DS 203 ASSIGNMENT: 11

NAME	ROLL NO	DEPARTMENT
AMAN RISHAL CH	22B3914	ELECTRICAL DD
RAHUL B	22B3976	ELECTRICAL DD
HRISHIKESH S	22B4217	ELECTRICAL DD

AN OVERVIEW TO THE PROBLEM

- The given data set is taken from a Chemical plant over a period of 2 years and 9 months
- The dataset consists of a total of 241 parameters,20 controllable and the rest are not
- Aim: To create a ML model so that we can use it to predict the values of the controllable parameters, so that the vibrations of the chemical plant can be kept in control.
- Preferred ML models: Simple Regression, Gaussian, Bayesian, Decision Trees,

  Random Forest, Neural Networks.

AN OVERVIEW TO THE PROBLEM

- The given data is good overall; but some work need to be done on it before utilizing the data to train a Machine Learning model.
- The data is having some blank columns and some missing entries.
- Some values are not in numeric form (#N/A,#REF,etc).
- We have used the Gaussian model to rectify the above mentioned errors (missing values and non numeric text) in the given data as Gaussian model will not bias the given data and will give better approximations to the missing values.
- We have removed the columns that were fully blank
- There is no need to do any standardisation or transformation to the given data as it have been cleaned using the Gaussian, IQR, Moving Average and later; Multicollinearity removal.

AN OVERVIEW TO THE PROBLEM

- It is **not necessary** to train individual models for the parameters c51,c52,c53 and c54 as we can use all of them in the same model itself.

- Efficient ML algorithms such as Neural Networks, Random Forest, Decision Trees can be used to make a model incorporating all the above mentioned parameters.

- The many columns provided can give rise to the condition of multicollinearity; and it is important to remove the columns which are multicollinear before training the ML model.

AN OVERVIEW TO THE PROBLEM

- The model trained without removing the multicollinear parameters will definitely take more time for computation.
- Multicollinearity also undermines the Statistical significance of a dependent variable and can lead to unstable and unreliable coefficient estimates.
- We have used the statistical parameter VIF (Variance Inflation Factor) (VIF > 10) to remove the multicollinear factors from the data after cleaning and smoothening.

INITIAL ANALYSIS OF THE DATA

- Average of Data

Average of c51	Average of c52	Average of c53	Average of c54
9.386950731	9.107621557	9.826082694	9.183064304

- Variance of Data

Var of c51	Var of c52	Var of c53	Var of c54
7.437704804	4.933849772	41.01548842	33.82352756

- Max of Data

Max. of c51	Max. of c52	Max. of c53	Max. of c54
17.69081122	14.93354856	27.3436991	25.95647196

- Min of Data

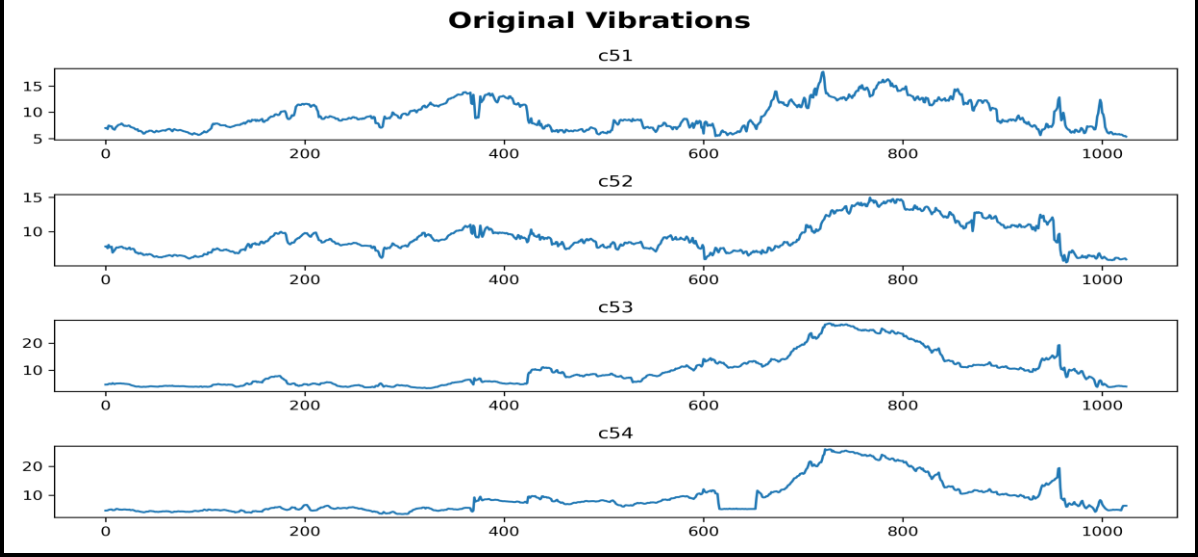
Min. of c51	Min. of c52	Min. of c53	Min. of c54
5.349019912	5.561128462	3.360440328	3.415696138

The Analysis of the data given (for the controllable parameters) shows that the values of only the parameters c53 and c54 have been above 20 (critical stage) initially.

