MA432 Final Project

Climate Data Analysis Using Ridge and Lasso Regression

Daniel Khalil

Computer Science

Abstract

Choosing regression algorithms for the right problem is important to maintain the accuracy of your approximation. This means comparing algorithms together is useful for understanding the effectiveness of specific models when applied to a specific problem. In this report we compared Lasso Regression and Ridge Regression techniques on spatially approximating ice sheet elevations given coordinates. This ultimately found that Ridge and Lasso regression applied to a dataset with only coordinates and heights, perform very similarly. The difference in performance between he two algorithms is small. Although lasso regression does perform slightly better than ridge if you add more features to the coordinate data.

Introduction

This report sets out to determine the functional differences between Lasso and Ridge regression, and their effective performance of spatially approximating ice sheet elevations, given latitude and longitude values for a specific year. This work is based off ALPS (A Unified Framework for Modeling Time Series of Land Ice Changes) [1][2], as it uses the modeled time series generated from the work done in this paper and creates a spatial approximation at a fixed time in 2006 using the framework developed from their work. Many compare the difference between Ridge Regression and Lasso Regression as the only difference between the two is that Lasso regression can completely disregard or zero out

coefficients, and ridge can only come close to zero. This means they are similar, but they are used for very different problems.

Connection to Linear Algebra

Climate Data itself may not have much to do with linear algebra, but the analysis or approximation of certain data will absolutely involve a level matrix math. For this project, the goal was to predict a height value given a latitude and longitude coordinate. One of the most well-known ways to perform an effective prediction of dependent variables given independent variables is through the utilization of regression models. The two regression models employed in this project are Ridge Regression and Lasso Regression. Both are modifications to the Linear Regression algorithm that remedy issues Linear Regression models may not be built to solve.

Linear Regression is a Machine Learning Algorithm that is utilized to construct a function approximation based on a set of input and output data. To do this the algorithm requires initially a set of data points correlating to independent and dependent variables. These data points are normally split up into a matrix X of independent variables or features, and a column vector y of dependent variables or target values. Data in rows X correspond to data in the same row from y.

$$\textbf{Feature Vectors:} \quad \boldsymbol{x_1, x_2, \ldots, x_N} \in \mathbb{R}^{\boldsymbol{2}}$$

$$\textbf{Target Values:} \quad \boldsymbol{y_1, y_2, \ldots, y_N} \in \mathbb{R}$$

X and y are used as the parameters of a function to calculate the estimator β. The estimator is a vector generated from X and y containing regression coefficients that when dotted to any set of independent variables matching the row dimension of X can produce a vector of predicted values. The most common estimator used for Linear Regression is called Ordinary Least Squares (OLS).

OLS requires a goal of finding numbers $\beta_0, \beta_1, \ldots, \beta_d \in \mathbb{R}$ such that

$$\boldsymbol{y_i \approx \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_d x_{id} - y_i}$$

The goal of linear regression is to minimize the cost of the function below to construct an optimal β.

$$\boldsymbol{minimize\ \beta \sum_{i=1}^{N} \frac{1}{2}(\beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_d x_{id} - y_i)^2}$$

This can be rewritten in the notation

$$\boldsymbol{minimize\ \beta \qquad \frac{1}{2}||X\beta - y||_2^2}$$

$$\boldsymbol{X = \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1d} \\ \vdots & \vdots & \vdots & \vdots \\ x_{N1} & x_{N2} & \ldots & x_{Nd} \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}}$$

In the end this problem can be simplified into this function of β given the parameters X and y.

$$\boldsymbol{\beta = (X^T X)^{-1} X^T y}$$

Given the calculated estimator β, and any matrix $\bar{X}$ with row length equivalent to X, a prediction can be made using the function

$$\boldsymbol{p(\bar{X}) = \bar{X}\beta}$$

The main difference between basic Linear Regression and its derivations is the way models calculate the β.

