actual**

## Support Vector Machine

- Support Vector Machine and ANN were the two such models, which were found to be comparatively weak learners for this task.

(Table 4.SVM)

Property	R_sq Value	MSE
Hardness	-0.102	18.429
Core Loss Before Aging	0.1040	0.913
Core Loss After Aging	0.1047	0.92447
Permeability	-0.0228	333.4026
B50	-0.006	0.0219

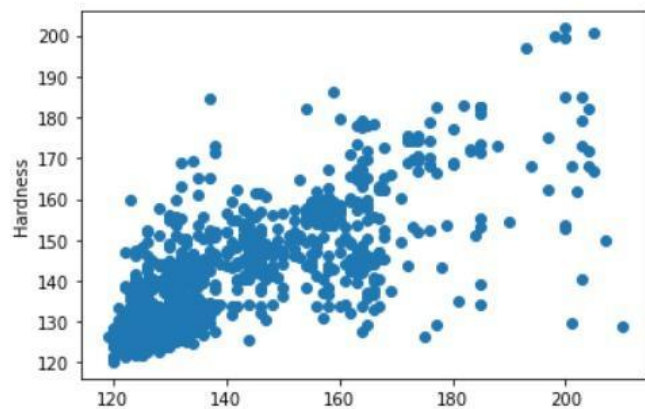
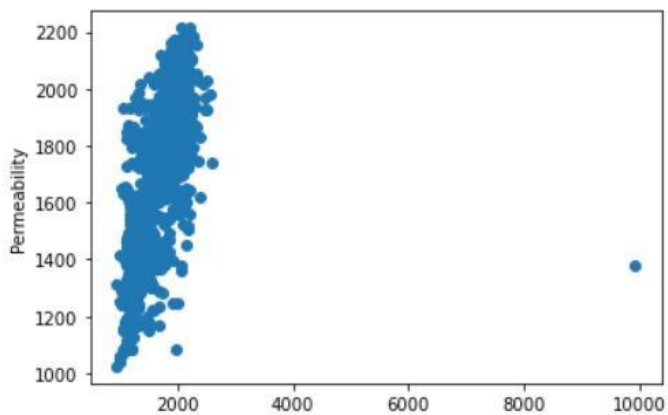
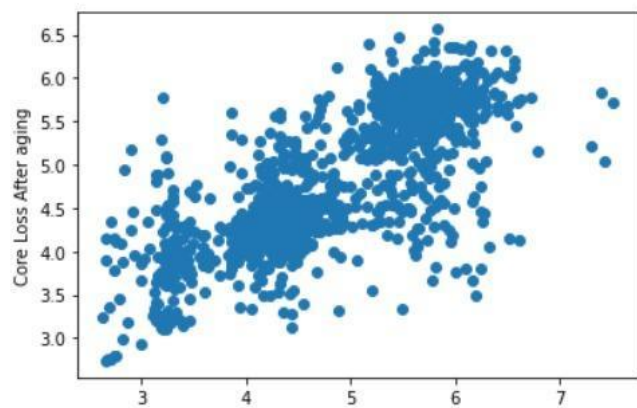
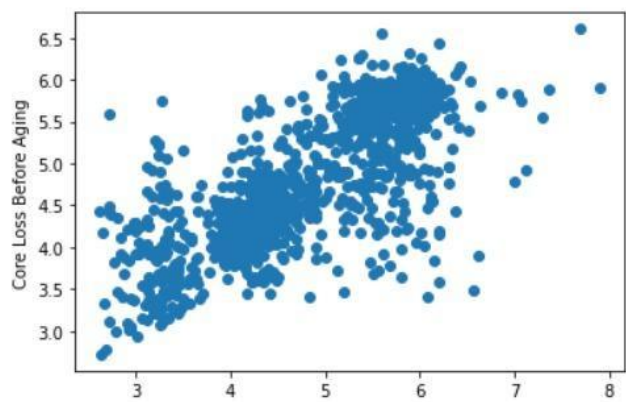
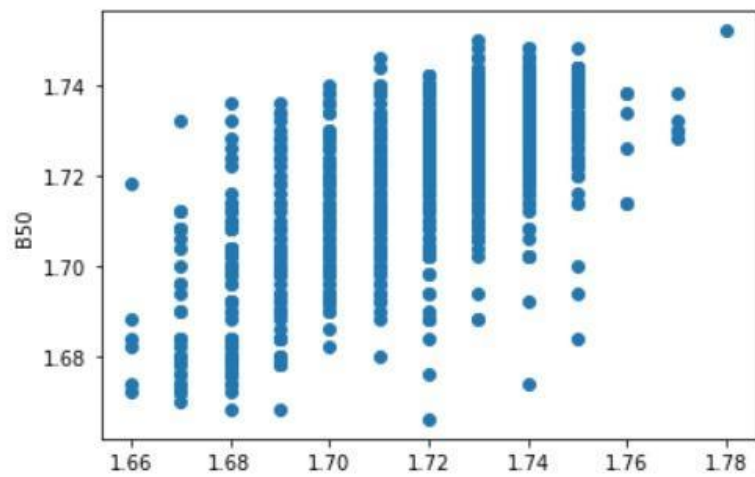


## KNN

- KNN is a non parametric technique, and in its classification it uses  $k$ , which is the number of its nearest neighbors, to classify data to its group membership. It is majorly used for classification problems , but it was found out to be a relatively weak learner for our model .