INITIAL ANALYSIS OF THE DATA

Comments about overall Data :

- The first column contains the date in **DD-MM-YYYY** format
- Some columns are having text values like **#REF** and **#N/A**
- Some of the given entries of the columns like c199,c202,c204 are blank ,whereas all the entries of the columns c199, c202,c204,c226,c229 are blank
- The columns c168,c169,c170, and c171 are such that a large part of their entries are of constant magnitude.



Var of c168	Var of c169	Var of c170	Var of c171
2.02061E-14	2.44249E-14	0	1.95399E-14

CLEANING DATA

Removed all the columns which initially had **all** blank entries:- c199, c202, c204, c226, c229

GP	GQ	GR	GS
c198	c199	c200	c201
100.517		2.586091	2.507866
96.94789		2.589318	2.5112
93.48395		2.618084	2.530159
95.8924		2.554376	2.484523
97.37379		2.51867	2.459546
98.47292		2.536975	2.471625
97.62111		2.53167	2.471207

c198	c200	c201
97.16989	2.487639	2.440718
97.97415	2.487766	2.440297
98.40201	2.469735	2.427115
98.66846	2.462269	2.422036
98.46452	2.455048	2.417258
97.95814	2.459595	2.420789

We have used the **Gaussian model** to extrapolate the values of the entries which were initially incorrect (like #REF, #N/A, Blanks)

c188	c189	c190	c191
#REF!	#REF!	#REF!	2.516714
#REF!	#REF!	#REF!	2.507422
#REF!	#REF!	#REF!	2.430812
#REF!	#REF!	#REF!	2.488885
#REF!	#REF!	#REF!	2.531291
#REF!	#REF!	#REF!	2.561872

c188	c189	c190	c191
16.89092	2.253755	6.315017	2.533336
12.58376	2.215944	6.021459	2.546965
12.75744	2.269627	5.555942	2.577342
11.35161	2.341348	5.5453	2.582252
21.12817	2.444164	6.046923	2.586503
23.06395	2.434864	6.64167	2.592952
26.16356	2.389421	6.410926	2.619476

CLEANING DATA

- Cleaning using IQR:

Used IQR to clean the data by finding out Quartiles Q1 and Q2; and then defining IQR to be **Q2-Q1**

Here we have set the lower limit as usual but we have set the upper bound to be higher than the standard upper bound as we are supposed to make a ML model to classify the states of the Chemical plant ; and in doing so we need to consider most of the upper values due to safety reasons.