Ridge Regression

Ridge Regression is a regression model that is used for analyzing data with multicollinearity (when independent variables are highly correlated with one or more other independent variables in a multiple regression equation) [3]. Using regular linear regression's (least squares) cost function is in-efficient for estimating multicollinear data as the variance is high making the prediction skewed or far from the correct value. Ridge regression tries to mediate this issue by adding bias to this cost function, to make predictions more reliable when dealing with multicollinearity. The estimator for ridge regression applies a modification to the traditional the Linear Regression OLS estimator through an addition of L2 regularization to the cost function. This addition would look like this:

$$minimize\ \boldsymbol{\beta} \qquad \frac{\mathbf{1}}{\mathbf{2}}||\boldsymbol{X\beta} - \boldsymbol{y}||_{\mathbf{2}}^{\mathbf{2}} + \frac{\boldsymbol{\alpha}}{\mathbf{2}}||\boldsymbol{\beta}||_{\mathbf{2}}^{\mathbf{2}}$$

The solution to this optimization problem by setting the gradient of this equation equal to 0, and solving for β:

$$\boldsymbol{X^T(\,X\beta - y) + \alpha\beta = 0}$$

$$\boldsymbol{\left(X^T X + \alpha I\right)\beta = X^T y}$$

$$\boldsymbol{\beta = \left(X^T\,X + \alpha I_p\right)^{-1} X^T y}$$

This k value is Other than the estimator differences all other steps are synonymous to Linear Regression models.

Lasso Regression

Lasso works differently from Ridge regression in that its goal is to make β sparse. This means that the goal of the algorithm is to reduce the number of non-zero elements in the

vector β. Therefore, running Lasso Regression will essentially eliminate features that are less important to the contribution to y_i. To do this a L1 regularization term must be applied to the optimization problem.

$$\boldsymbol{minimize\ \beta} \qquad \frac{\mathbf{1}}{\mathbf{2}}||\boldsymbol{X\beta} - \boldsymbol{y}||_{\mathbf{2}}^{\mathbf{2}} + \boldsymbol{\alpha}||\boldsymbol{\beta}||_{\mathbf{1}}$$

This addition makes things less straight forward to compute since the term $\alpha||\beta||_1$ is not differentiable. This optimization problem can be solved in many ways but for this paper we will use the proximal gradient of the L1 norm method. This method is used to solve non-differentiable convex optimization problems. With this algorithm a function can, be constructed to shrink components of β towards the origin per iteration by using the proximal operator of R (β). This proximal operator contains a parameter $\bar{\alpha}$ which determines the shrinkage size per iteration.

$$\beta^{t+1} = prox_{\lambda R}(\beta^t - \alpha\nabla L(\beta^t))$$

$$prox_{\bar{\alpha}R}(\hat{\beta}) = \arg\min_{\beta} R(\beta) + \frac{1}{2\bar{\alpha}}\left\|\beta - \widehat{\beta}\right\|_2^2$$

$$prox_{\bar{\alpha}R}(\hat{\beta}) = \arg\min_{\beta} \alpha\|\beta\|_1 + \frac{1}{2\bar{\alpha}\alpha}\left\|\beta - \widehat{\beta}\right\|_2^2$$

This function will run for an optimal number of iterations and can slowly approximate β. Like Ridge Regression and Linear Regression given any input X dotted to this β an approximate solution can be found related to the data points fitted to the regression model.

Methodology

This section is a description of the process taken to compare both ridge and lasso regression's performance on their spatial approximation of Greenland's ice sheet in 2006. The following two sections will describe how we created the dataset of coordinate height

pairs in 2006 using the (Approximation by Localized Penalized Splines) ALPS framework, as well as the experimental design which includes test cases that will provide context to the ranking of these two models.

