UNSW

School of Computer Science & Engineering,

Faculty of Engineering University of New South Wales

Africa Soil Property Prediction

(A Kaggle Research Prediction Competition)

To predict physical and chemical properties of soil using spectral measurements

See website link - here

COMP9417: Machine Learning and Data Mining

Group Project

Student ID	Student Name
z5185318	Chukwuemeka Eze
z5353088	Hanbo Jiang
z5291577	Yunyi Li
z5328237	Zhaohui Xu
z5226354	Zihao Yang

Table of Contents

Section 1: Introduction

1.1 - Problem Description

In order to plan for sustainable agricultural programs and natural resources management in the geographical area of data sparse Africa, a digital mapping of soil properties is required. In view of this and in contrast to using the conventional reference tests, which are slow and expensive, a low cost analysis of soil samples obtained from infrared spectroscopy plus georeferencing of soil samples and greater availability of earth remote sensing data, a predicted results could be obtained of soil functional properties (Ca, P, pH, SOC, Sand) at unsampled locations. Soil functional properties are those properties related to a soil's capacity to support essential ecosystem services such as primary productivity, nutrient and water retention, and resistance to soil erosion (JCstat et al., 2014).

The problem was to predict some soil properties (Ca, P, pH, SOC, Sand) based on the Near Infrared (NIR) data. Spectral features and spatial features were present for analysis and training. There were 1158 instances in train data and 728 instances in test data with 3578 spectral features and 16 spatial features.

1.2 - Motivation for the Project

This project was undertaken to advance our knowledge and learning experience. At completion, we will compare our results with those in the leaderboard of the kaggle competition to see how well we have been able to learn and apply our knowledge.

1.3 - Project Goals

The goal of this project is to predict 5 target soil functional properties (Ca, P, pH, SOC and Sand) from a set of diffuse reflectance infrared spectroscopy measurements. By applying machine learning models and techniques we aim to solve the problem by closely predicting these five soil properties from the data provided. And using appropriate evaluation metrics, provided in the competition is Mean column-wise root mean square error (MCRMSE), to evaluate and compare results of the models and select the best sets with the lowest RMSE.

The remainder of this report presents our findings and outcomes of applying predictive machine learning models to soil samples obtained and made available in a kegel's AfSIS - Africa Soil Information Service competition of 2014 to address the problems discussed

above. The remaining five sections of this report are organized as follows. In section 2 we explore the data provided using some data preprocessing techniques. Section 3 discusses the feature selection methods. Section 4 discusses the modeling approach while in section 5, we discuss the results following models evaluation. Section 6, discusses other approach that could improve our model prediction.

1.4 - Analysis of Existing Literature & Solutions

One of the most common supervised machine learning problems is the multivariate/multiple regression problem, in which values in most cases, continuous, which explains the relationship between multiple independent or predictor variables and one dependent or criterion variable. A dependent variable is modeled as a function of several independent variables with corresponding coefficients, along with the constant term. Multiple regression requires two or more predictor variables, and this is why it is called multiple regression.

General form: $Y_i = f(X_i, \beta) + e_i$

Expanded form: $yi = \beta 0 + \beta 1 xi1 + \beta 2 xi2 + ... + \beta pxip + \epsilon$; **for** $i = 1$ *to* n **observations**

where, yi = dependent variable

xi =explanatory variables

$\beta 0$ = y-intercept (constant term)

βp =slope coefficients for each explanatory variable

ϵ = the model's error term (also known as the residuals)

The report of the competition's winning solution outlined a unique but time costly approach in which a total of 59 models were trained with an average of 8 to 10 combined to predict each of the five target variables. Different feature engineering techniques were applied to reduce dimensionality, transform the independent variables and remove outliers. One of such transformations was log transformation of log(P+1) on the 'P' target variable.

Third ranked solution's approach was different from the winning solution but more simple with a different model each for each target value. However, similar transformation such as discrete wavelet transformation of subinterval of was applied to smoothen the variables and remove noise. In our solution, we adopted a more simpler approach; we trained several similar models for each of the target variables and then selected the best set of models that gives the lowest MCRMSE or average RMSE value.

Section 2: Data, Preprocessing and Transformation

2.1 - Data Source and Collection Method

The dataset as presented in kaggle, was sourced from measurements of diffuse reflectance infrared spectroscopy plus georeferencing of soil samples. The amount of light absorbed by a soil sample is measured at hundreds of specific wavebands across a range of wavelengths to provide an infrared spectrum. Conventional reference soil tests are calibrated to the infrared spectra on a subset of samples selected to span the diversity in soils in a given target geographical area. The calibration models are then used to predict the soil test values for the whole sample set. The predicted soil test values from georeferenced soil samples can in turn be calibrated to remote sensing covariates, which are recorded for every pixel at a fixed spatial resolution in an area, and the calibration model is then used to predict the soil test values for each pixel. The result is a digital map of the soil properties. There were 1158 instances in train data and 728 instances in test data with 3578 spectral features and 16 spatial features.

2.2 - Preprocessing

In our Exploratory Data Analysis, we observed that ‘‘Ca’ and ‘P’ are highly skewed (see appendix 2.2). Compared to ‘SOC’ . Further analysis suggests significant evidence of outliers in the response variable distributions. So we applied some techniques detailed in the data cleaning section below to clean the dataset.

2.2.1 **-** Data Cleaning

Outliers - Exploratory Data Analysis (EDA) reveals outliers, so we used interquartile range (IQR), the best suited method to identify extreme values and outliers from the samples in a skewed distribution. Outliers are predominant in ‘P’ and ‘Ca', See Fig 2.2.1.

Null values - We checked for null values which are intended to be replaced by the column mean. However the dataset does not have null values.