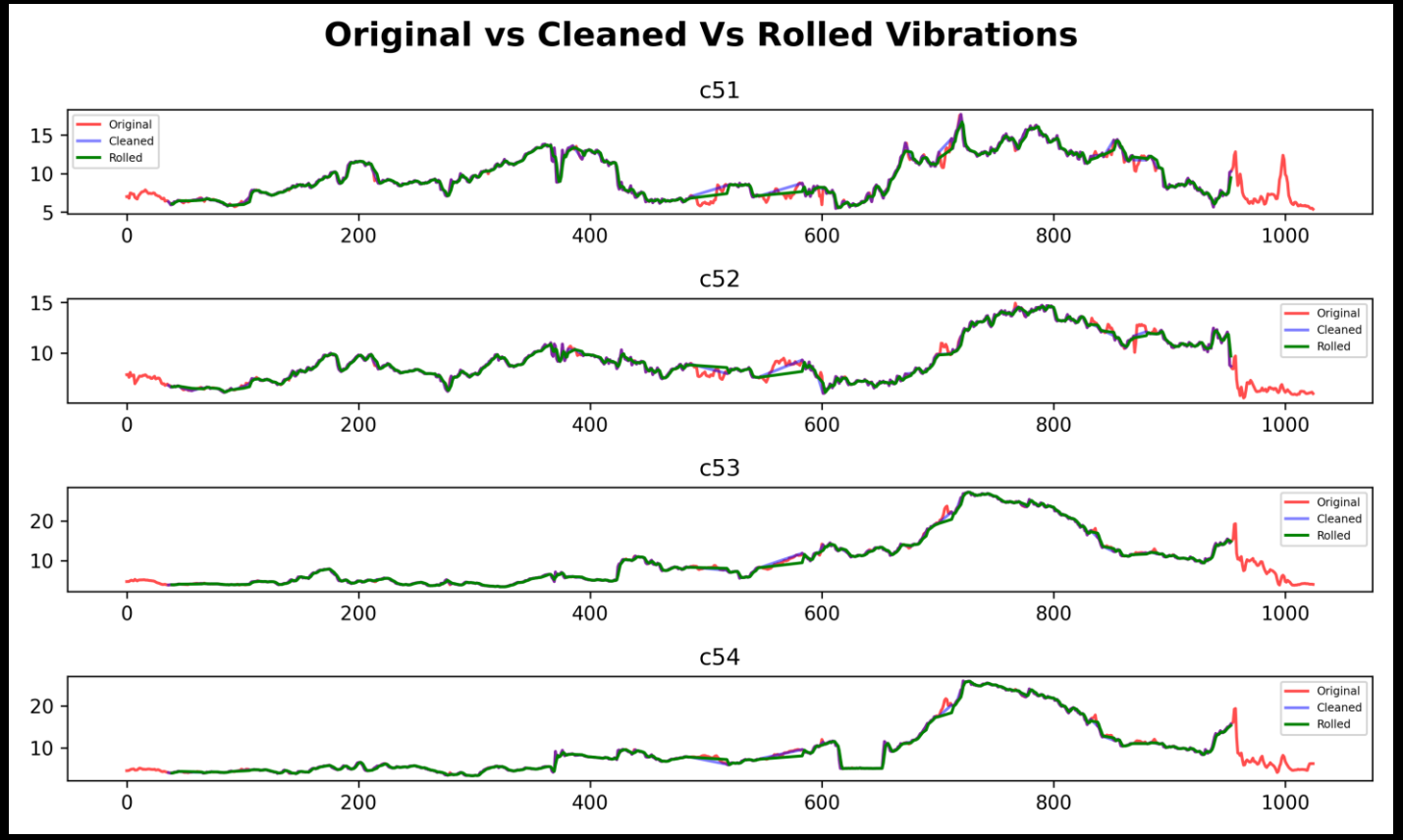
c51	c52	c53	c54
6.981073	7.870521	4.715955	4.625416
7.01398	7.897945	4.732056	4.63741
6.77104	7.609317	4.741603	4.700245
7.453711	8.095649	4.950099	4.85625
7.412639	7.739171	5.060381	4.973109
7.30411	7.781164	4.929154	5.00931
7.319898	7.845785	4.91882	5.040255
6.885967	6.975943	5.267061	4.787791
6.752732	7.151048	5.05981	4.777922
6.664089	7.333466	4.880307	4.809105
7.06607	7.554524	5.063776	5.049722

c51	c52	c53	c54
6.050934	6.693091	3.897615	4.078224
5.991519	6.65284	3.904386	4.080829
6.131923	6.692375	3.897154	4.166603
6.246532	6.697803	3.912271	4.235339
6.322695	6.713214	3.942046	4.282418
6.381808	6.721139	3.991391	4.34557
6.446825	6.745651	4.037169	4.419404
6.447283	6.693647	4.068853	4.453106
6.438777	6.529943	4.072757	4.385244
6.520137	6.475657	4.117821	4.297735
6.579224	6.458615	4.173709	4.172966

ORIGINAL V/S CLEANED DATA

Removed the columns without much variance (Q3 = Q1 approximately) :-
c2, c82, c110, c168, c169, c170, c171
(c156 is a similar column but we have not removed it as it is a controllable parameter)

➢Moving Average:
Used the moving average Algorithm to roll the data of each column with window size 3



Here the data set from the extreme ends of left and right are removed using the data cleaning process mentioned above

MULTIPLE LINEAR REGRESSION

This was the most basic model of all and it did not give any satisfactory results

- Statistical Values obtained:

	c51	c52	c53	c54
R2	0.647	0.780	0.884	0.853
MSE	199.82	155.700	1708.252	1284.36
F-statistic	77.69	138.60	299.46	212.22

As we can see that the value of R2 obtained here is not satisfactory and kills the purpose of training the ML model.

So we need to move on to another model which can give us a good value of R2.

DECISION TREE

This model improved upon the problem of R2.
But it created a new problem of **overfitting** of the model.
Thus we could not obtain any satisfactory results while doing the back-testing.

- Statistical Parameters:

  R^2: 0.971

  Since R^2 is very high, the model fits the data well. But Decision Tree models commonly over-fit data.