1	3000005	19	490934.6	7361392	1584.529
2	3000005	19	490934.6	7361392	1584.529
3	3000005	19	490934.6	7361392	1584.529
4	3000005	19	490934.6	7361392	1584.529
5	3000005	19	490934.6	7361392	1584.529
6	3000005	19	490934.6	7361392	1584.529
7	3000005	19	490934.6	7361392	1584.529
8	3000005	19	490934.6	7361392	1584.529
9	3000005	19	490934.6	7361392	1584.529
10	3000005	19	490934.6	7361392	1584.529
11	3000005	19	490934.6	7361392	1584.529
12	3000005	19	490934.6	7361392	1584.529
13	3000005	19	490934.6	7361392	1584.529
14	3000005	19	490934.6	7361392	1584.529
15	3000005	19	490934.6	7361392	1584.529
16	3000005	19	490934.6	7361392	1584.529
17	3000005	19	490934.6	7361392	1584.529
18	3000005	19	490934.6	7361392	1584.529
19	3000005	19	490934.6	7361392	1584.529
20	3000026	31	444591.8	7419801	2196.752
21	3000026	31	444591.8	7419801	2196.752
22	3000026	31	444591.8	7419801	2196.752
23	3000026	31	444591.8	7419801	2196.752
24	3000026	31	444591.8	7419801	2196.752
25	3000026	31	444591.8	7419801	2196.752
26	3000026	31	444591.8	7419801	2196.752
27	3000026	31	444591.8	7419801	2196.752
28	3000026	31	444591.8	7419801	2196.752
29	3000026	31	444591.8	7419801	2196.752
30	3000026	31	444591.8	7419801	2196.752
31	3000026	31	444591.8	7419801	2196.752
32	3000026	31	444591.8	7419801	2196.752
33	3000026	31	444591.8	7419801	2196.752
34	3000026	31	444591.8	7419801	2196.752
35	3000026	31	444591.8	7419801	2196.752
36	3000026	31	444591.8	7419801	2196.752
37	3000026	31	444591.8	7419801	2196.752
38	3000026	31	444591.8	7419801	2196.752

Figure 1: *Table of selected entry IDs.*

x	y	height
490934.6	7361392	2.873002
444591.8	7419801	0.979612

Figure 2: *Height approximation for 2006.*

Dataset Prep

The climate analysis time series dataset (43012 entries) is composed of (2789) sets of IDs with each corresponding to a pair of singular coordinates (x, y). There are multiple entries per ID with data denoting the coordinate pairs, ice height, and thicknesses, at different times. This means that (given a particular coordinate) a regression model can approximate ice height with time as a parameter. The model that can approximate heights given a time is the (ALPS) framework, which was provided by Dr. Prashant Shekhar. Since the goal of this project is to give a spatial approximation for 2006, we created an algorithm that executed the above model for each individual time series ID, producing a new dataset with (2789) entries, and containing columns (X-coordinate, Y-coordinate, Height).

This dataset can now be fit into any regression model to approximate a height given an x and y coordinate pair.

Experimental Process

There will be 3 different required tests for this report to determine the performance of Lasso and Ridge regression when applied to the dataset. The first test will calculate the average mean-squared-error while using the same arbitrary α parameter of 1 with the following split for training data and testing data for Lasso, Ridge, and plain Linear Regression. These test cases will split percentages of training and testing data from the 2789 data points generated from the Climate Analysis ALPS section above.

Test Case ID	Training	Testing
1	50%	50%
2	55%	45%
3	60%	40%
4	65%	35%
5	70%	30%
6	75%	25%
7	80%	20%
8	85%	15%
9	90%	10%
10	95%	5%

Figure 3: *Training/testing data split.*

The next test will modify lasso regression by adding more features to its data. The idea behind doing this is that Lasso Regression seems to be constructed with large numbers of features in mind, so it might outperform others when fitted with more features. This could have interesting effects since we would also increase the number of features for the other 2 models and see if they lose performance or maintain similar values. This can be tracked

by taking the Average MSE of each model through using the first test case and averaging all the results for different feature lengths.

Feature Length	Lasso	Ridge
2		
100		
500		
1000		
5000		
10000		
50000		

Figure 4: *Test case with different feature lengths.*

Additional features will be created through a fixed random combination of coordinates. A random function will be generated per column to generate features from the first two columns.

$$X_{iN} = X_{1N} * rand(0-1) + X_{2N} * rand(0-1)$$

Lastly a test will be run to find the most optimal α values for each of these algorithms. The first test case will be executed with a suite of different α values per iteration, as shown in the table to the right.

α
0.001
0.01
0.1
1
10
100
1000

This will be the last test case execute in this report. These metrics explained above will be sufficient for a discussion on the performance of Lasso vs Ridge regression.

Results

The first test utilizes the standard 2 feature $\alpha = 1$. The initial Training/Testing Split data received wasn't very promising as it didn't show much of a difference in performance between Linear, Ridge, or Lasso Regression. But on a much smaller level there were significant differences. This preliminary test would be run for the next 2 test cases with different feature lengths and different α values.

