**Automatic Feature grouping in high dimension penalized grouping**

**Abstract**

*In learning with data which is high dimensional, the grouping of characteristics is quite useful. The techniques described in this literature improve feature selection stability and reduces estimation variance, resulting in better model accuracy and also assisting with data interpretation. One such tool is discussed in Bondell, H.D. and Reich, B.J. [8] for selecting variables and arranging them into prediction clusters at the same time which shows good results. In addition to increasing the accuracy of predictions and interpretation. The purpose of this study is to compare ridge, lasso, elastic-net, group lasso, OSCAR (octagonal shrinkage and clustering algorithm for regression), and linear regression models to see which one works best as demonstrated by experimental results on a simulated and soil data.*

**Introduction**

The term "high dimensional" refers to a situation in which the number of dimensions is staggeringly large, making calculations extremely difficult. The number of characteristics in high-dimensional data can outnumber the number of observations. The common thread running through these issues is that when dimensionality rises, the volume of space expands rapidly, that the accessible data becomes sparse. The amount of data required to produce a statistically valid and credible result sometimes climbs exponentially as the dimensionality increases. Data is rarely distributed randomly in high-dimensions and is highly correlated, with misleading correlations typical. In high dimensions, the distances between a data point and its closest and farthest neighbors can become equidistant, possibly jeopardizing the accuracy of some distance-based analytic techniques. Datasets for high-dimensional data are often unstructured, which might make them more challenging to use. Furthermore, large datasets can contain noise and uncertainties. It can be challenging to process and apply practical data mining tools to such noisy data.

When more dimensions are added to a data set, the harder it becomes to anticipate specific values. That would seem having more is better. When it comes to introducing variables, on the other hand, the converse is true. Each additional variable reduces the prediction power exponentially. So, dimensions are reduced to make data easier to comprehend, either numerically or graphically. A technique like multidimensional scaling is used to detect commonalities in data to minimize dimensionality by combining related data into groups. Clustering can also be used to group items [8].

This study aimed to identify the approaches used to manage high-dimensional data and determine which one performs best in terms of accuracy or error rate. Which means exploring various approaches to deal with high-dimensional data and comparing them using simulation and soil dataset. In the literature, such strategies have been thoroughly explained. For example, ridge and lasso regression [1] use penalization to minimize features near to 0 or precisely zero and what impact did clustering have on the accuracy and error rate of the model as explain in [8]. Elastic-net which uses powers of both lasso and ridge to give predictor estimates [5]. Fused Lasso which has the capability to fuse and defuse a feature in order to improve model accuracy [6]. All the above-mentioned techniques are model tuning strategies that can be applied to multicollinear data analysis.

The data is a study about correlations betwixt soil parameters and forest diversity in North Carolina's Appalachian Mountains, and each of these approaches was applied separately on a soil dataset with 15 predictors and 20 samples which was also used in [8] for testing their models’ prediction. 2500 square meter sites were examined. The results are the number of various plant varieties examined inside the plot and the 15 loam parameters employed as forest diversity predictors. Each plot's soil measurements are an average of five measurements obtained at random intervals throughout the plot. The first seven variables all have something to do with the quantity of positively charged ions or cations. The quantity of cations is represented by % base saturation, exchangeable cations (CEC), and the summation of cations; cations include magnesium, calcium, potassium, etc.

The rest of the manuscript is organized in the same way section 2 does, including a literature review. Section 3 goes over the methodologies, section 4 explaining simulation results, while section 5 goes through the dataset and its characteristics correlation. The final section evaluates and discusses the findings.

**Related Literature**

When fitting a multiple regression model to a dataset, the p-value and analysis of the variance table are used to determine how significant our model is. The second most significant factor in determining which predictors are the most important and the least important as discussed in [23]. Therefore, researchers have had difficulty selecting a set of best predictors to include in their models. Variable selection is the process of selecting the best predictors from a pool of all available predictors.

As fitting a model on high dimensional data, such as when n approaches p, where n is the number of samples and p is the number of predictors, numerous studies have been conducted to deal with variable selection.

The most widely used sparse modeling algorithm is Lasso [1]. In the presence of most associated traits, however, it prefers to choose only one. As a result, estimation might be unreliable, and the resulting model can be challenging to understand as emphasized in the Journal of the Royal Statistical Society, Model selection and estimation using grouped variables [7].

Another flaw with Lasso [1] is that it is unable to locate feature groupings which is mentioned in [7] for the reason why group lasso outperforms lasso in most of the cases. Feature grouping can diminish estimator variance [2] and increase feature selection stability [3]. It also aids the discovery of co-regulated genes in microarray analysis [4]. This differs from the group lasso [7], which necessitates prior knowledge of the feature groups as shown in equation (7), (8) and (9).

A combinatorial optimization challenge which aims to improve a method by employing mathematical techniques to shrink the size of viable outcomes or speed up the search process which acts as a detailed search for feature groups. Instead, an adequate regularize might be used to stimulate its development. The elastic net [5] is a well-known technique which is utilized for regularization and variable selection. It outperforms the Lasso Regression [1] in a simulation conducted in [5]. The fused Lasso [6] explicitly encourages the successive feature coefficients to be comparable in circumstances where the features are organized in some meaningful way. However, such an ordering is not always present and must be predicted before the fused Lasso [6] can be employed.

The elastic net [5] approach is used to make regularization and variable selection on real-world data. A simulation shows how it outperforms the Lasso Regression [1] in providing a grouping effect for predictors that are highly associated with one another. When the number of predictors p is substantially more than the number of samples n, this strategy comes in handy.

OLS is very well recognized for its poor performance in both forecasting and understanding. To strengthen OLS, penalization approaches have been proposed. For instance, Ridge regression [18] reduces the leftover sum of the squares according to a limit on the variables' L2-norm. Ridge regression, as a continuum downsizing technique, achieves superior predictive accuracy via a bias-variance exchange. It cannot build a precise model because it always retains all variables in the model. In comparison, optimal group selection provides a scant model, but it is exceedingly variable due to its inherent discreteness, as Breiman points out in [19].

A comparable study using OSCAR [8] picks the features by making cluster groups achieve maximum precision for test data, illustrating predictor choice and their clustering. This method uses penalized least squares and a penalty function to reduce coefficients to zero. Furthermore, this penalty function generates comparable coefficients for some predictors with similar predictions, resulting in predictive clusters. In terms of prediction accuracy and model complexity, this study compares existing variable selection strategies with OSCAR's [8] supervised clustering of predictors, which adds automatic feature grouping to reveal additional grouping information. This was done using a simulation as well as real-world soil data.

This research focuses on efficient sparse modeling using automatic feature grouping [9], highlighting the benefits of feature grouping in high-dimensional data, enhancing feature selection, and reducing variance, resulting in a generalized model. This research in [9] on efficient sparse modeling with automatic feature grouping employs accelerated gradient methods to demonstrate how the critical projection in OSCAR [8] can be addressed using a simple iterative group merging algorithm that decreases the time complexity. This has been demonstrated on both toy and real-world data to show that OSCAR [8] is a competitive sparse modeling strategy that can do automatic feature grouping.

The RMSE has been utilized as a standard statistical tool to evaluate the abovementioned methodologies' predictive accuracy. Another relevant measure is mean absolute error (MAE). Even though they have both been used to evaluate the performance of a model for several years, there is no agreement on which metric is the best suitable for model uncertainty. Because the Root means the square error is widely used as benchmark measurement for model uncertainty in the environmental sciences discussed in [20], [21] and [22], we chose it as the performance appraisal metric for this study.

All of the techniques mentioned above focus on real-world data while studying variable selection on high-dimensional data. This study is based on soil data from research on correspondence between soil properties and forest variety in North Carolina's Mountains. There were no comparisons made with the results of other research studies, but comparisons were made on this soil dataset with the OSCAR [8], Multi-Linear Regression [23], Lasso [1], Ridge, Elastic-net, group lasso [7], and fused lasso [6]. Results were drawn, which are available in the literature.

**Methodology**

**Regression?**

Regression is the study of the relationships between variables or features in data such as one may be able to predict the value of the unknown variable for some know variables. In regression, the target variable or the variable to be predicted is known as the dependent variable, and the other variables are independent. The predictions are an estimate produced by the regression model based on some features. As a result, regression is a metric for determining the average relationship between variables. It can be used to measure mutual relationships by regression analysis. Regression analysis is helpful in business and economic research, which can also help to form policies. The nature of its relationship between variables is based on correlation, which measures the degree of relationship between X and Y. Whereas the correlation does not help make predictions, regression gives us that ability. The connection between the reliant and autonomous factors is inspected utilizing relapse investigation, a prescient displaying method.

The initial application of regression analysis is evaluating the strength of predictors. Regression can be used to determine the strength of an individualistic variable's effect on the reliant variable, such as the robustness of correspondence between sales and marketing spending or the association between age and income. Second, forecasting and effect can be used to forecast effects for the impact of changes, i.e., regression analysis can assist us to understand how much the reliant features change when one or more individualistic features change.