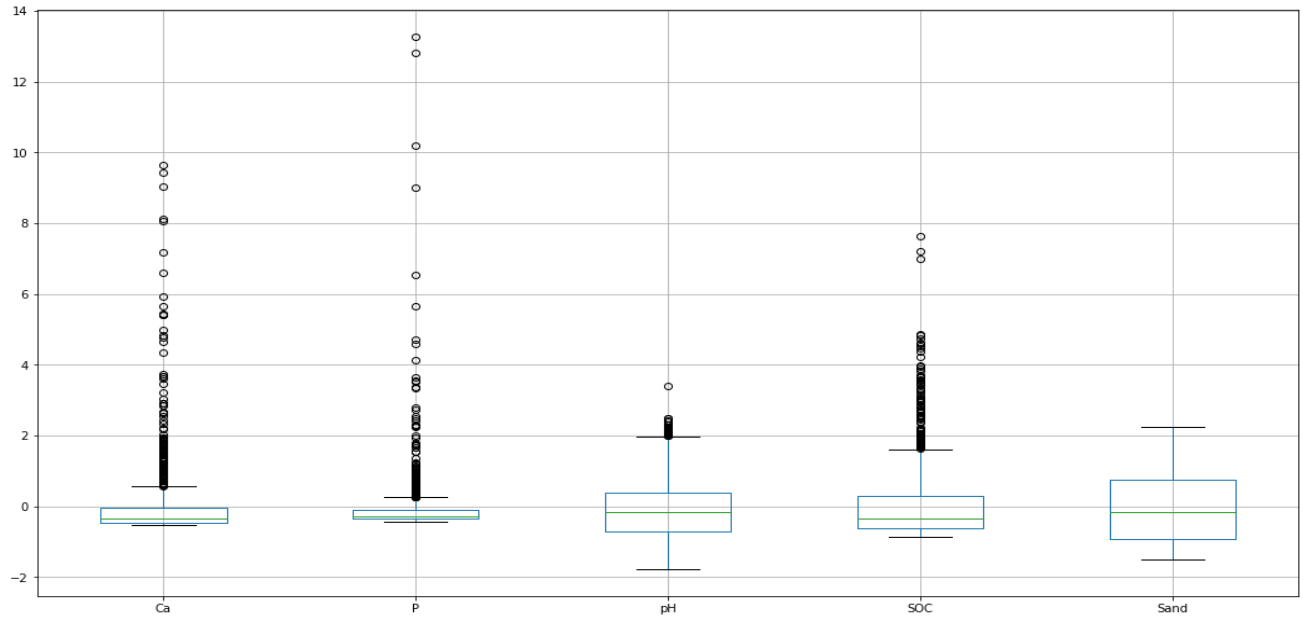


Fig 2.2.1: Boxplot showing outliers in the response variables, especially in 'Ca', 'P' and 'SOC'.

2.3 - Feature Engineering

2.3.1 - Normalization

The purpose of data normalization is to unify data from different sources under the same order of magnitude, which has two major advantages:

(1) normalization improves the speed of solving the optimal solution by gradient descent

(2) Normalization has the potential to improve accuracy

However, the training data provided was already mean centered and scaled, so normalization was not needed.

2.3.2 - Train Test Split and Standardization

It is important to specify an appropriate ratio between training set and test set due to the future potential implementation of the model. What is more, we choose to divide the dataset by row count in that there exists a single sample ID and complete information of this sample in each row. Specifically, we splitted the provided training data into training set and validation set with the 4:1 ratio, with the provided test set prepared for the final test of the tuned model.

Section 3: Feature Selection Methods

The dataset has five response variable, unique soil sample identifier, 3595 independent numerical features comprising spectral and spatial measurements and one categorical feature, depth; with 2 categories: "Topsoil", "Subsoil", that groups the samples into Topsoil and subsoil. Thus, it was dropped for this problem. The response variables were separated from the dataset so we have X (features space) and Y(response variables). To determine important features necessary to predict each of the response variables as well as improve efficiency of training models, further analysis was conducted using two important techniques; Principal component Analysis (PCA) and Correlation Analysis.

3.1 - Principal Component Analysis

Due to the highly dimensional feature space (3590) we fear the problem of curse of dimensionality may occur. So, we applied a common linear reduction technique - Principal Components Analysis (PCA) to transform the original training dataset which yielded 16 principal components.

Although in the evaluation, it shows that there was insignificantly poor outcome in the overall prediction using a PCA reduce feature space to train the models compared the outcome for correlation analysis on the feature space.

3.2 – Correlation Analysis

Fig 3.1 Shows the level of correlation that exists between predictors and response variables, and table 3.1, for each response variable, the number of predictors with correlation coefficient threshold of 0.1 for pearson and spearman methods.

Table 3.1: Number of Features with correlation coefficient >= 0.1

Correlation Methods	**Ca**	**P**	**pH**	**SOC**	**Sand**	**Performance(sec)**
Pearson	3536	885	1323	3563	3535	142.91
Spearman	3520	1473	1528	3562	3534	108.45