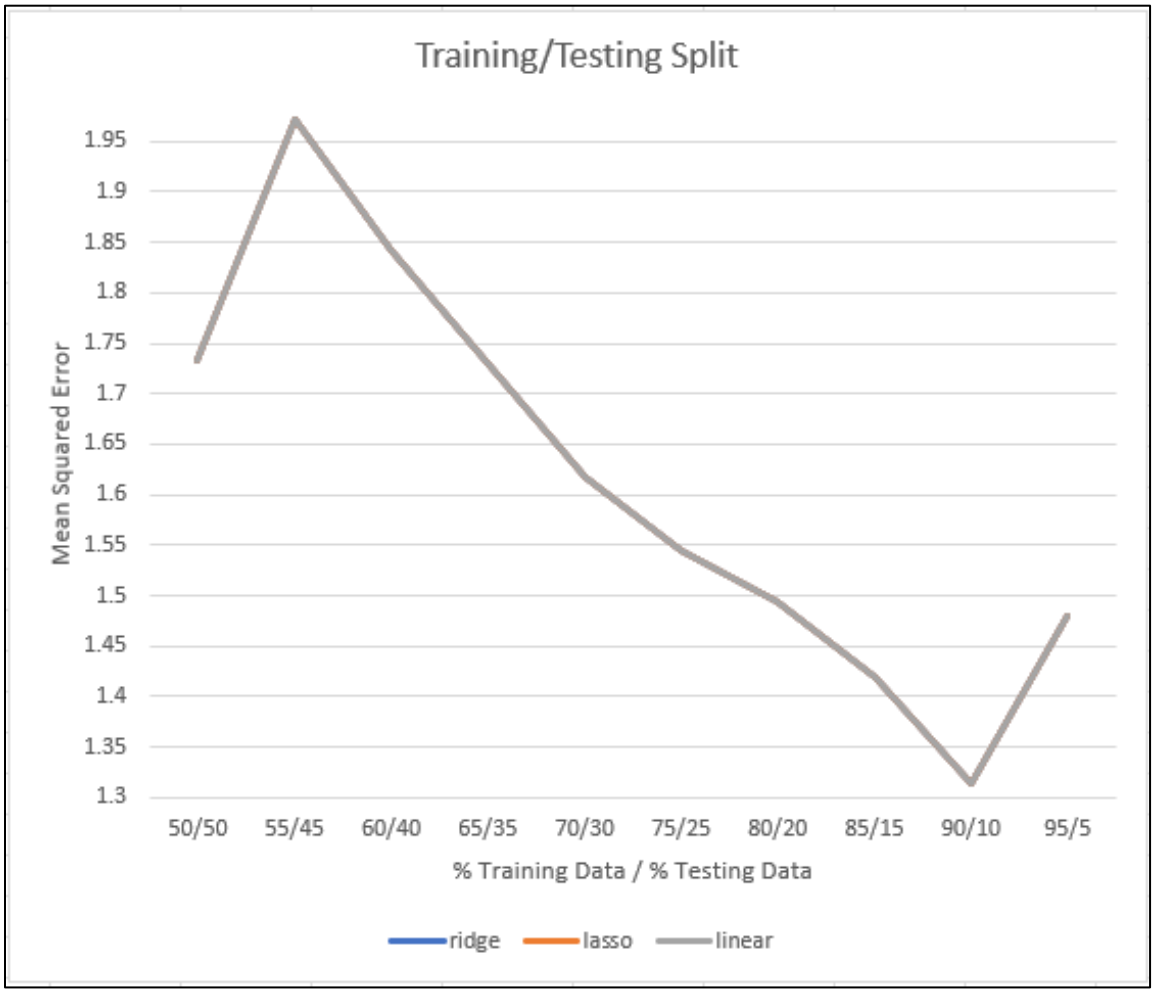
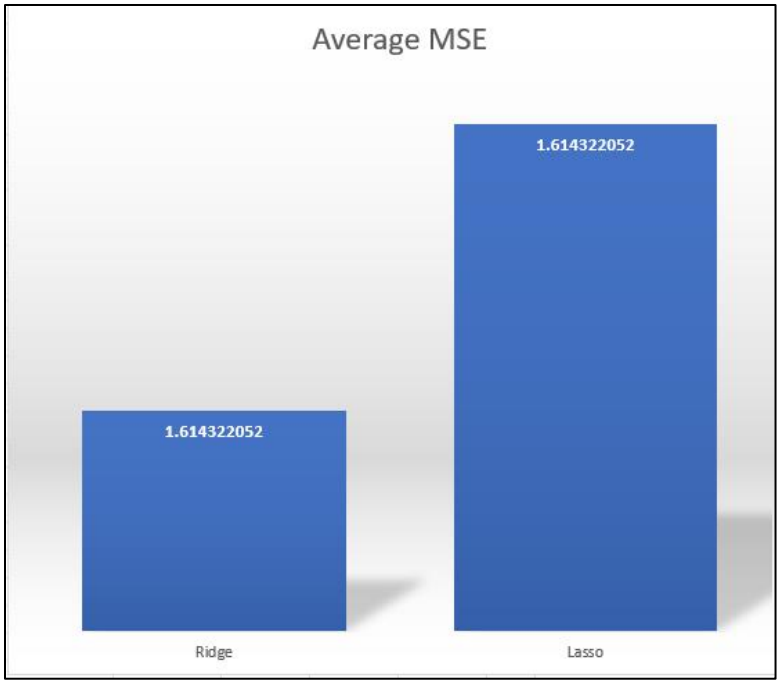