In simple linear regression [23], data is modeled using a straight line, used with continuous variables, and measured by loss, R-squared which is a quantitative metric that measures the amount of variation described by predictor factors.

**Regularization?**

It is a technique used to penalize the model for the overfitting of data and adding the parameter. As a result, it tunes the model for a better outcome. A model that is fitted on some training data may give zero error for training, but when it is testes against the testing data, huge errors are observed compared to actual values so to deal with this situation regularization tunes the function by adding a penalty to the error function. Now, this penalty will be used to shrink the coefficients in order to reduce the error. This overfitting can also be controlled by increasing the size of the dataset, which means converting high dimensional data to low dimensional data.

**Bias and Variance**

In statistics, the bias and variance of estimators are two important properties to examine. The disparity betwixt the true population criterion and the predicted reckoner is known as bias obtained from [24]:

(1)

It assesses the accuracy of the estimators. The dispersion, or uncertainty, in these estimates is measured by variance. For more information refer to [24].

(2)

The residuals can be used to estimate the unknown error in variance

(3)

*β = True Parameters*

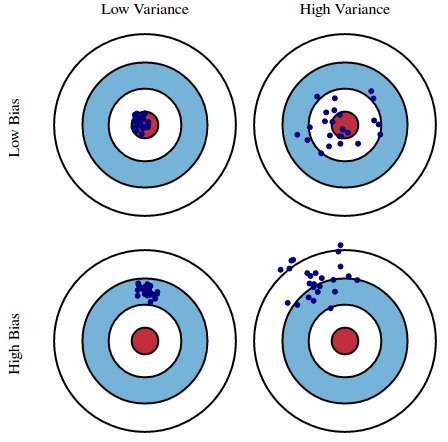
*y = Targets*

*σ2 = standard deviation*

*e = residuals*

*X = data variables*

Fig. 1 from [24, Fig. 1] depicts how variance and bias can be visually displayed. Consider the red circle to be the literal population criterion we're trying to estimate, β, and the values of our estimations arising from four distinct estimators are the shots at it.



*Fig. 1 Bias and Variance*

Variance & bias should be minimal as shown in Fig 1 in order to achieve accurate results, as substantial values out-turn in defective model predictions. The fallacy withinside the version may be divided into 3 categories: fallacy because of a huge variation, fallacy because of robust bias, and the rest obtained from [24, eq. (2)].

(4)

The unbiased property of the OLS estimator is a desirable feature. However, there might be a lot of variation. This occurs, for example, when the predictor variables have a high degree of correlation. There is a plethora of predictors. The above-mentioned variance formula (3) reflects this: as m gets close to n, the variance addresses to infinity. The typical method to this hassle is to lessen variation on the rate of bias. This technique is called regularization, and it's far clearly beneficial to the model's prediction performance.



*Fig. 2 Plot of Model Complexity against Error*

The change of evaluations develops as the model intricacy, which on account of straight relapse can be considered the number of predictors, increments, while the inclination lessens. The impartial OLS would put us on the right half of the diagram, which is not great. This is why we use regularization to decrease variety to the detriment of some inclination, consequently moving left on the diagram and closer to the ideal which is also emphasized in [24, Fig. 6].

So, the optimal goal here is to reduce model complexity, or the number of predictors, based on our discussion thus far. This could be done using forward or backward selection, but the effect of the deleted variables on the response would be unknown. Setting the coefficients of predictors to zero can be thought of as removing them from the model. Maybe than requesting them to be zero, we ought to rebuff them in case they are excessively far from nothing, consequently driving them to be little endlessly. Model intricacy can be decreased while keeping all factors in the model as discussed in Ridge Regression [13].

**Ridge Regression**

Ridge regression is a model tuning technique that can be used to analyze data with multicollinearity. L2 regularization is achieved using this method. When there is a problem with multicollinearity, least-squares are unbiased, and variances are significant, the projected values are far from the actual values. Its primary purpose is to generate a model that could generalize patterns that work best on training and testing datasets. Overfitting occurs when the model performs well on the training dataset but poorly on the testing dataset, so here, ridge regression [13] works by applying a penalization such as reducing the bias and variance to overcome overfitting.

In Ridge regression, the OLS loss function reduces variable guesses to zero by minimizing the sum error terms and penalizing the magnitude of parameter estimates.

(5)

Solving for β^ will give the ridge regression estimates for the coefficients

(6)

*I = identity matrix*

*λ = regularization penalty.*

*As λ→0, β^ridge→β^OLS;*

*As λ→∞, β^ridge→0.*

As a result, setting λ to 0 has the same impact as employing the OLS, whereas increasing λ increases penalization.

Setting λ to 0 is equivalent to employing the OLS, whereas increasing its value penalizes the size of the coefficients more.

**Advantages of Ridge Regression**

Ridge Regression eliminates overfitting, which occurs when regular squared error regression fails to distinguish the less significant features and uses them all, resulting in overfitting. It adds a slight bias to the model to fit it to its actual values, so it does not need unbiased appraisers. Ridge regression is crucial when the set of instances exceeds the training data since it outperforms the traditional summation of squares technique. The ridge estimator is exceptionally good at enhancing the least-squares approximation when multi-collinearity is present. They add just enough bias to produce the estimates approximations of genuine population values that are reasonably dependable. When there is multicollinearity, the ridge estimator is preferable for improving the least-squares estimate.

**Disadvantages of Ridge Regression**

Ridge regression, while enhancing test accuracy, employs all of the dataset's input features instead of stepwise approaches, which only use a few key variables for regression. If the feature is not essential, ridge regression decreases the coefficients theta to shallow values, but it will not eliminate them; therefore, we will keep it in our model. This flaw is overcome by using Lasso regression [1]. In the closing model, they include all of the predictors, so it cannot make feature choices. They make the coefficients less and smaller until they reach zero. They give up variation in exchange for bias.

**Lasso Regression**

Lasso regression is a technique of linear regression that is centered on contraction. In shrinkage, statistical measures are condensed towards a critical premise, such as the average. The lasso technique favors simple, sparse models, such as those with a small number of features. This sort of extrapolation is appropriate for systems with high multicollinearity or for automating parts of the model recruitment process like selecting features and variable removal. In Lasso [1], L1 regularization is utilized, and it adds a penalty equal to the values of the coefficients, which means it can completely shrink some coefficients to 0. This type of generalization can result in scant systems, with more significant penalties resulting in calculated values near to 0, which is ideal for basic models.

On the other hand, L2 regularization does not lead to the loss of variables or weak models. As a consequence, the Lasso is far more straightforward to comprehend than the Ridge. Lasso regression objective is to minimize the equation below referenced from [1, eq. (1)].

(7)

λ is a tuning parameter that determines how strong the L1 penalty is. When λ is set to 0, no parameters are removed. The result is the same as that obtained using linear regression [23]. More and more coefficients are assigned to zero and removed as λ grows. Bias grows as λ increases and variance grows as λ decreases.

**Advantages of Lasso Regression**

It can avoid overfitting, just like any other regularization method. Even if the number of characteristics exceeds the number of data, it can be used. It has the ability to select features and in terms of inference and fitting, it is quick as discussed in the Journal of Royal Statistics Tibshirani, R. Regression shrinkage and selection via the lasso [1].

**Disadvantages of Lasso Regression**

The lasso-selected model is not stable. The model selection outcome is difficult to comprehend when there are many highly linked traits, lasso may choose one or a subset of them randomly which is one reason why group lasso outperforms lasso in [7]. The outcome is determined by how well the plan is implemented. People introduced elastic net [5] to improve things. In terms of use, the prediction performance of lasso is inferior to that of Ridge regression.

**Elastic-Net Regression**

The elastic net [5] is a penalizing regression analysis framework that incorporates the L1 and L2 penalties during training. The hyperparameter "alpha" is used to determine how much weight each L1 and L2 penalty is given. The impact of the L1 penalty is valued by one minus the alpha value, whereas the impact of the L2 penalty is valued via one minus that alpha value. An alpha of 0.5, for example, would give each penalty a 50% contribution to the loss function. An α value of 0 provides the L2 penalty total weight, while a value of 1 gives the L1 penalty total weight. Elastic net [5] permits a balance of both penalties, which might result in a more excellent performance on particular tasks than a model with only one penalty. Another hyperparameter called " λ " regulates how the total of both penalties is weighted in the loss function. With a value of 1, the entirely weighted punishment is imposed by default; with 0, the punishment is not applied. λ values of 1e-3 or even below are relatively common.

The coefficients to the variables are essential information; nevertheless, ridge regression [13] does not guarantee that all irrelevant coefficients will be removed, which is one of its shortcomings over Elastic Net Regression [5]. It employs Lasso [1] and Ridge Regression regularization to eliminate any non-informative coefficients while leaving the informative ones. The ENR equation is represented below from [5]:

(8)

**Advantages of Elastic-Net Regression**

For n<<p, there is no difficulty picking more than n predictors, however, lasso [1] saturates. Elastic Net combines lasso and ridge properties. Elastic Net lowers the influence of many features but not eliminating them. Elastic Net is thought to be superior to Ridge and Lasso regression in terms of bias handling. Collinearity is better handled by Elastic Net than by combining ridge and lasso regression. As far as complexity is concerned, Elastic Net outperforms ridge and lasso regression because the number of variables is not significantly reduced. Elastic Net permits a balance of both penalties, which might result in more excellent performance on tasks than a model with only one penalty.