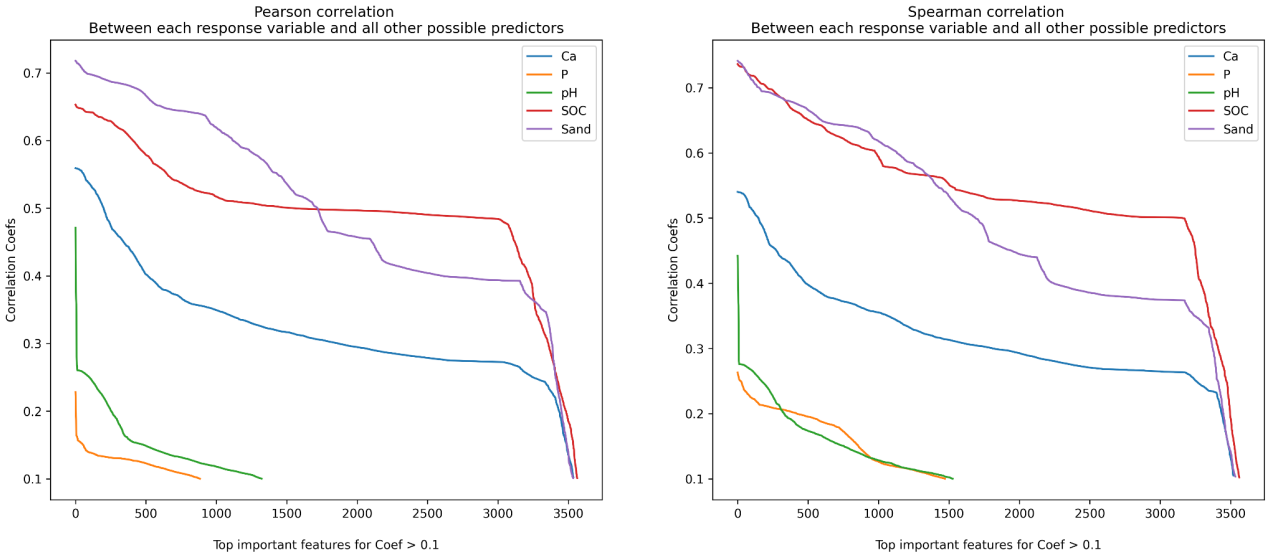


Fig 3.1: Correlation between independent variables and response variables (Left- pearson, Right- spearman)

Section 4: Model Training

4.1 - Approach to Modelling

The problem was a Multivariate Regression with Multiple Dependent Variables. However the relationship between the explanatory variables and response variables were not completely linear. So, we adopted two modeling approaches. First, EDA shows that not all explanatory variables were linearly correlated to response variables. So, we build a base model to see how the explanatory variables would predict each of the response variables. Secondly, we used grid search with k-fold cross-validation on the training set to tune hyperparameters for each of the candidate models. Four model algorithms were built this way and evaluated. We further use the neural network model given its robustness in the training model with high dimensional feature space. Models that produced the best outcome on the validation sets for each of the response variables were chosen for predicting the test-set. The competitions provided a separate test-set but there was no way to evaluate our model on this test-set because their response values were not made available for evaluation purposes. So we split our training dataset according to section 2, subsection 2.2.4

4.1.1 – Gradient Boosting Regression (GBR)

With Gradient Boosting Regression, an ensemble of decision tree models built sequentially,

each tree is trained to correct the errors of the previous trees. It optimizes a loss function, in this case, absolute loss, using gradient descent to iteratively improve the model's predictions. Table 4.1.1 below highlights the search space and selected optimum values

Table 4.1.1: Gradient Boost hyperparameter tuning.

Parameter name	Rationale and impact	Search space	Optimal value selected				
			Ca	P	pH	SOC	Sand
n_estimators	Decision trees that determines the total number of base weak learners	(50, 100, 200)	50	200	200	200	200
learning_rate	determines the step size or shrinkage factor for the base model	(0.01, 0.1, 1.0)	1.0	0.1	0.1	0.1	0.1
max_depth	Max number of levels a decision tree can have in the ensemble, determines complexity	(2, 3, 5, 7)	3	7	7	7	7

4.1.2 – Support Vector Regression (SVR)

This machine learning algorithm aims to find the optimal hyperplane to minimize the prediction error. It uses support vectors (data points closest to the hyperplane), to define the

regression line. The equation for SVR is: y = w * x + b

where y is the predicted output, w is the weight vector, x is the input feature vector, and b is the bias term. SVR seeks to find the optimal values for w and b to achieve the best regression performance. Table 4.1.2 shows the hyperparameters optimized and used in the algorithm.

Table 4.1.2: SVM hyperparameter tuning.

Parameter name	Rationale and impact	Search space	Optimal value selected				
			Ca	P	pH	SOC	Sand
C	Regularization parameter	(1000.0, 10000.0, 15000.0)	15000.0	15000.0	15000.0	10000.0	10000.0
kernel	kernel function used for transforming the input data into a higher-dimensional space	('linear', 'rbf', 'poly')	poly	poly	rbf	rbf	poly
degree	It specifies the degree of the kernel function used	(2, 3)	3	3	2	2	2
coef0	A constant term, it controls the bias in the decision function	(0.01, 0.05, 0.1)	0.1	0.1	0.01	0.01	0.1

4.1.3 Neural Network Regression (NNR)

Neural Network Regression is a type of machine learning model that uses artificial neural

networks to predict continuous target variables. It involves training a network with multiple hidden layers to learn complex patterns and make accurate predictions based on input features. Table 4.1.3 below shows the hyperparameters selected from after optimization run.

Table 4.1.3: NN Regression hyperparameter optimization

Parameter name	Rationale and impact	Search space	Optimal value selected				
			Ca	P	pH	SOC	Sand
Learning Rate	Controls step size and affects convergence speed	0.0001, 0.001, 0.01	0.001	0.001	0.001	0.001	0.001
Dropout Rate	It's to regularize the model to prevent overfitting.	0.1, 0.2, 0.3	0.2	0.2	0.2	0.2	0.2
Optimiser	Determines how model weights are updated during training.	SGD, Adam	SGD	SGD	SGD	SGD	SGD
Batch Size	Affects convergence speed and generalization.	32, 64, 128	32	32	32	32	32

4.1.4 –Elastic Net Regression (ENR)

Elastic Net is a hybrid regularization technique that combines the properties of both Lasso and Ridge Regression. It adds both L1 and L2 penalty terms to the linear regression objective function, controlled by two tuning parameters, lambda (λ) and alpha (α). Elastic Net combines the advantages of Lasso's feature selection capabilities and Ridge's stability in the presence of multicollinearity, making it suitable for handling datasets with correlated features.

Table 4.1.4: Elastic Net hyperparameter tuning.