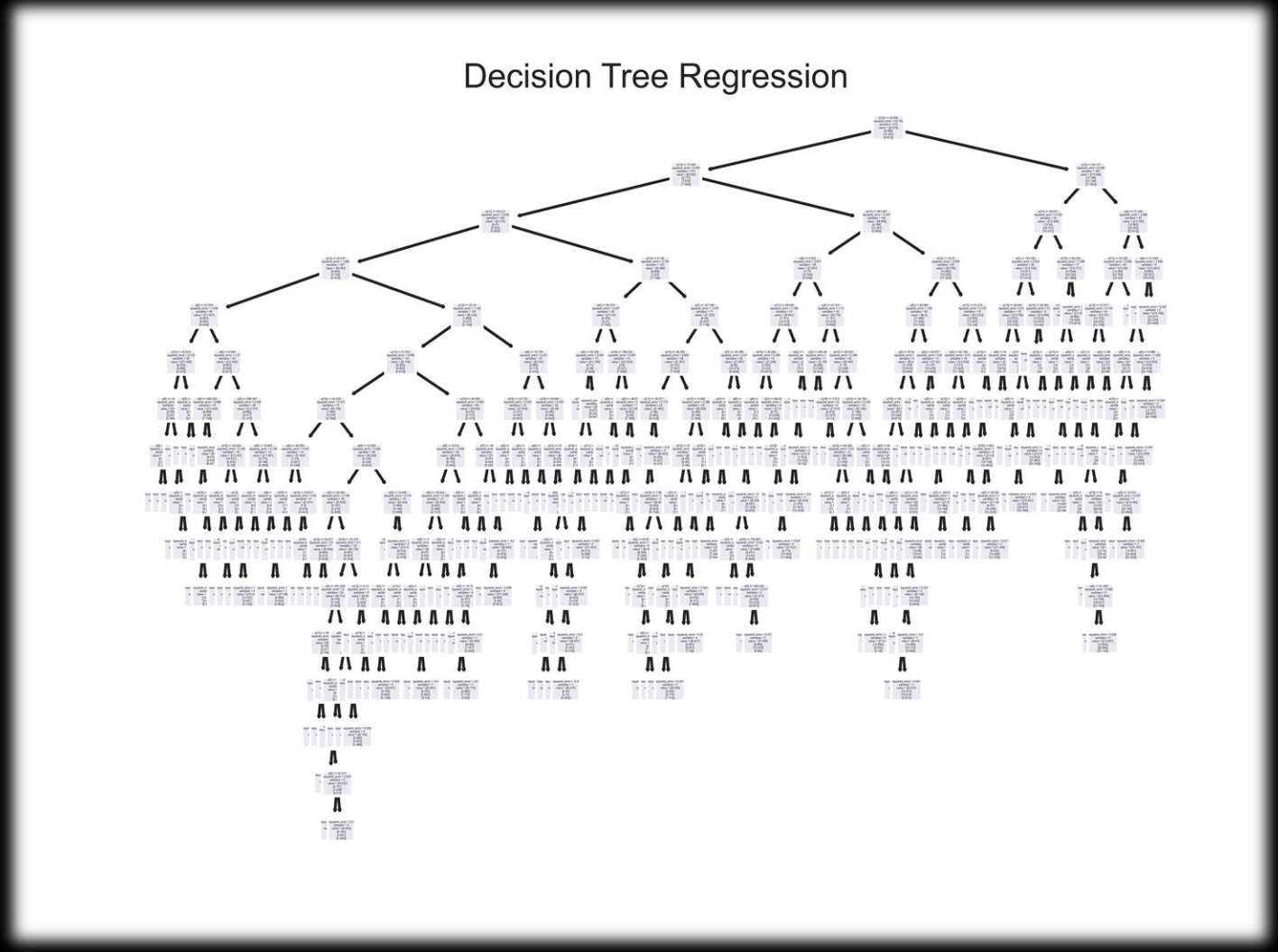
  MSE: 0.380

  Mean Cross Validation Score(NMSE): -10.43

  MCVS is high in magnitude indicating that the model over-fits the data irrespective of training sample selection

DECISION TREE

- The model is **overfitting** and thus we can not use it to predict the future values as it can lead to very poor real life performance while predicting using new data.
- The Decision tree has been modelled by setting the random seed value to be 42.
- K-fold cross validation has been used to asses the performance of the decision tree by setting the value of k to be 5



ELIMINATING MULTICOLLINEARITY

- We have used the Statistical property VIF to eliminate the columns which are Multicollinear
- Used a VIF score of 10 as the upper limit to eliminate multicollinearity
- Elimination of VIF was done sequentially in a Loop. At each iteration, after removing the column with the highest VIF ,we recalculated VIF before further removing columns in the next loop

```
while max(vif_data['VIF'])>10:
    column_to_be_removed_index = vif_data['VIF'].idxmax()
    column_to_be_removed = vif_data['Variable'][column_to_be_removed_index]
    # print(f"Removing column {column_to_be_removed} with vif␣
↪{vif_data['VIF'][column_to_be_removed_index]}")
    removed_multicollinear_columns.append(column_to_be_removed)
    X_new = X_new.drop(column_to_be_removed,axis=1)
    vif_data = vif_data.drop(column_to_be_removed_index,axis=0)
```

- After the process, only 10 columns remained

['c6' , 'c14' , 'c15', 'c36','c113', 'c147', 'c156', 'c184', 'c185', 'c208']