**Disadvantages of Elastic-Net Regression**

LASSO & Ridge’s combination is more computationally intensive.

**Group-Lasso Regression**

The grouped lasso is an expansion of the lasso in linear regression models that allows for selecting features on predefined sets of items. The estimates have an appealing virtue of being GroupWise orthogonal reparameterization invariant. For accurate regression prediction, selecting grouped variables from is dataset is the main objective of group lasso. Instead of using stepwise backward elimination to pick factors, we focus on estimation accuracy and investigate lasso [1], LARS [25], and non-negative garrote [25] extensions for feature selection.

For a vector η ∈ R^d, d >=1, and a symmetric d×d positive definite matrix K:

The solution to provided positive definite vectors K1 to KJ is known as the grouped lasso approximation. This result is the one obtained in [7, eq. (2.1)].

(9)

where λ >= 0 is a tuning parameter.

[11] presented expression (9) as a lasso extension for picking groups of variables, along with a computational algorithm.

[17] Took a similar method, selecting *Xj* and *Kj* as kernel functions and replicating kernel of the 2d matrix generated by *the jth* factor, respectively. When *p1* to *PJ =1*, it is evident that equation (9) simplifies to the lasso. The penalty function employed in expression (9) is a hybrid of lasso's l1-penalty and ridge regression's l2-penalty. In the case where all *Kjs* are identity matrices. The above expression causes lasso to be reduced when *p1 = PJ = 1*, an intermediate penalty function between l1 and l2.

[11] suggested a sequential expression optimization technique (9). A more intuitive technique was made. Group lasso implementation was based on the shooting algorithm [12] for the lasso. The algorithm has been proven reasonably stable, achieving a reasonable convergence tolerance after a few rounds. However, as the number of predictors grows, the computational overhead increases rapidly.

**Advantages of Group Lasso Regression**

Group lasso is excellent at picking individual variables, their group counterparts are better at identifying factors. When the total features exceed the size of samples, group lasso is used directly.

**Disadvantages of Group Lasso Regression**

It has a slow computation and because its solution path is not piecewise linear, it necessitates extensive computing in large-scale problems.

**Fused Lasso Regression**

The fused lasso [6] is a generalization technique used for issues involving characteristics that can be sorted in some meaningful fashion. The fused lasso [6] penalizes the L1-norm of parameters and their successive variances. As a result, it promotes the sparsity of the coefficients and the sparsity of their differences. When the size of predictors is substantially more than the sample size, fused lasso comes in handy. The technique is also used to the support vector classifier's underlying 'hinge' loss function. To show the approaches, we use data from protein mass spectrometry and gene expression.

Starting with just a simple linear model referenced from [6, eq. (1)]:

Errors with a sustained variance and 0 means. We should emphasize that p can be larger than N, and in most of the instances we investigate, it is substantially larger than N. Ridge regression [13] is one of many normalized or penalizing regression algorithms developed along with partial least squares [14], and principal components regression [13]. The lasso [1] is an alike ridge, but instead of using the squares of coefficients, it employs absolute coefficients. The lasso finds the coefficients to satisfy the equation below obtained from [6, eq. (2)]:

(10)

We get the least - square result, or some of the possible least - honest answers, when s is large enough. The solutions are sparse for smaller values of s; for example, few attributes are 0. From the standpoint of data analysis, this is desirable since it picks the most important variables while eliminating the remainder. Partial least squares, main components, and ridge regression, unlike the lasso, do not yield scant models. Although group choice yields scant models, it is not a convergent operation; optimal subset selection is combinative and not feasible for p > 30. The lasso can be employed even if p > N, but if no two predictors are completely collinear, this has an optimal solution. The fact that the number of non-zero coefficients is at most min (N, p) is an intriguing property of the solution. If p = 40 000 and N = 100, the solution will have at most 100 non-zero coefficients. The 'basis pursuit' signal estimation approach, proposed by [15], uses the same notion as lasso but in the spectrum or other dimensions. In this case, one downside of the lasso is that it overlooks feature ordering of what we would expect. Fused lasso is suggested by the equation below obtained from [6, eq. (3)]:

(11)

The first restriction promotes sparsity in the coefficients, while the second promotes sparsity in the differences. [16] coined the term "fusion" who proposed to use the penalty of form *Σj |βj −βj−1| α s2* for various values of *α=0, 1, 2* and so on.

**Advantages of Fused Lasso Regression**

In situations where the features have a natural order, the fused lasso appears to be a suitable method for regression and classification. We can fuse or defuse a group of variables in the fused lasso by adding or dropping a variable which is discussed in [7].

**Disadvantages of Fused Lasso Regression**

Computational speed is one of the drawbacks of employing the fused lasso [7].

**OLS Multi-Linear Regression**

Multiple linear regression [23] is a statistical approach for deciding the outcome of a parameter based on values of multiple variables. Multiple regression is a kind of regression analysis. The response variable is the one we are expecting, and the exogenous variables are indeed the ones we are using to do so. This technique can be used to calculate the model's fluctuation and the percentage influence of each explanatory variable on the total variance. This regression analysis can be classified into two types: linear & non-linear. The following method can be used to alter the simple linear regression equation from [23] to include many dependent variables:

(12)

*Y is the expected or dependent variable.*

*A0 is y-intercept when X1, X2 are equal to 0*

*The regression coefficients A1 and A2 represent the change in y as a function of a one-unit change in X1 and X2.*

*For each independent variable, ANXN is the slope coefficient.*

Considering knowledge from another parameter, analysts can use a simple linear method to estimate the value of a variable. The goal of linear regression is to create a smooth contrast between two parameters. Multiple regression is a type of regression in which the response variable has a linear relationship with two or more factors. If both factors do not follow a perfect line, it is called non-linearity [23]. To visually monitor a specific response, both linear and non-linear regression employs different variables. Non-linear regression is extremely hard to implement since it relies on hypotheses learned via experimentation.

**Observational independence:**

According to the model, the information is irrelevant to each other. The model states that perhaps the residual values are unrelated. Multivariate normality occurs when residuals are dispersed in some predictable pattern.

**Advantages of OLS Multi-Linear Regression**

The ability to identify the relative importance of one or even more predictor variables upon that criterion. It is sensitive enough to detect anomalies and the essential benefit of Multi-Linear regression [23] is that It facilitates the comprehension of links among data samples. This would make the correspondence among dependent and independent factors clearer to understand.

**Disadvantages of OLS Multi-Linear Regression**

Multivariate approaches are a little more complicated and necessitate much math. The output of the multivariate regression model can be challenging to comprehend at times since some loss and error outputs are not identical. Smaller datasets are not well-suited to this strategy. As a result, the same cannot be said of them. For larger datasets, the outcomes are better.

**OSCAR (octagonal shrinkage and clustering algorithm for regression)**

OSCAR [8] is a method for selecting variables and organizing them into prediction clusters simultaneously. In addition to improving prediction accuracy and understanding, these generated clusters could be further investigated to see what leads to a group's similar actions. The process relies on a punitive least-square, which uses a mathematically simple punishment mechanism to reduce some coefficients to zero. Furthermore, because this punishment leads to exact equality of some parameters, it promotes linked variables with effects similar to the outcome to cluster into prediction groups consisting of a single factor.

The OSCAR [8] is built using a constrained least-squares problem, similar to earlier techniques. For the coefficients, L1 norm and L ∞ norm's weights were combined and employed as the constraint. Its optimization problem is defined below obtained from [8, eq. (1)]:

(13)

The calibration constants c and t,

c controls the relative weighting of norms, while the magnitude is controlled by t.

L ∞ norm promotes coefficients equality.

For the features which are not 0, the OSCAR’s optimization framework favors a cheese-pairing solution. On analyzing the solution of least squares, we get to know how this penalty will support grouping and sparsity simultaneously. The sum of squares loss function is represented below from [8, eq. (2)]:

(14)

The contours are in the form of ellipses which are centered at the solution which is Beta0.

Because the predictors are normalized, the central axis of the curves is at ±45◦ degrees to the floor when p = 2. Because the contours are XTX, a high correlation would result at -45◦, but a negative correlation would be the opposite.

First-ever time the boundaries of the sum-of-squares error function reach the limitation area on the *(β1, β2)* plane, the answer is straightforward. The LASSO and the Elastic Net [5] that use a combination of L1 and L2 penalties have different constraint regions mentioned in [8]. Because the outlines are more capable of hitting the vertices, the non-differentiability of LASSO [1] or Elastic Net [5] just at the axis encourages randomness, with LASSO [1] accomplishing a higher amount owing to the straight border. Meanwhile, if multiple variables are firmly linked, Elastic Net is more likely to incorporate both in the equation than just one.