Parameter name	Rationale and impact	Search space	Optimal value selected				
			Ca	P	pH	SOC	Sand
alpha	Constant that multiplies the L1 term, controlling regularization strength	(0.00001, 0.0001, 0.001, 0.005, 0.01, 0.1, 0.3, 0.5, 0.7, 0.9, 1, 5, 10)	0.00001	0.00001	0.00001	0.001	0.0001
max_iter	The maximum number of iterations.	(2000, 5000, 7000)	2000	5000	5000	5000	2000
L1_ratio	The ElasticNet mixing parameter	(0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1)	0.4	0.5	0.5	0.4	0.5

4.1.5 – Random Forest Regression (RFR)

Random Forest Regression combines multiple decision trees to make predictions in regression tasks. It works by constructing a forest of trees, each trained on a random subset of the data, and then averaging their predictions to obtain the final prediction. The equation for Random Forest Regression is: $y = \Sigma(wi * yi) / \Sigma wi$

where y is the predicted output, wi is the weight of the ith tree, and yi is the prediction of the ith tree. Table 4.1.5 below shows tunable hyperparameters optimized for best outcome for each response variable predictions

Table 4.1.5: Random Forest hyperparameter optimization

Parameter name	Rationale and impact	Search space	Optimal value selected				
			Ca	P	pH	SOC	Sand

n_estimator	Number of trees in the forest.	10,50,100,200	10	100	100	300	100
max_depth	Maximum depth of the tree	5,10,15,20,25	15	N.A.	15	25	20
min_samples_leaf	Minimum number of samples at an internal node	2,5,10,20	1	5	1	1	2
min_samples_split	Minimum number of samples at a leaf node	1,2,5,10	5	5	2	2	5
max_features	Number of features to consider.	None, 'sqrt', 'log2'	“sqrt”	“sqrt”	N.A.	“sqrt”	N.A.

4.2 - Feature Importance

See Appendices 4.2.1 to 4.2.4 showing tables for top 10 important features for individual model’s response variables.

Section 5: Model Evaluation & Discussion

5.1 - Model Evaluation

As requested by the competition host, Root Mean Squared Error (RMSE) is the square root of Mean Square Error (MSE), which provides a measure of the average error in the same units as the response variables. **MCRMSE** computes the column-wise average RMSEs of all the response variables, with lowest value indicating the best prediction. Tables 5.1 and 5.2 present best performances respectively for train and test evaluations among all the prediction

models fitted. boldly highlighted values indicate the best prediction for each response variable and for MCRMSE in each model.

Table 5.1: Train Evaluation

Regressor	**RMSE**					**MCRMSE**
	Ca	**P**	**pH**	**SOC**	**Sand**	
GBR	**0.01004**	**0.00047**	**0.00135**	**0.00014**	**0.00054**	**0.00251**
SVR	0.06513	0.07806	0.12592	0.08500	0.19994	0.11081
NNR	0.3722	0.3699	0.3743	0.3706	0.3751	0.3724
ENR	0.1087	0.1105	0.1982	0.0762	0.0920	0.1425
RFR	0.046	0.047	0.164	0.104	0.174	0.107

Table 5.2: Test Evaluation

Regressor	**RMSE**					**MCRMSE**
	Ca	**P**	**pH**	**SOC**	**Sand**	
GBR	0.15575	0.13134	0.47848	0.30700	0.35178	0.28487
SVR	**0.11950**	0.17374	0.36185	**0.21777**	**0.32596**	**0.23976**
NNR	0.4165	0.4231	0.4098	0.4428	0.3988	0.4182
ENR	0.2301	0.1987	**0.2954**	0.3600	0.4166	0.3711
RFR	0.137	**0.1240**	0.498	0.337	0.422	0.304
Final Prediction Errors	**0.11950**	**0.1240**	**0.2954**	**0.21777**	**0.32596**	**0.216526**

See appendix 5.1 for more information on evaluation performances of models on both full and reduced feature space (PCA and Correlation Analysis)

5.2 - Discussion of Best Models

We could see in the test evaluation in table 5.2, Support Vector Regressor (SVR) performs best in predicting target variables ‘Ca’, ‘SOC’ and ‘Sand ', Random Forest Regressor (RFR) performs best for ‘P’ and Elastic Net Regressor (ENR), best for ‘pH’. So we choose these three models for our final predictions for each of the response variables they performed best on. Their average RMSE (i.e MCRMSE) is **0.216526.** This result shows an improved result compared to the competition’s winning score on the kaggle leaderboard. Remember, our models were trained on the training set only with 20% set aside for final test evaluation. We could not carry out our final evaluation on the competition’s provided test dataset ‘sorted_test.csv’ since there were no response variables on this dataset to compare our models' predictions with.

We observed the following.

1. Tree-based models tend to overfit. When you compare results of train errors for the two tree-based models GBR and RFR, Both show very low training errors compared to others. However these were not reflective in their test errors. Although it could be argued that a wider search space may allow the hyperparameter tuning algorithm to select even far optimal hyperparameter values to generalize well.
2. Outliers have a huge effect on the final outcome of regression models as we observed that when our SVR model was trained on the original dataset without removing the outliers, the MCRMSE value was **0.4487**. However, the value dropped to **0.2576** as values for these outliers were replaced by the upper and lower band and **0.2397** when they were completely dropped.
3. We observed that feature reduction did not necessarily improve the model, but increase efficiency in model training. Properly cleaned data improves the accuracy of prediction. As shown in both train and test evaluation, training our models on the original feature space show slightly (but insignificant) better prediction than in the reduced feature space see appendix 5.1.

Section 6: Extensions

A closer look at the response variables shows that there is a common relationship that may exist among some groups (**‘Ca’ and ‘pH’**, **‘SOC’ and ‘Sand’,** and then **‘Ca’ and ‘SOC’**) out of the five response variables. see figure 6.0 below. We also believe that considering these

relationships in our modeling could improve the accuracy of our prediction according to (Lee W, et al., 2012). Prediction accuracy could be improved when the response variables with common structure are modeled jointly. We shall consider this suggestion to further improve our model to achieve the best prediction possible.