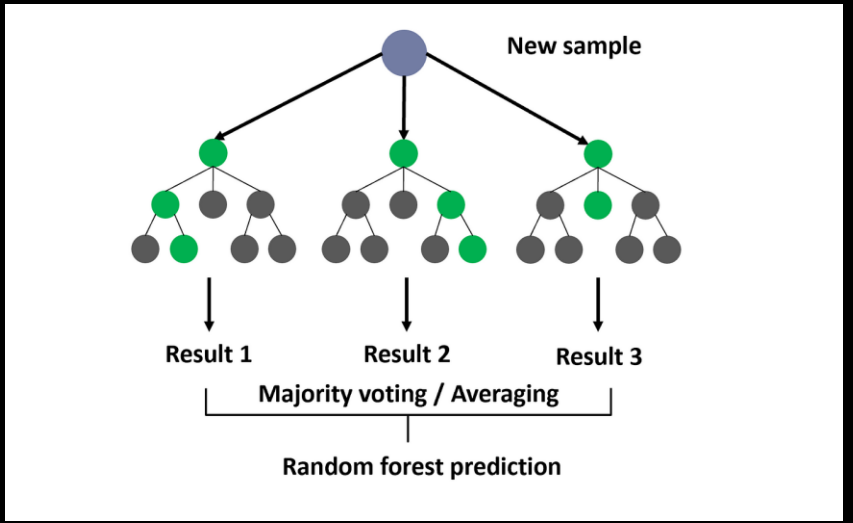
RANDOM FOREST ALGORITHM

- Random Forest Algorithm can be regarded as an improved version of the Decision Trees.
- It utilizes an ensemble of decision trees and then combine their predictions to give overall better results .

This algorithm performed well as it had improved upon the major drawbacks of the previous models:

(i) An increased and statistically satisfactory value of R2

(ii) The problem of overfitting has been taken care of



RANDOM FOREST ALGORITHM

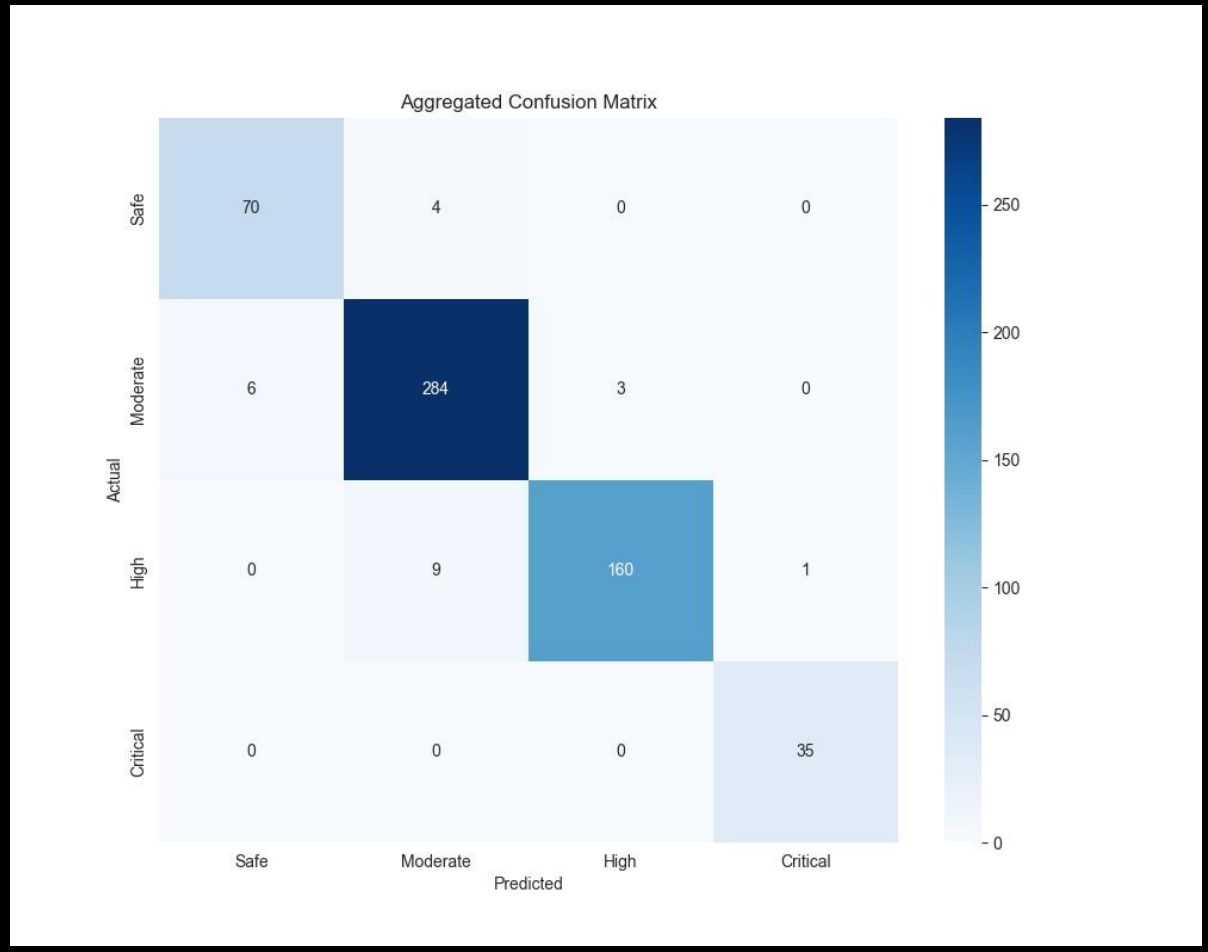
Statistical Parameters:

- R^2:0.985
  Since R^2 is very high, the model fits the data well. Random Forest Algorithms commonly over-fit data.
- MSE: 0.181
  MSE is low indicating that the model is not overfitting .
- Mean Cross Validation Score(NMSE): -1.371
  MCVS is small in magnitude indicating that the model fits the data irrespective of training sample selection
- Out-of-The Bag(OOG) score: 0.985
  Accuracy measured by OOG is high. So model fits the testing set.

RANDOM FOREST ALGORITHM

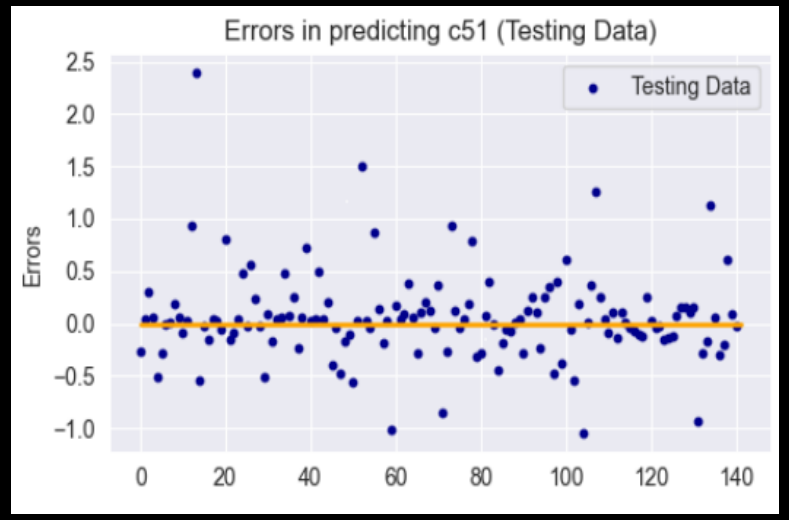
Confusion Matrix

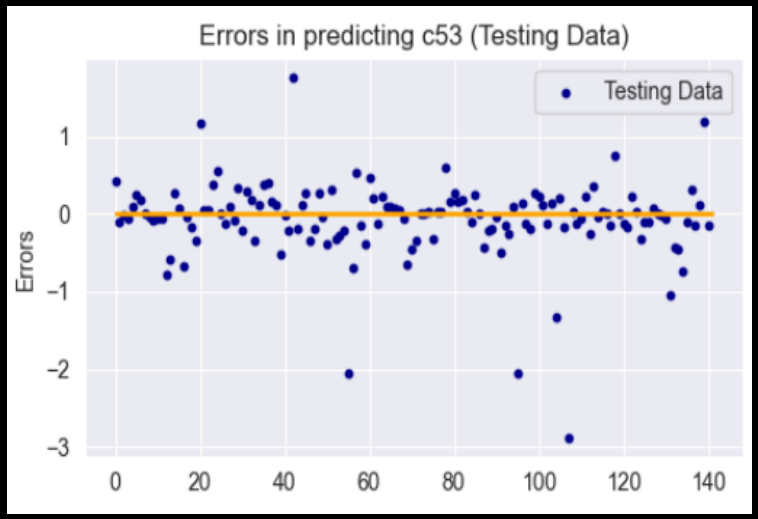
- The model does pretty good in classifying the vibrations with respect to the original data.
- The model will perform great in real time as it has classified **all the critical values** as critical value itself.
- This is extremely important in the context of the given problem.