Consider OSCAR's representation in terms of the penalized least-squares criteria with λ as the penalty parameter. Supposing set of related variables *x1 to XP* are in the order so their coefficient approximations satisfy *0 < |βˆ1|≤···≤|βˆQ|* and *βˆQ+1 = ··· = βˆp = 0*. Let *0 < ˆθ1 < ··· < ˆθG* denotes values which are not equal to 0 of set *|βˆj |,* so that *G ≤ Q.*

*For every g = 1 to G,*

*let Gg = {j : |βˆj | = ˆθg}*

*Constructing group n x G*

*Related variable matrix X∗ ≡ [x∗ 1 to x∗ G] with*

*x∗ g = j∈Gg sign(βˆj )xj obtained from [8, eq. (4)].*

When the outlines of the sum-of-squares function meet with the octagonal restriction area for the very first time, the OSCAR solution occurs.

**Exact Grouping Property**

The OSCAR representation like a problem of optimization in penalized form is as follows referenced from [8, eq. (3)]:

(15)

The restriction limit t and the penalization variable now have an explicit relationship.

This transformation comprises combining predictors with comparable sizes of factors by a combining information of their quantities, just like when launching a unique forecast from the group mean. As a result, the summed weights are as follows obtained from [8]:

This can also be explained as active group of related variables with *0 < θ1 < ··· < θG from the equation below referenced from* [8, eq. (5)]:

(16)

Equation can be used to calculate the λ value for a solution obtained from a given t value.

Which can be shown as equation (17) from [8, eq. (7)]:

(17)

**Computation and Cross validation:**

A computational technique is used to determine the OSCAR approximation for one particular group of calibration parameters (t, c). *βj = β+ j − β− j* with both *β+ j and β− j* being positive then *|βj | = β+ j + β− j.* Furthermore, for bilateral maxima, *p(p−1)/2* factors *ηjk* for *1 ≤ j* is introduced, making the optimization issue more difficult as mentioned in [8, eq. (8)]:

Subject to

(18)

The minimization is done with the extended parameter vector in mind. This forms a quadratic problem with total parameters of *(p2 + 3p)/2* and total linear constraints of *p2 + p + 1*. Although the constraint matrix is big, it is also extremely sparse. In [26] a quadratic programming approach designed primarily for substantial sparse matrix issues was used to execute the optimization. This algorithm can immediately solve problems with a few predictors around 100.

**Choosing the tuning parameters:**

Lowering the out-of-sample forecast error approximation can be used to choose the tuning parameters *(c, t).* This can be easily estimated if a validation set is supplied. In the absence of a validation set, 5-fold cross-validation can be used to determine the standard errors. The amount of positive actual variables is a reliable measure of degrees of freedom for the LASSO [10], [5]. Calculate the degrees of freedom infused lasso [6] by counting the positive, unique blocks of coefficients.

**Advantages of OSCAR:**

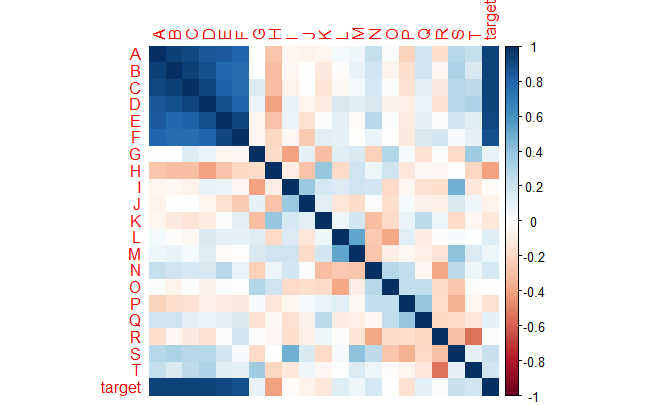
It is the latest approach for choosing variables in regression analysis while also doing guided clustering. In contrast to least-squares extrapolation criteria, the OSCAR penalty can be used for other optimizations. Like other penalized regression techniques, the OSCAR solution can be viewed as the post method for early distributions.

**Disadvantages of OSCAR:**

Many conventional solvers may have problems immediately tackling the quadratic programming issue due to its size.

**Simulation**

To evaluate the performance of the aforementioned strategies, a simulation study was conducted. With p = 20 covariates and a sample size of n = 25, this is an example problem. Each of the first six variables has a true coefficient of one and is normally distributed with a strong AR (1)-type connection. The remaining 14 variables are unaffected independent normals.



*Fig. 3 Corrplot showing strong correlation between first 6 features*

Fig. 3 shows demonstrates for the sample problem such as the first 6 features A to F have a strong AR (1)-type connection and the remaining 14 variables are unaffected independent normals.

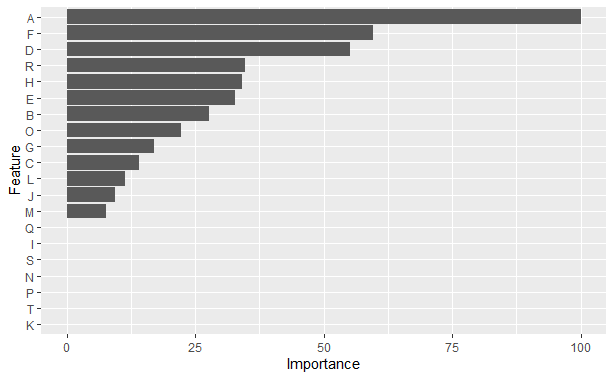
The dataset was then divided into 70/30 ratio for training and testing in order to calculate error rates like RMSE and r-squared. The model was given a collection of features X and a response vector Y as inputs. X is the n by p design matrix. The coefficients of the standardized predictors will be returned. The train Control parameters are stored in a control specifications variable, which is used for cross-validation up to 5 folds to find the best value for λ. The models were then trained on this simulated dataset and the following coefficients were returned.

**Table 1**

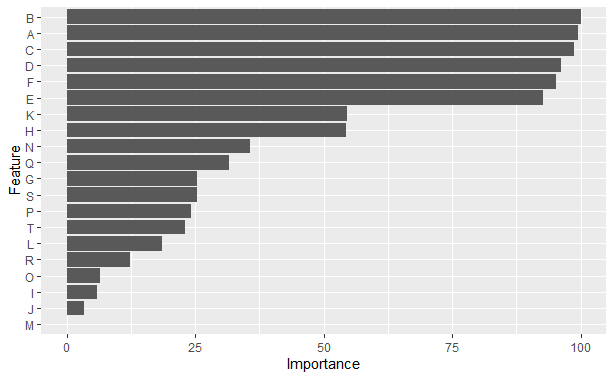
*Estimated coefficients for simulated dataset*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Variables | Lasso | Ridge | Elastic-Net | Group Lasso | OSCAR |
| A | 0.292 | 0.053 | 0.151 | 0.202 | 0.204 |
| B | 0.081 | 0.053 | 0.134 | 0.162 | 0.139 |
| C | 0.041 | 0.052 | 0.100 | 0.086 | 0.114 |
| D | 0.161 | 0.051 | 0.101 | 0.112 | 0.338 |
| E | 0.096 | 0.049 | 0.112 | 0.139 | 0 |
| F | 0.174 | 0.051 | 0.156 | 0.240 | 0.154 |
| G | 0.050 | 0.014 | 0.028 | 0.042 | -0.017 |
| H | -0.099 | -0.029 | -0.066 | -0.069 | -0.138 |
| I | . | 0.003 | . | . | -0.091 |
| J | 0.027 | 0.002 | . | . | 0.043 |
| K | . | -0.029 | . | . | 0.055 |
| L | 0.033 | 0.010 | . | . | 0.141 |
| M | 0.023 | 0.000 | . | . | 0 |
| N | . | 0.019 | . | . | 0.101 |
| O | 0.065 | -0.004 | . | . | 0.154 |
| P | . | -0.013 | . | . | -0.084 |
| Q | . | -0.017 | . | . | 0 |
| R | 0.102 | 0.007 | . | . | 0.152 |
| S | . | 0.014 | . | . | 0.064 |
| T | . | 0.012 | . | . | 0 |

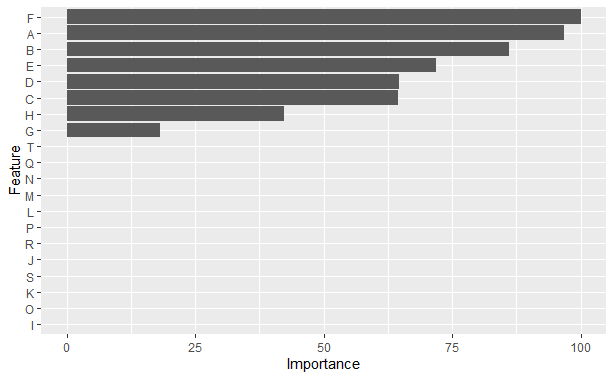
The estimated coefficients for the simulated dataset are shown in Table 1. Seeing for the Lasso, we know that the first six features with strong correlation are penalized to a lesser extent than other features, and the remaining features were reduced to zero as done by the Lasso. For the ridge, no feature was eliminated, but the strongly correlated features were given greater weight. Because the elastic net is a hybrid, it lowered more than half of the features to zero, except for the first six, which were merely punished to a degree. By generating groups of characteristics, Group Lasso achieves comparable outcomes to the elastic net. For the OSCAR, we can see that it has used cluster formation to perform penalization, and the estimated coefficients are shown in table 1.