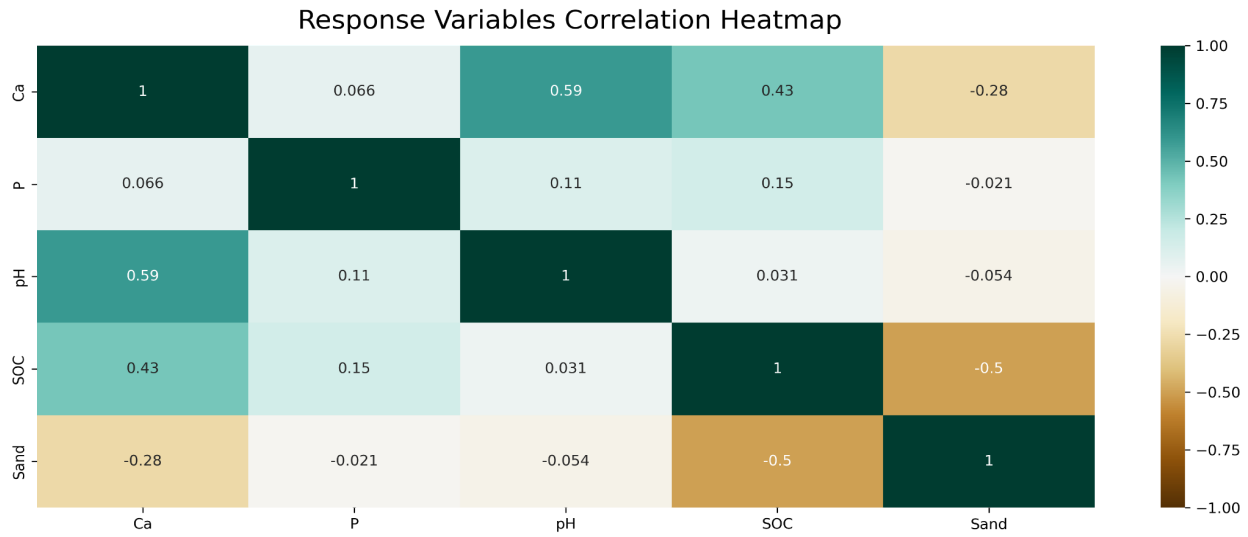


Fig 6.0: Correlation, showing the relationship that exist amongst response variables

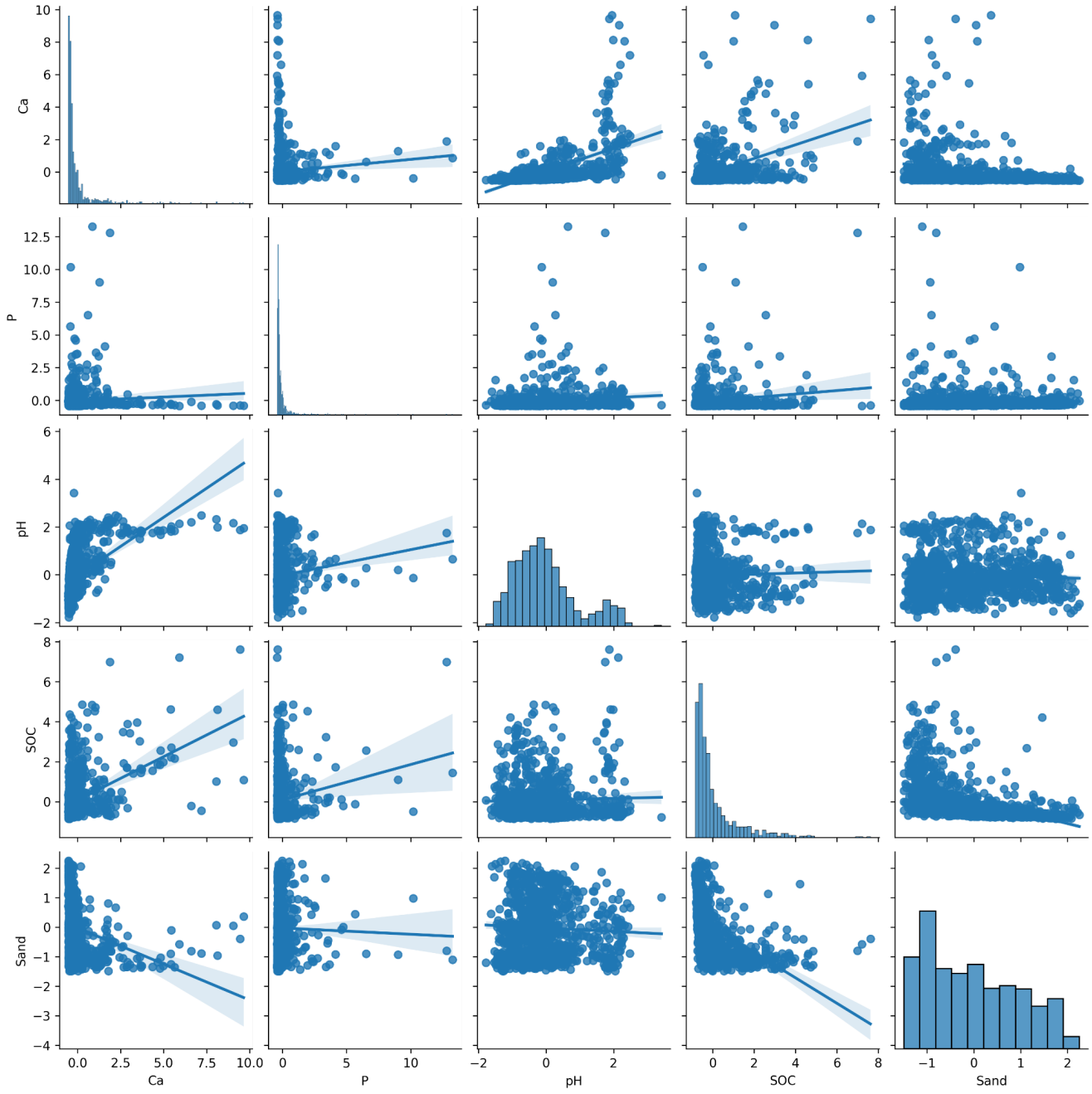
References

1. JCstat, Joyce, Shepherd, K., & Walsh, M. (2014). *Africa Soil Property Prediction Challenge*. Kaggle. https://kaggle.com/competitions/afsis-soil-properties

2. Lee W, Du Y, Sun W, Hayes DN, Liu Y. (2012). *Multiple Response Regression for Gaussian Mixture Models with Known Labels*. Stat Anal Data Min. 2012 Dec 1