(Table 5. KNN)

Property	R_sq Value	MSE
Hardness	0.602	137.32856
Core Loss Before Aging	0.567	0.38344
Core Loss After Aging	0.56	0.3754
Permeability	0.314	121442.29
B50	0.46	0.0002



## Conclusion:

- Electrical steel (lamination steel, silicon electrical steel, silicon steel, relay steel, transformer steel) is an iron alloy tailored to produce specific magnetic properties. These properties are affected by many process parameters.
- In the present work, a machine learning model was developed for predicting 5 important material properties - Hardness, core Loss Before Aging, Core Loss After Aging, Permeability and B50 using process parameters.
- Total 5831 data points were considered and 35 variables were used in building the model.
- The machine learning approach was applied using 5 different algorithms Multiple Regression, ANN, Random Forest, SVM, and KNN. Among all, the r-square value was highest, and the mean square error was lowest for the random forest algorithm, which is suggested for implementation.

### **R<sup>2</sup> value**

(Table 6. R<sup>2</sup> comparison)

<b>Algorithm</b>	<b>Hardness</b>	<b>Core Loss Before Aging</b>	<b>Core Loss After Aging</b>	<b>Permeability</b>	<b>B50</b>
Multiple Regression	0.592	0.8832	0.8906	0.753	0.7208
ANN	0.968	0.939	0.9037	-0.002	0.768
Random Forest	0.9755	0.9459	0.9314	0.8612	0.8359
SVM	-0.102	0.1040	0.1047	-0.0228	-0.0006
KNN (5 neighbour)	0.602	0.567	0.56	0.314	0.46



## MSE Value

(Table 7. MSE value comparison)

Algorithm	Hardness	Core Loss Before Aging	Core Loss After Aging	Permeability	B50
Multiple Regression	2.89602	0.2943	0.3194527	304.52119	0.0109
ANN	3.565277	0.2367	0.26322	299.89238	0.010091
Random Forest	2.576	0.212	0.2312	127.6016	0.008453
SVM	18.429	0.913	0.92447	333.4026	0.0219
KNN (5 neighbour)	137.32856	0.38344	0.3754	121442.29	0.0002

## **Chapter 5. Reference**

1. Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. [online] . Available at:  
<https://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/understanding-machine-learning-theory-algorithms.pdf>.
2. Nabi, J. (2019). *Machine Learning — Fundamentals*. [online] Medium. Available at:  
<https://towardsdatascience.com/machine-learning-basics-part-1-a36d38c7916>  
[Accessed 8 Oct. 2020].
3. www.javatpoint.com. (n.d.). *Machine Learning Random Forest Algorithm - Javatpoint*. [online] Available at:  
<https://www.javatpoint.com/machine-learning-random-forest-algorithm>.
4. Azur, M.J., Stuart, E.A., Frangakis, C. and Leaf, P.J. (2011). Multiple imputation by chained equations: what is it and how does it work? *International Journal of Methods in Psychiatric Research*, 20(1), pp.40–49.
5. Abayomi, K., Gelman, A. and Levy, M. (2008). Diagnostics for multivariate imputations. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 57(3), pp.273–291.

6. Jaadi, Z. (2019). *A Step by Step Explanation of Principal Component Analysis*.  
[online] Built In. Available at:  
<https://builtin.com/data-science/step-step-explanation-principal-component-analysis>.
7. Kenton, W. (2020). *How Multiple Linear Regression Works*. [online]  
Investopedia. Available at: <https://www.investopedia.com/terms/m/mlr.asp>.
8. in.mathworks.com. (n.d.). *Multiple linear regression - MATLAB regress - MathWorks India*. [online] Available at:  
<https://in.mathworks.com/help/stats/regress.html> [Accessed 27 Jun. 2021].

## **Chapter 6. Glossary**

- **Machine Learning**- Machine learning is an artificial intelligence (AI) function that imitates the workings of the human brain in processing data and creating patterns for use in decision making.
- **Neural Network**- A neural network is a series of algorithms that endeavors to recognize underlying relationships in a set of data through a process that mimics the way the human brain operates.
- **Training dataset**- Training data is the majority part of the available dataset that is used to teach a machine learning model.
- **Keras**: Keras is an open-source software library that provides a Python interface for artificial neural networks.
- **Epochs**: One epoch means that each sample in the training dataset has had an opportunity to update the internal model parameters.
- **R<sup>2</sup> score**: In statistics, the coefficient of determination, denoted  $R^2$  or  $r^2$  and pronounced "R squared", is the proportion of the variance in the dependent variable that is predictable from the independent variable(s).