****

*Fig. 4 Variable importance for Lasso*

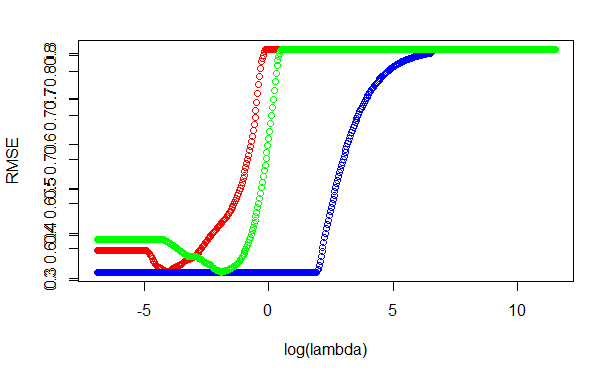
****

*Fig. 5 Variable importance for Ridge*

****

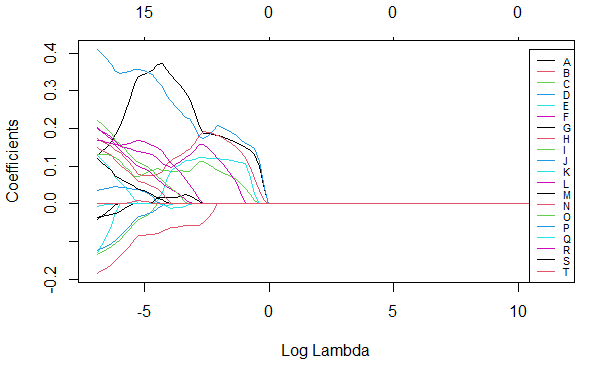
*Fig. 6 Variable importance for Elastic-Net*

Fig. 4, 5 & 6 show the importance given to the variables while training the model respectively.

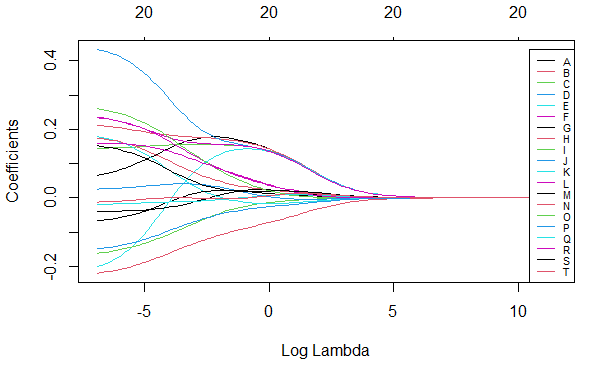
****

*Fig. 7 Plot for finding optimal λ for lasso, ridge & elastic-net*

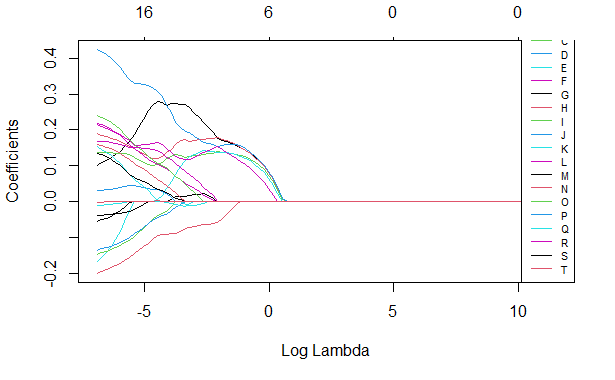
In the case of lasso, ridge, and elastic-net regression, Fig. 7 shows the graph for determining the best value for λ. This is a plot of log (λ) against RMSE, with the best λ value chosen for the model with the lowest RMSE value. The red line represents the lasso, the blue line represents the ridge results, and the green line indicates elastic-net.

****

*Fig. 8 Plot showing how increasing λ shrinks the coefficients for lasso regression*

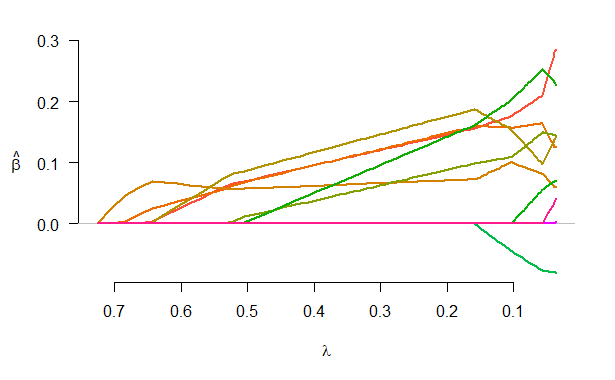


*Fig. 9 Plot showing how increasing λ shrinks the coefficients for ridge regression*

****

*Fig. 10 Plot showing how increasing λ shrinks the coefficients for Elastic-net regression*

Fig. 8, 9 & 10 shows how increasing the value of λ shrinks the coefficients for ridge, lasso & elastic net respectively. Each line represents the coefficients for one variable for various λ s. The higher the λ, the smaller the coefficients become as they approach 0.



*Fig. 11 Plot of fit for the group lasso*

**Table 2**

*Calculated values of RMSE & R-squared for all models on simulated dataset*

|  |  |  |
| --- | --- | --- |
| **Models** | **RMSE** | **R-squared** |
| Lasso | 0.2340 | 0.9806 |
| Ridge | 0.9653 | 0.9980 |
| Elastic-Net | 0.1923 | 0.9915 |
| Group Lasso | 0.1897 | 0.9884 |

Table 2 compares all models in terms of RMSE and r-squared, making it easier to assess their performance. Training and testing groups were created using the simulated dataset. The training set was used to understand the patterns in the dataset, and the testing set was used to determine how accurate the model was. The models were evaluated for the test sample after training, and the predictions were compared to the actual test data to obtain the RMSE and R-squared errors. Table 2 demonstrates that in this example, the ridge did the poorest in terms of RMSE, whereas the lasso performed significantly better, with an RMSE score of 0.234. Elastic Net came in second, outperforming the lasso, a combination of both, with an RMSE of 0.1923. Finally, the group lasso demonstrates that grouping is critical to model performance, with an RMSE of 0.1897, the best in this scenario.

**DATASET**

This example is based on a survey of correlations among soil characteristics and valuable forest variety in the Appalachian Region of North Carolina. Twenty 500 square meter sites were examined. The results are the number of various plant varieties examined inside the plot and the 15 loam parameters employed as forest diversity predictors. Each plot's soil measurements are an average of five measurements obtained at random intervals throughout the plot. Figure 3 depicts several highly associated predictors. The first seven variables all have something to do with the quantity of positively charged ions or cations. The quantity of cations is represented by % base saturation, exchangeable cations (CEC), and the summation of cations; cations include magnesium, calcium, potassium, etc. In rare circumstances, the close bilateral relationships between such factors can reach 0.95. The potassium and sodium correlations are not quite as strong as the others.

Salt plus phosphorus, as well as soil pH and interchangeable acidity, 2 acidification metrics, have a strong relationship.

Dataset consists of all numeric features

**BaseSat**: a summary of the cation abundance;

**SumCation**: a summary of the cation abundance;

**CECbuffer**: a summary of the cation abundance;

**Ca**: Calcium which is the example of cation

**Mg**: Magnesium which is the example of cation

**K**: Potassium which is the example of cation

**Na**: Sodium which is the example of cation

**P**: Phosphorus which is the example of cation

**Cu**: Copper which is the example of cation

**Zn**: Zinc which is the example of cation

**Mn**: Manganese which is the example of cation

**HumicMatter**: Compounds found in humus that constitute significant components

**Density**: Measure of density for the soil data

**pH**: hydrogen ion concentration in a solution on average

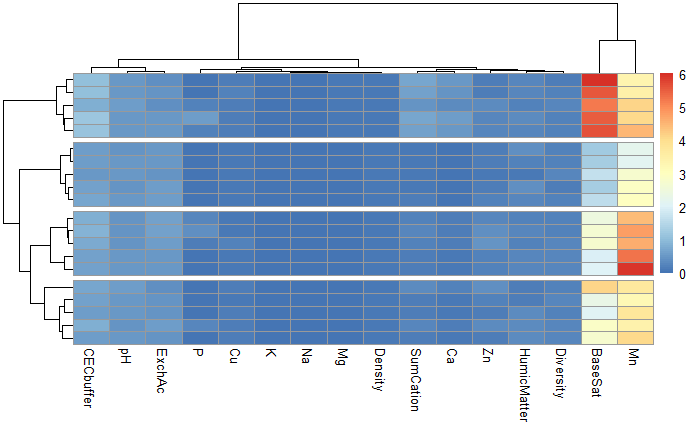
**ExchAc :** numericattribute representing exchangeable acidity

**Diversity:** Target attribute / dependent variable for forest diversity prediction

**Heatmap**

A heatmap is a graphical depiction of data that use a color-coding method to represent various values. Heatmaps are utilized in a variety of analytics applications, but they're most typically employed to display user behavior on webpage designs. Heatmaps can provide a more detailed picture of how features are actually behaving.