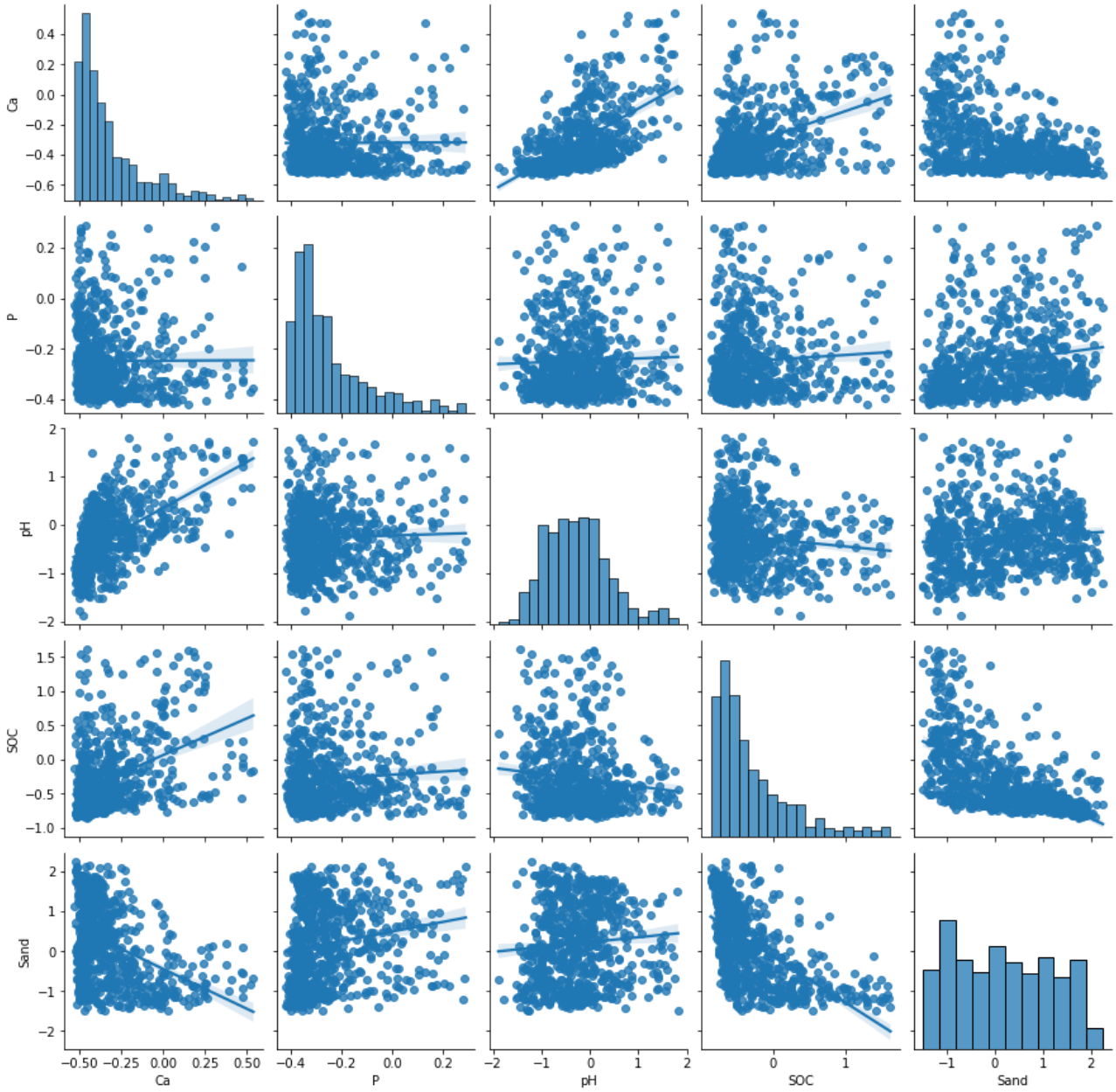
Appendices

Appendix 2.2.1: A pairplot of Response variables before applying data cleaning

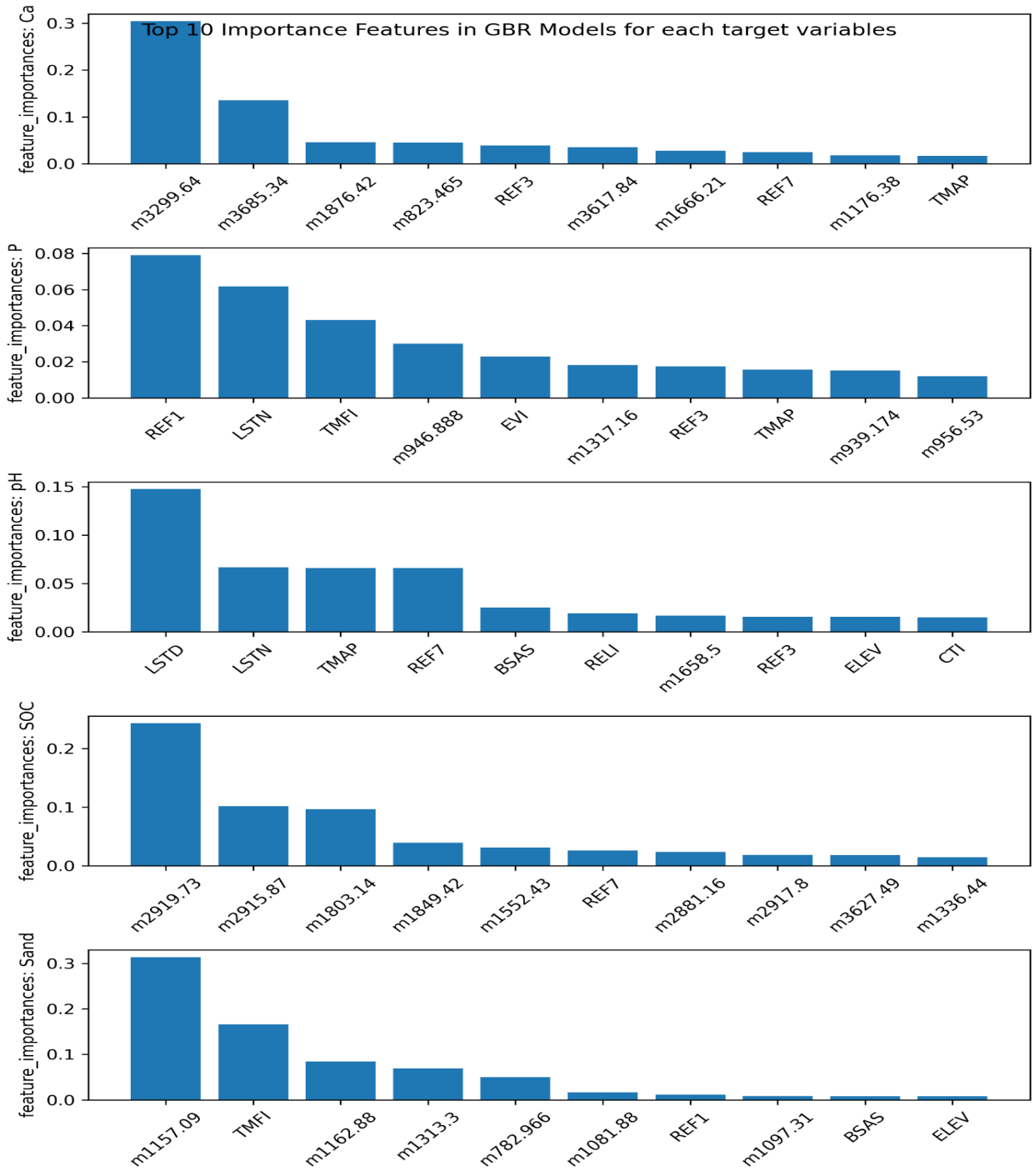


This plot shows how highly skewed 'Ca', 'P', 'SOC' compare to other response variables. It further highlighted some linear relationship that exist between them

Appendix 2.2.2: A pairplot of Response variables After applying data cleaning

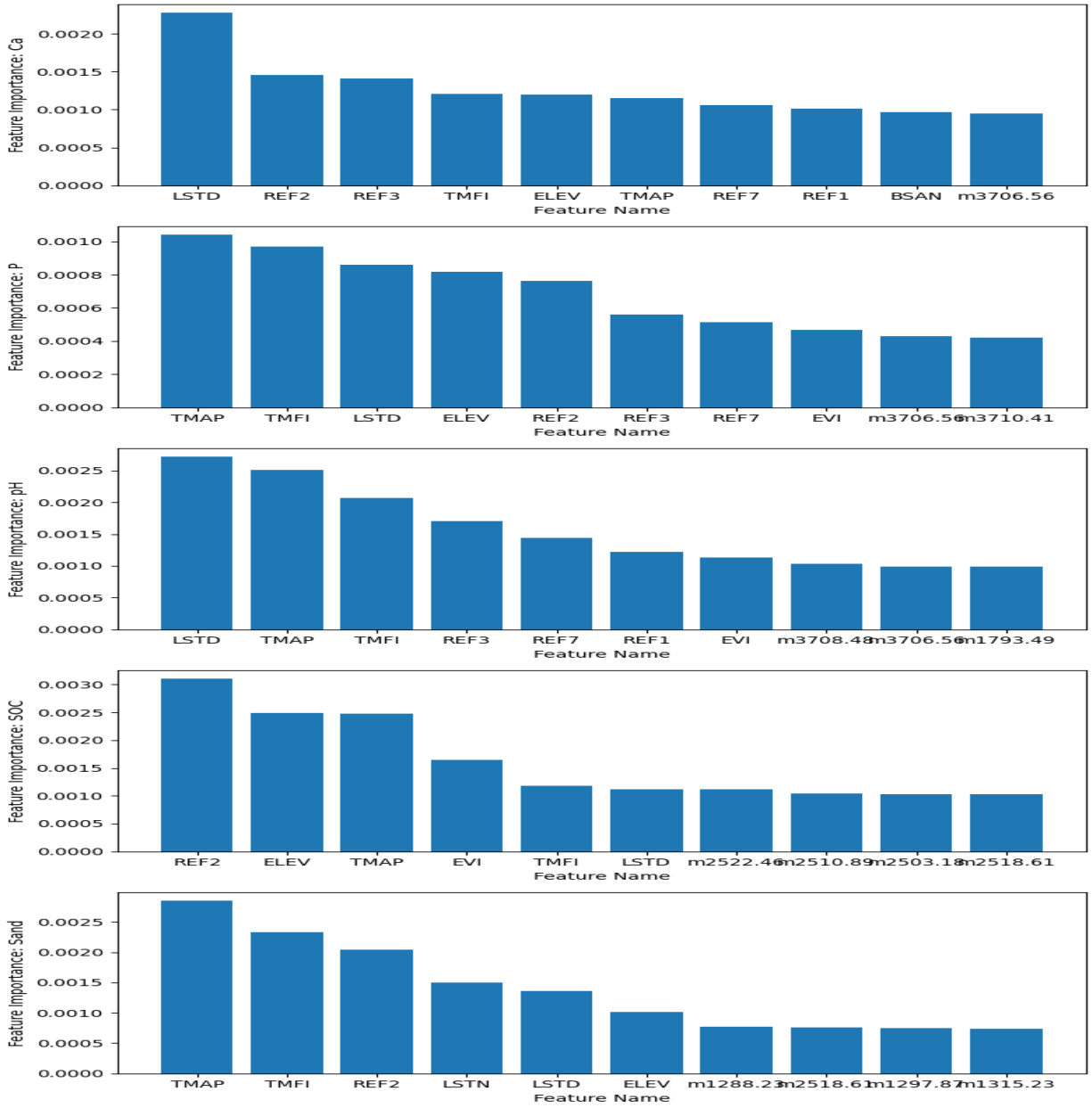


Appendix 4.1.1: GBR feature Importance For Each Response Variable



Showing Feature importance for each of the response variable for Gradient Boosting Regression

Appendix 4.1.2: NNR feature Importance For Each Response Variable



Appendix 4.2.1 - Top 10 important features by GBR model' Response Variables

Ca	P	pH	SOC	Sand
m3299.64	REF1	LSTD	m2919.73	m1157.09
m3685.34	LSTN	LSTN	m2915.87	TMFI
m1876.42	TMFI	TMAP	m1803.14	m1162.88
m823.465	m946.888	REF7	m1849.42	m1313.3
REF3	EVI	BSAS	m1552.43	m782.966
m3617.84	m1317.16	RELI	REF7	m1081.88
m166.21	REF3	m1658.5	m2881.16	REF1
REF7	TMAP	REF3	m2917.8	m1097.31
m1176.38	m939.174	ELEV	m3627.49	BSAS
TMAP	m956.53	CTI	m1336.44	ELEV

The feature importance for each model can be found in the Appendix.