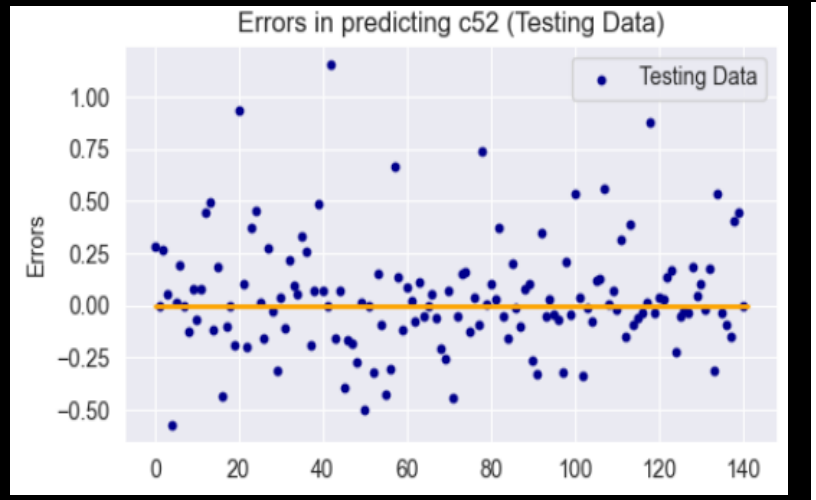


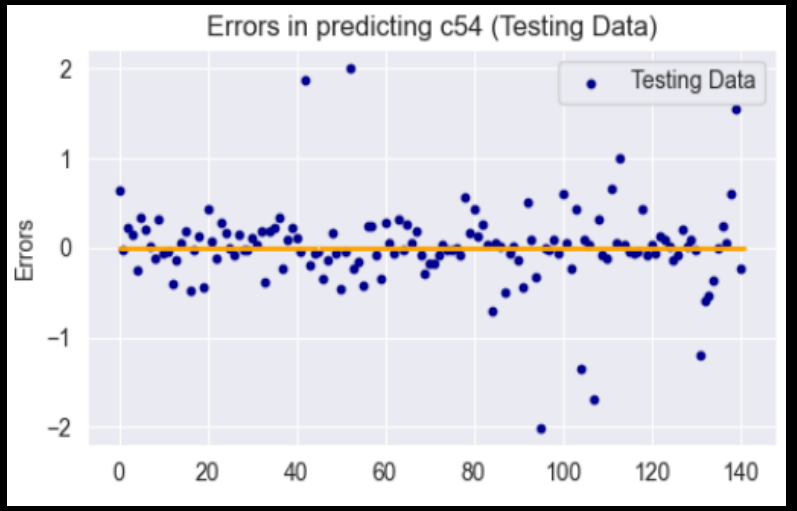
RANDOM FOREST ALGORITHM

- The errors in predicting the columns by the model is distributed randomly for each of the columns
- Thus we can conclude that the model have captured all the essential patterns in the data
- The errors are also distributed symmetrically across the X –axis









FEATURE IMPORTANCE

- We used ***permutation_importance*** to compute Permutation Feature Importance
- It involves random shuffling and measuring the change in model performance after permutation
- The principle is that important features will cause a larger drop in performance when permuted
- Preferred over feature importance because FI is biased towards high cardinality features

FEATURE IMPORTANCE

Feature	Permutation Importance
C155	1.536
C158	0.105
C143	0.075
C161	0.071
C157	0.018
C39	0.017

Feature	Permutation Importance
C31	0.012
C139	0.001
C28	0.001
C26	0.001
C33	0.001
C30	0.000

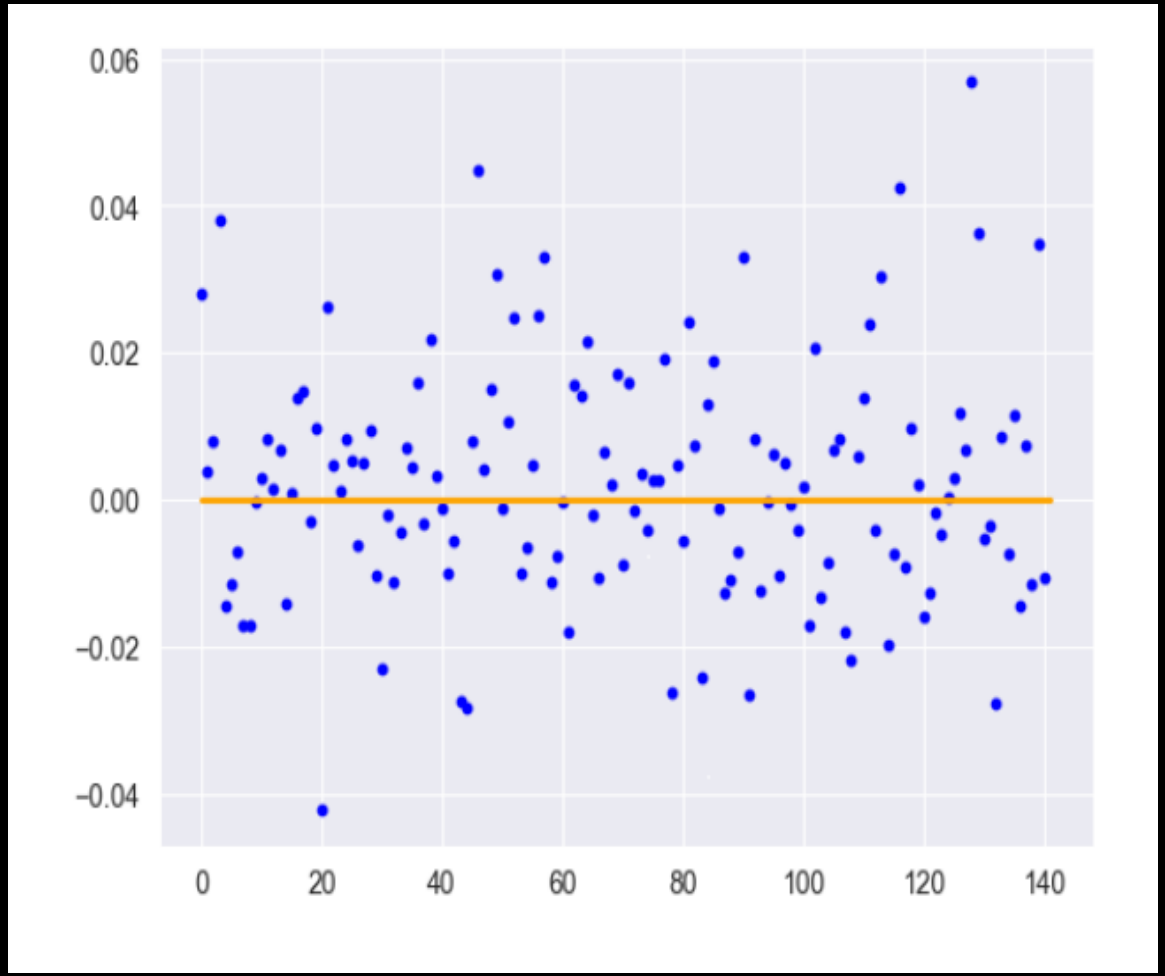
*The above values are rounded-off to 3 decimal places