**Complex Detailed Heatmap of our soil dataset:**

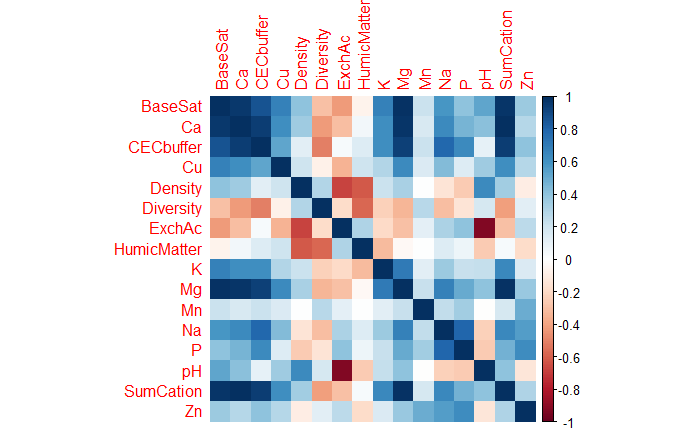


*Fig. 12 Complex heatmap for the soil dataset*

**Corrplot:**

The R package corrplot is a visual exploring tool for correlation matrices that includes automatic variable reordering to aid in the detection of hidden patterns. Corrplot is simple to use and offers a wide range of plotting options in terms of visualization approach, graphic layout, color, legends, and text labels, among other things.

**Corrplot for our soil dataset:**



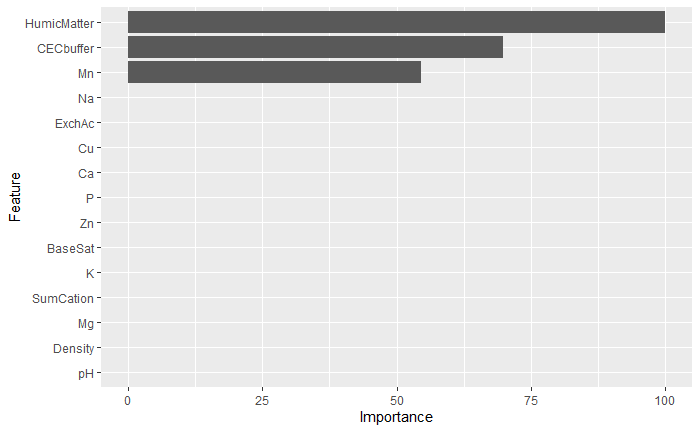
*Fig. 13 Corrplot of soil dataset representing correlation between features*

The convergent validity of the 15 soil data variables are visually shown. A block in the colored image represents the magnitude of each pairwise association.

**ggplot**:

ggplot2 is a charting program that makes it simple to create complex charts using the information in the data framework. It includes more effective feedback to determine which factors must be charted, how they are being presented, and other graphic cues. For Lasso, ridge, elastic-net, and OLS multi-linear regression, we utilized ggplot to show the variable importance. So, to look at which aspects were deemed the most significant for that model and which features were deleted or decreased to 0 for that model.

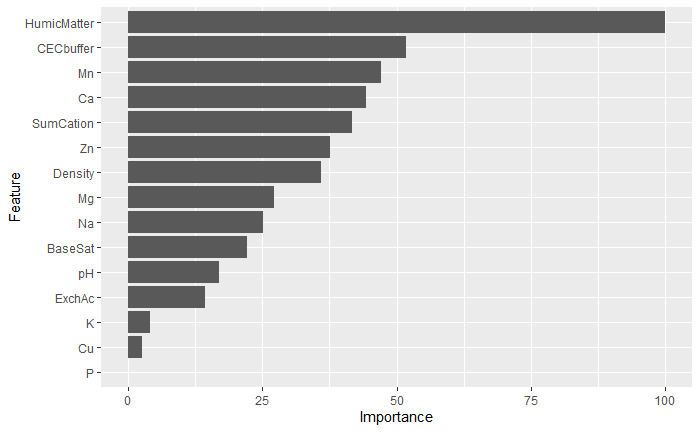
**LASSO REGRESSION:**



*Fig. 14 ggplot of variable importance for Lasso Regression*

We can see that lasso has reduced 12 features to 0 by looking at the variable relevance for lasso regression. HumicMatter, CECbuffer, and Mn are the main essential aspects considered here. CECbuffer was reduced to 69.85 percent and Mn to 54.57 percent, with HumicMatter having the highest importance of 100 percent. While the rest of the dataset's features were reduced to a value of 0.

**RIDGE REGRESSION:**

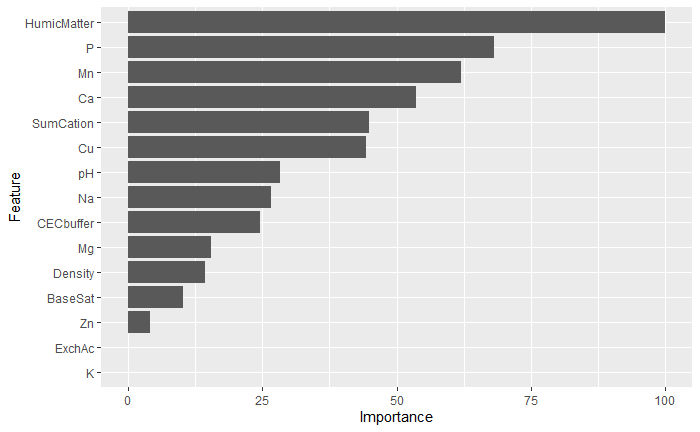


*Fig. 15 ggplot of variable importance for Ridge Regression*

When we look at the variable relevance for ridge regression, we can see that rather than entirely lowering all of the dataset's features to 0, as shown in lasso, all of them have some importance. In this scenario, HumicMatter has 100 percent importance, which was also the case in the lasso. CECbuffer was reduced to 51.79%, Mn 47.09%, Ca 44.3%, SumCation 41.6%, Zn 37.5%, Density 35.9%, Mg 27.1%, Na 25%, BaseSat 22.2%, pH 17% ExchAc 14%, K 4.1%, Cu 2.66% and P 0%.

According to ridge regression, the only feature that was punished the most was phosphorus; the rest were decreased to a specific point.

**ELASTIC-NET:**

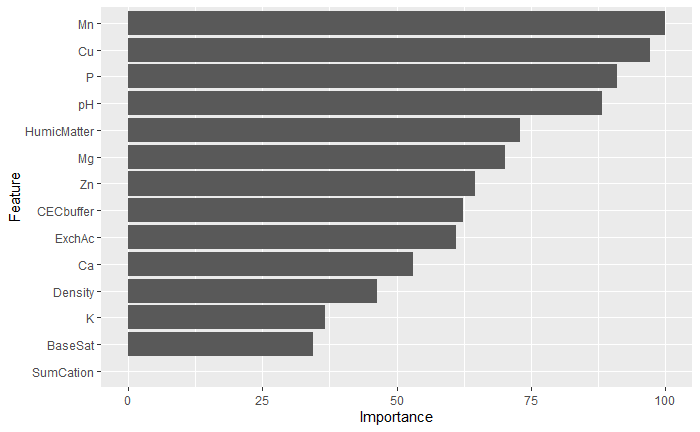


*Fig. 16 ggplot of variable importance for Elastic-Net Regression*

The elastic net is a hybrid of the ridge and the lasso. This means combining the ridge and lasso powers to create a powerful model. The elastic-net variable importance tells us that some characteristics will be fully lowered to 0 (which is used in lasso), and some features will be penalized (used in the ridge). The HumicMatter has the highest priority of 100 percent, as it did in ridge and lasso. The second most essential feature was phosphorus, which was reduced to only 68.1%, Mn to 61%, Ca 53.5%, SumCation 44.8%, Cu 44.2%, pH 28.2%, Na 26.6%, CECbuffer 24.6%, which was considered the second most important for ridge and lasso but not in the case for elastic-net. Mg was reduced to 15.5%, density 14.3% BaseSat 10.2%, Zn 4% and ExchAc and K to 0%.

Only two features, ExchAc and K, were lowered in the case of elastic-net. The remainder of the attributes, on the other hand, were penalized and lowered to a certain extent.

**OLS MULTI LINEAR REGRESSION**



*Fig. 17 ggplot of variable importance for OLS Multi-Linear Regression*

For the variable importance of OLS MLR, we see that the penalization is very low for each feature. In this case, Manganese was considered the most important for the OLS MLR model with an importance of 100%. Cu was considered second with 97% importance, p 90.9%, pH 88.1%, HumicMatter 72.8%, which was considered the most important for the ridge, lasso, and elastic-net models. Mg 70.1%, Zn 64.5%, CECbuffer 62.3%, ExchAc 61%, Ca 52.9%, density 46.3%, K 36.6%, BaseSat 34.5% and SumCation 0%. Here SumCation was the only feature that the OLS MLR model gave 0% importance.