Appendix 4.2.2 - Top 10 important features by NNR model' Response Variables

Ca	P	pH	SOC	Sand
m6738.14	m6842.45	m7096.24	m6813.78	m7075.36
m7247.26	m6995.13	m6784.12	m6753.56	m7298.44
m6967.63	m7123.14	m7197.48	m7286.17	m7169.51
m6882.78	m7007.24	m6951.19	m7147.82	m7021.29
m6821.45	m7213.33	m6866.33	m6892.11	m6789.16
m6971.23	m7151.36	m7205.72	m7028.41	m7234.08
m7112.17	m7310.47	m6942.15	m7106.95	m6979.15
m7188.46	m7052.22	m7221.84	m7269.73	m6917.23
m7202.19	m6989.11	m6718.39	m6935.27	m7048.57

m7301.83	m7264.97	m7037.91	m6982.74	m6732.47

Appendix 4.2.3 - Top 10 important features by Elastic Net model' Response Variables

Ca	P	pH	SOC	Sand
m3230.15	REF3	RELI	REF3	m3984.43
m3398.19	TMAP	m2395.14	m2573.81	REF7
m2155.78	CTI	m1355.44	m1245.11	m3391.92
m2504.88	ELEV	m1149.19	LSTD	m1271.58
m2929.13	m1169.32	m1996.23	m2559.37	m2249.69
m1145.27	m986.341	m1996.23	m2034.73	m3316.53
m4591.22	m1237.21	TMAP	m3674.63	m3659.73
m1721.19	m3528.14	m1985.85	m2214.32	m2349.67
m2491.23	m3312.78	LSTD	m1303.43	m2067.80
m1113.19	m2512.98	CTI	m3668.36	m2603.24

Appendix 4.2.4 - Top 10 important features by RF model' Response Variables

Ca	P	pH	SOC	Sand
m1297.87	m1806.99	m1806.99	REF7	m1060.67
m1351.87	m1220.73	m1220.73	BSAN	m1062.6
m1330.66	m5044.93	m5044.93	TMAP	m2501.25
m1332.59	m1222.66	m1222.66	LSTD	m2499.32
m1328.73	m1226.52	m1226.52	EVI	m1087.67
m1321.01	m1238.09	m1238.09	m1666.21	m2503.18

m1346.09	m4626.44	m4626.44	m1664.29	m2508.96
m1278.59	m4356.46	m4356.46	m1662.36	m2528.25
m5172.21	m1211.09	m1211.09	REF2	m1058.74
m1853.28	m2048.05	m2048.05	LSTN	m2524.39

Appendix: 4.1.4: Error scores at different learning rate for the ENR model

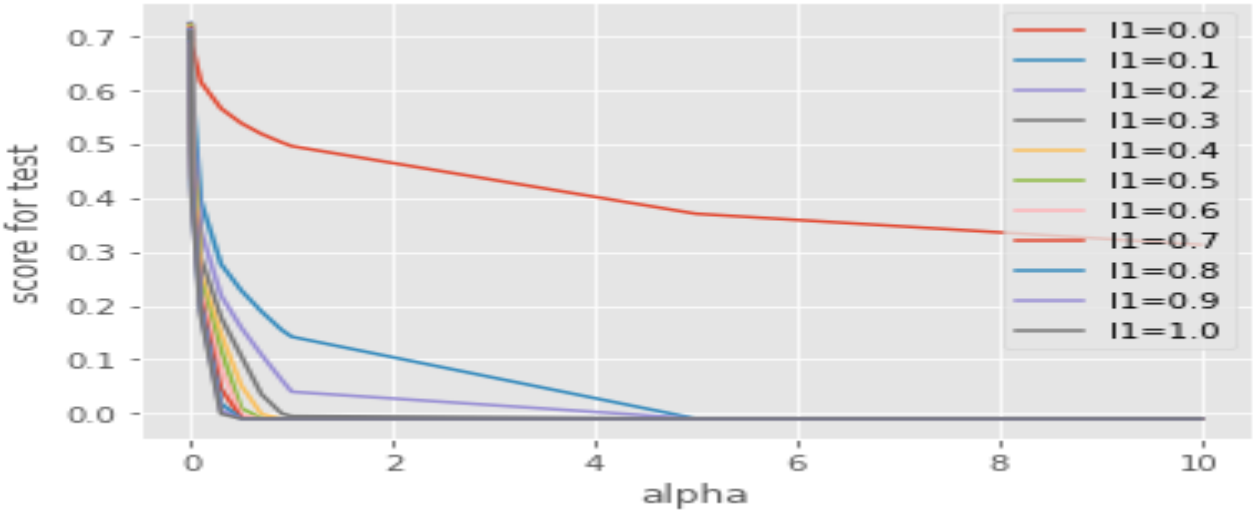


Table 5.2: Comparing the results of the test evaluation for SVR Model when feature reduction was applied with outliers removed.

	RMSE					MCRMSE
	Ca	P	pH	SOC	Sand	
Complete feature set	0.11949	0.17373	0.36185	0.21777	0.32596	**0.23976**
PCA reduced feature set (16 features)	0.18375	0.23291	0.78018	0.33078	0.40723	0.38697
Pearson Correlation's reduced feature sets	0.11760	3.35384	0.36317	0.21861	0.33115	0.87687
Spearman Correlation's reduced feature sets	0.10902	0.16449	0.38114	0.21735	0.35387	**0.24517**