ANALYSING SPECIFIC ENERGY- C241

- C241 has a mean of 2.18 with a standard deviation of 0.09
- Since it's standard deviation is much smaller than it's mean, the model should accurately model the deviation from the mean.
- MLR can't model such small bidirectional data (results in small R^2)
- Decision tree , as seen before, is probable to overfitting when the values are close.
- Rainforest Regression can fit the data without overfitting by selecting appropriate number of branches (n_estimators)
- Alternate Model which can be used to model C241: Gaussian Regression, SVD

MODELLING SPECIFIC ENERGY- C241

- We used Random Forest Regression to model c241 in terms of all other parameters.
- The Mean Cross Validation Score (5 fold – NMSE) of the model for the entire data is -0.002 indicating the model works regardless of selection of training data
- We split the cleaned data into training (80%) and testing (20%).
- We had $R^2 = 0.960$ and $MSE = 0.0003$ against the test data suggesting that the model didn't underfit or overfit the data.



FEATURE IMPORTANCE OF C241

Feature	Permutation Importance
C143	0.468
C142	0.287
C139	0.025
C31	0.020
C15	0.019
C28	0.019

Feature	Permutation Importance
C158	0.001
C113	0.001
C6	0.001
C155	0.001
C161	0.001
C39	0.000

*The above values are rounded-off to 3 decimal places

FEATURE IMPORTANCE OF C241

- The parameter c143 is having the most important feature which can be used to predict the vibrations to the system.
- An automated vibration controller and reduction system should control the parameters c15,c28, c31,c139,c142 and c143 as per the model.
- We used Permutation Importance instead of Feature Importance to eliminate any bias against features with high cardinality.

DIFFICULTIES FACED

- The data given required some cleaning using a model, and figuring out the most appropriate model to clean the data was a challenge.(Gaussian was chosen at last)
- The problem of Multilinearity was a bit challenging to overcome as the code we made to remove all the multilinear columns required a huge computational power and thus we ended up using the Google Cloud to perform the computation.
- While modelling using random forest regression selection of appropriate value of iterations (n_ estimators) so that model doesn't overfit the data.

ACHIEVEMENTS

Model Of Vibrations using Controllable and Operating Parameter

- The value of R^2 obtained is very much close to one and we have also removed any high bias/high variance.
- Removed multicollinear columns to finally get 10 independent variables from over 200 variables
- Model correctly predicts vibrations which can be seen through low MSE value
- Used only one model for all four vibrations resulting leading to easier analysis of the model and predictions

ACHIEVEMENTS

Model Of Vibrations using Controllable Parameter and their Classification

- The model accurately classified most of the high and critical values in high and critical itself
- The model classified all the critical data to be in critical itself ,that is, it is having **100% accuracy** in classifying the **critical values**
- The model is excellent to be deployed in real time as it is having an overall accuracy of 96%
- Also since the error in more than 99% of the cases lies below 0.5, it is advisable to change the critical value to 19.5 instead of 20 as it will greatly improve the security.

ACHIEVEMENTS

Model Of Specific Energy

- Removed multicollinear columns to finally get 10 independent variables from over 200 variables which increased accuracy of the model while reducing computation
- Created highly accurate prediction model with high R^2 and low MSE value

THANK YOU