**Table 3**

*Estimated coefficients for soil dataset*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Variables | Lasso | Ridge | Elastic-Net | Group Lasso | Fused Lasso | OLS MLR | OSCAR |
| BaseSat | . | -0.001 | -0.002 | 0.001 | 6.975 | -0.090 | . |
| SumCation | . | -0.001 | -0.008 | -0.009 | 0.919 | 0.177 | -0.178 |
| CECbuffer | -0.012 | -0.002 | -0.005 | -0.001 | 0.856 | 3.495 | -0.178 |
| Ca | . | -0.001 | -0.010 | -0.238 | 0.781 | -3.010 | -0.178 |
| Mg | . | -0.001 | -0.003 | 1.568 | 0.115 | -0.849 | . |
| K | . | 0.000 | . | -1.502 | 0.027 | -0.071 | -0.178 |
| Na | . | -0.001 | -0.005 | -2.655 | 0.011 | . | . |
| P | . | 0.000 | 0.013 | 0.033 | 0.628 | 0.030 | 0.091 |
| Cu | . | 0.000 | 0.008 | 0.051 | 0.180 | 0.073 | 0.237 |
| Zn | . | 0.001 | -0.001 | -0.043 | 0.389 | -0.025 | . |
| Mn | 0.009 | 0.001 | 0.012 | 0.011 | 3.468 | 0.033 | 0.267 |
| HumicMatter | -0.017 | -0.003 | -0.019 | -0.231 | 0.307 | -0.090 | -0.541 |
| Density | . | 0.001 | 0.003 | 0.000 | 0.023 | -0.058 | . |
| pH | . | 0.001 | 0.005 | 0.000 | 0.117 | 0.087 | 0.145 |
| ExchAc | . | -0.001 | . | 0.000 | 0.268 | -1.060 | . |

The estimated coefficients of the 15 predictors for the seven models are shown in Table 3. After completing 5-fold cross-validation to assess the ideal value, all coefficients were calculated, and models were trained after assigning a specific penalty to the predictors based on the models' concerns. Lasso's penalty is set to alpha = 1, implying that it can decrease the features to zero, as shown in the table. Only CECbuffer, Mn, and HumicMatter are lowered, while the rest are punished and reduced to zero. The second method is ridge regression, which works well with our soil dataset. The value for alpha for the ridge is set to 0, indicating that it can punish and reduce features but not to zero, similar to how PCA (principal component analysis) is used to make feature selection by reducing the dataset's dimensions. All of the predictors are penalized and decreased to some extent, but not to zero, implying that some information from each predictor is used to enrich the model. The value for alpha in elastic-net, which is a blend of ridge + lasso, is set at 0.5 to combine the powers of both models. As a result, elastic-net can penalize predictions to some extent while also reducing some to zero. As shown in the table, potassium (K) and ExchAc have been decreased to zero, while the remaining attributes have been penalized and reduced to some degree. Because it is a blend of ridge & lasso, it produces results somewhere in the middle.

Furthermore, taking into account group lasso, which groups the predictors, was not done in any of the prior models presented. When cross-validation is used for gene data, the best group is picked for the highest correlated predictors. When looking at group lasso, it is evident that the highly linked variables prefer to stay together, whereas others join the group for a short time before being pulled away. When we look at the predictor coefficients for group lasso, we can see that each coefficient is penalized for the best group picked, with some being decreased to near zero, implying that they are less essential to supplement the model. The fused lasso [6] takes two alpha values and can fuse and defuse a cluster of features by adding or eliminating one. Some predictors are also defused by penalizing them, but none are reduced to exactly 0 via fused lasso. When it comes to multi-linear regression, OLS has low penalizing power. That is, it cannot make feature selection based on punishing or lowering predictors. Looking at the table of coefficients, we can see that OLS MLR has reduced the predictors by a small amount, but not exactly 0, and has considered all of the features to be necessary, resulting in a change in results because no good feature selection or feature reduction is being performed on this high-dimensional soil data. As a result, it prioritizes all features over the ridge, lasso, and elastic net. Finally, the OSCAR [8] approach can pick variables by grouping them into clusters. All of the predictors chosen by the LASSO and additional cation covariates are included in the fivefold cross-validation OSCAR solution (Table 3).

The OSCAR [8] approach incorporates all four cationic variables into a single model with six non-negative variables. The cation variables are all related to the same underlying component and are highly correlated. As a result, instead of considering them as separate variables and selecting a sample at random, utilizing their sum as a derived predictor may yield a more accurate estimate of the root cause, and hence a precious and accurate model. We can see that CECbuffer is the first cation-related covariate to enter the OSCAR [8] and lasso because lasso is a specific instance of OSCAR. Manganese (Mn) and HumicMatter are two more covariates that the OSCAR and Lasso might choose from.

**Results and Discussion**

**Performance evaluation criteria for finding optimal λ:**

The key challenge in this study was determining the best tuning parameter λ for the model in order to reduce RMSE and R-squared errors. Because this is a regression problem and the aim is Diversity, which is a continuous variable, the evaluation metrics RMSE and R-squared are utilized. Coming towards the selection of optimal λ the trainControl function is using from the caret package to implement cross-validation up to 5-folds and the method is initialized in the control specifications variable.

The following code demonstrates the syntax used to accomplish cross-validation:

*Control\_specifications <- trainControl (method = "cv", number = 5,*

*save Predictions = "all")*

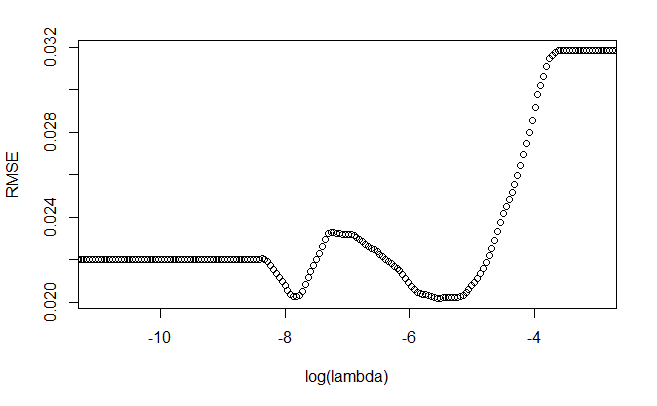
Now a vector was created to store the potential values for λ. The vector’s values ranged from -5 to 5 with length = 500.

Following code shows how the vector was initialized:

*lambda \_vector <- 10^seq (5, -5, length=500)*

The models were now trained with the train function, and the regression models lasso, ridge, and elastic-net were implemented with the “glmnet” package, which was also supplied a vector of λ values. Following successful training, we obtain the optimal tuning parameter λ, with the lowest RMSE. The log (λ) graphs were plotted against the model's RMSE.

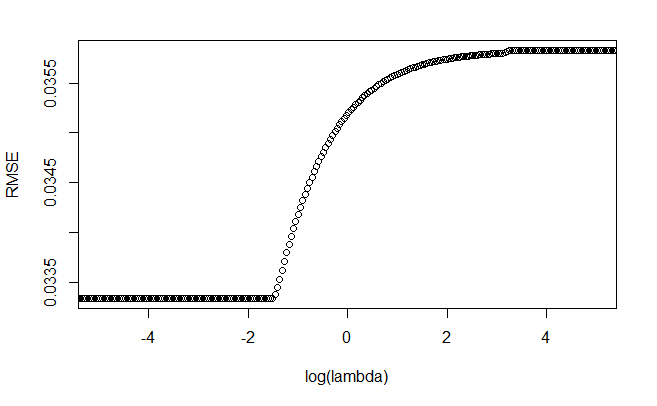
**Lasso Regression:**

****

*Fig. 18 Plot of RMSE against log (λ) for Lasso Regression*

Looking at the graph, we can see that in the case of lasso regression, the ideal value of λ is 0.004029122.

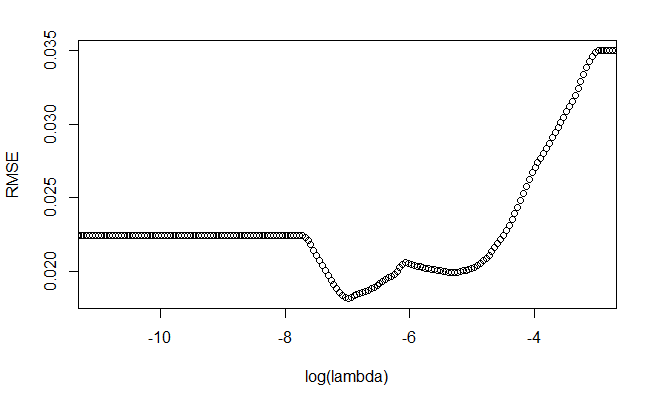
**Ridge Regression:**

****

*Fig. 19 Plot of RMSE against log(λ) for Ridge Regression*

In the instance of ridge regression, the best λ value is 0.2232016.

**Elastic-Net Regression:**

****

*Fig. 20 Plot of RMSE against log(λ) for Elastic-Net Regression*

Looking at the graph for elastic-net, we can see that the best tuning parameter λ is 0.0009202967 in this situation.

So, different regression models have different λ values, and they have different alpha values, which is the penalization parameter. This optimal tuning parameter is critical in this case because it determines the value of a low RMSE, and the less the value of RMSE, the better fit is seen. In the next section, we'll look at the RMSE and R-squared assessment measures.

**Regression Results (RMSE & R-squared):**

**RMSE**

The RMSE is a metric about how spaced out the data points are. Residuals are a measurement of how distant the data points are from the regression line. To put it another way, it shows how closely the data is grouped around the line of greatest fit. RMSE is commonly used to assess research findings in climatology, predictions, and multiple regression.

RMSE error metric is widely used, and it is regarded as a good general-purpose error meter for numerical forecasts.

RMSE can be calculated using the given formula:

(19)

**R-squared:**

In a regression model for a predictor variable, R-squared is a quantitative metric that measures the amount of variation described by predictor factors. R-squared demonstrates how much 1 variable's change explains the variability of the other, while correlation explains the significance of the association between two independent variables. The coefficient of determination is another name for it.

R-squared error can be calculated by the given formula:

(20)

**Table 4**

*Calculated values of RMSE & R-squared for all models on soil dataset*

|  |  |  |
| --- | --- | --- |
| **Models** | **RMSE** | **R-squared** |
| Lasso | 0.0312 | 0.0248 |
| Ridge | 0.0281 | 0.1174 |
| Elastic-Net | 0.0311 | 0.0352 |
| Group Lasso | 0.0278 | 0.1365 |
| OLS Multi-Linear Reg | 0.0684 | 0.0023 |

Table 4: Compares all of the models in terms of RMSE and r-squared, which aids in evaluating each model's performance. The soil dataset was split into two groups: training and testing. Our model used the training set to learn the patterns in the dataset and the testing set to see how accurate it was. After training, the models were tested for the test sample, and the predictions were compared to the actual test data, from which the RMSE and R-squared errors were calculated. Table 4 reveals that the OLS MLR performed the worst in this situation, with the largest RMSE and lowest R2, indicating that it could not fit the data well and that linear regression could not punish the features as effectively as other models. Lasso regression outperformed OLS MLR scoring, with RMSE of 0.0312 and R2 of 0.0248, both acceptable values. In this situation, Elastic-Net came in third with an RMSE of 0.0311 and an R2 of 0.0352, which are similar enough values and somewhat better outcomes than the lasso.

Furthermore, we have the Ridge regression, which penalizes features to some amount but not to 0 as the lasso does, ensuring that each feature provides meaningful information to the model. As can be seen, ridge outperforms the results of the previously stated models with an RMSE of 0.0281 and an R2 of 0.1174; this is the lowest RMSE figure and the highest R2 value to date. Next, we have the Group Lasso, a special case of OSCAR that performs the grouping of features such as gene groups. This model performs the best for our soil dataset, with the lowest RMSE of 0.0278 and the highest R2 of 0.1365, indicating that grouping improves the model's accuracy and yields the lowest values of errors.

**Conclusion**

We explored various approaches to deal with high-dimensional data in this study, and we compared them using simulation and soil datasets. We discovered that grouping had a significant impact on model correctness and error reduction. For the core projection step, we first looked at the properties of all the algorithms and how they function to come up with the best possible answer and which technique outperforms the others and why. OSCAR is a competitive regularize for classification and regression problems, with the extra capability of automatic feature aggregation, as computed and illustrated in the experiments.

**Acknowledgements**

**References**

[1] R. Tibshirani, “Regression shrinkage and selection via the lasso”, *Journal of the Royal Statistical Society*, Series B, 58(1): pp. 267–288, 1996

[2] X. Shen, and H.C. Huang, “Grouping pursuit through a regularization solution surface”. *Journal of the American Statistical Association*, 105(490): pp. 727–739, 2010

[3] R. Jornsten and B. Yu, “Simultaneous gene clustering and subset selection for sample classification via MDL”. *Bioinformatics*, 19(9): pp. 1100–1109, 2003

[4] M. Dettling and B. Buhlmann, “Finding predictive gene groups from microarray data”. *Journal of Multivariate Analysis*, 90(1): pp. 106–131, 2004.

[5] H. Zou and T. Hastie, “Regularization and variable selection via the elastic net”. *Journal of the Royal Statistical Society*. Series B, 67(2): pp. 301, 2005

[6] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu and K. Knight, “Sparsity and smoothness via the fused lasso”. *Journal of the Royal Statistical Society*, Series B, 67(1): pp. 91–108, 2005.

[7] M. Yuan and Y. Lin, “Model selection and estimation in regression with grouped variables*”. Journal of the Royal Statistical Society*: Series B, 68(1): pp. 49– 67, 2006.

[8] H.D. Bondell and B.J. Reich, “Simultaneous regression shrinkage, variable selection and clustering of predictors with OSCAR”. *Biometrics*, 64(1): pp. 115, 2008.

[9] L. W. Zhong and J. T. Kwok, “Efficient Sparse Modeling With Automatic Feature Grouping”, *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 9, pp. 1436-1447, Sept. 2012, doi: 10.1109/TNNLS.2012.2200262.

[10] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, “Least angle regression”, *Annals of Statistics 32*, 407–499, 2004.

[11] S. Bakin, “Adaptive regression and model selection in data mining problems”, *PhD Thesis. Australian National University*, Canberra, 1999.

[12] W. J. Fu, “Penalized regressions: the bridge versus the lasso”, *J. Comput. Graph. Statist*., 7, pp. 397–416, 1999.

[13] A. E. Hoerl, and R. Kennard, “Ridge regression: biased estimation for nonorthogonal problems”, *Technometrics*, 12, pp. 55–67, 1970.

[14] H. Wold, “Soft modelling by latent variables: the nonlinear iterative partial least squares (NIPALS) approach”, *In Perspectives in Probability and Statistics*, in Honor of M. S. Bartlett, pp. 117–144, 1975.

[15] S. S. Chen, D. L. Donohoand and M. A. Saunders, “Atomic decomposition by basis pursuit”. *SIAM Rev*., 43, pp. 129–159, 2001.

[16] S. Land and J. Friedman, “Variable fusion: a new method of adaptive signal regression”, *Technical Report. Department of Statistics, Stanford University*, Stanford, 1996.

[17] Y. Lin and H.H. Zhang, “Component selection and smoothing in smoothing spline analysis of variance models”, *Technical Report 1072. Department of Statistics, University of Wisconsin*, Madison, 2003: (Available from http://www.stat.wisc.edu∼yilin/.)

[18] A. Hoerl and R. Kennard, “Ridge regression”, *In Encyclopedia of Statistical Sciences*, vol. 8, pp. 129–136, 1988.

[19] L. Breiman, “Heuristics of instability and stabilization in model selection”, *Ann. Statist*., 24, pp. 2350–2383, 1996.

[20] S.A. McKeen, J. Wilczak, G. Grell, I. Djalalova, S. Peckham, E. Hsie, W. Gong, V. Bouchet, S. Menard, R. Moffet, J. McHenry, J. McQueen, Y. Tang, G.R. Carmichael, M. Pagowski, A. Chan, T. Dye, G. Frost, P. Lee, and R. Mathur, “Assessment of an ensemble of seven realtime ozone forecasts over eastern North America during the summer of 2004”, *J. Geophys*. Res., 110, D21307, 2005. doi:10.1029/2005JD005858,

[21] N. H. Savage, P. Agnew, L.S. Davis, C. Ordóñez, R. Thorpe, C.E. Johnson, F.M. O’Connor and M. Dalvi, “Air quality modelling using the Met Office Unified Model (AQUM OS24-26): model description and initial evaluation”, Geosci. Model Dev., 6, pp. 353–372, 2013. doi:10.5194/gmd-6-353-2013.

[22] T. Chai, H.C. Kim, P. Lee, D. Tong, L. Pan, Y. Tang, J. Huang, J. McQueen, M. Tsidulko, and I. Stajner, “Evaluation of the United States National Air Quality Forecast Capability experimental real-time predictions in 2010 using Air Quality System ozone and NO2 measurements”, *Geosci. Model Dev.,* 6, pp. 1831– 1850, 2013. doi:10.5194/gmd-6-1831-2013.

[23] Gülden and N. Güler, “A Study on Multiple Linear Regression Analysis”, *Social and Behavioral Sciences,* 106: pp. 234–240, 2013. DOI:10.1016/j.sbspro.2013.12.027.

[24] S. Fortmann-Roe, “Understanding the Bias-Variance Tradeoff”, [Online], 2012. Available: https://bit.ly/3AeuRip

[25] L. Breiman, “Better subset regression using the nonnegative garrote”, *Technometrics*, 37, pp. 373–384, 1995.

[26] P.E. Gill, W. Murray and M.A. Saunders, “Users guide for SQOPT 7: A Fortran package for large-scale linear and quadratic programming”, *Technical Report NA 05-1, Department of Mathematics, University of California*, SanDiego, 2005.