Figure 6: *Graph depicting results of Figure 3.*



When comparing the Average Mean Squared Error taken from each split from the above graph and averaged, Ridge outperforms lasso by a small margin. It is to be determined whether this difference is even worth mentioning but considering that Lasso Regression is built to handle larger feature sizes, and Ridge Regression is built to deal with multicollinearity, this set of features may not provide enough information for either regression algorithm to fully show their strengths or weaknesses.

Figure 7: *Difference in MSE for Ridge and Lasso.*

Average MSE of these algorithms compared to Linear Regression is more substantial than the difference between Ridge and Lasso Regression. Linear Regression had a 1000th of a point difference between Ridge and Lasso regression which proved that these regression techniques that branch off the old variant do outperform their predecessor.

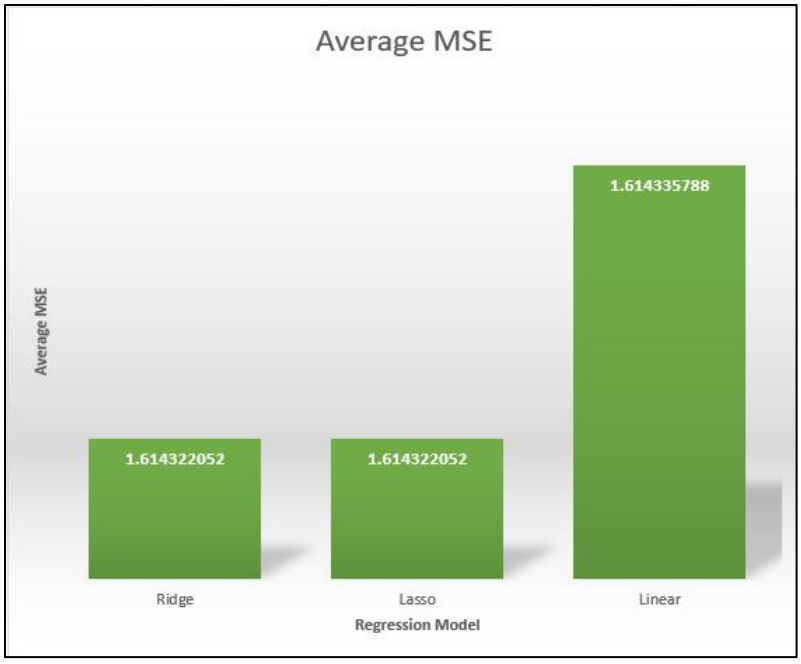


Figure 8: *Ridge and Lasso compared to Linear (MSE).*

The feature length test case-using α 's of 1 and varying feature lengths-showed an interesting behavior that proved that Lasso regression handles multiple features much more elegantly then Ridge or Linear Regression. Ridge regression outperformed Linear Regression until 10,000 features were applied to the dataset, and the jump from 5000, to 10,000 quadrupled the Mean Squared Error. Considering the first control tests showing minimal difference between Lasso and Ridge regression, this test highlights the power of L1 Normalization applied to the least squares function for these types of niche cases.

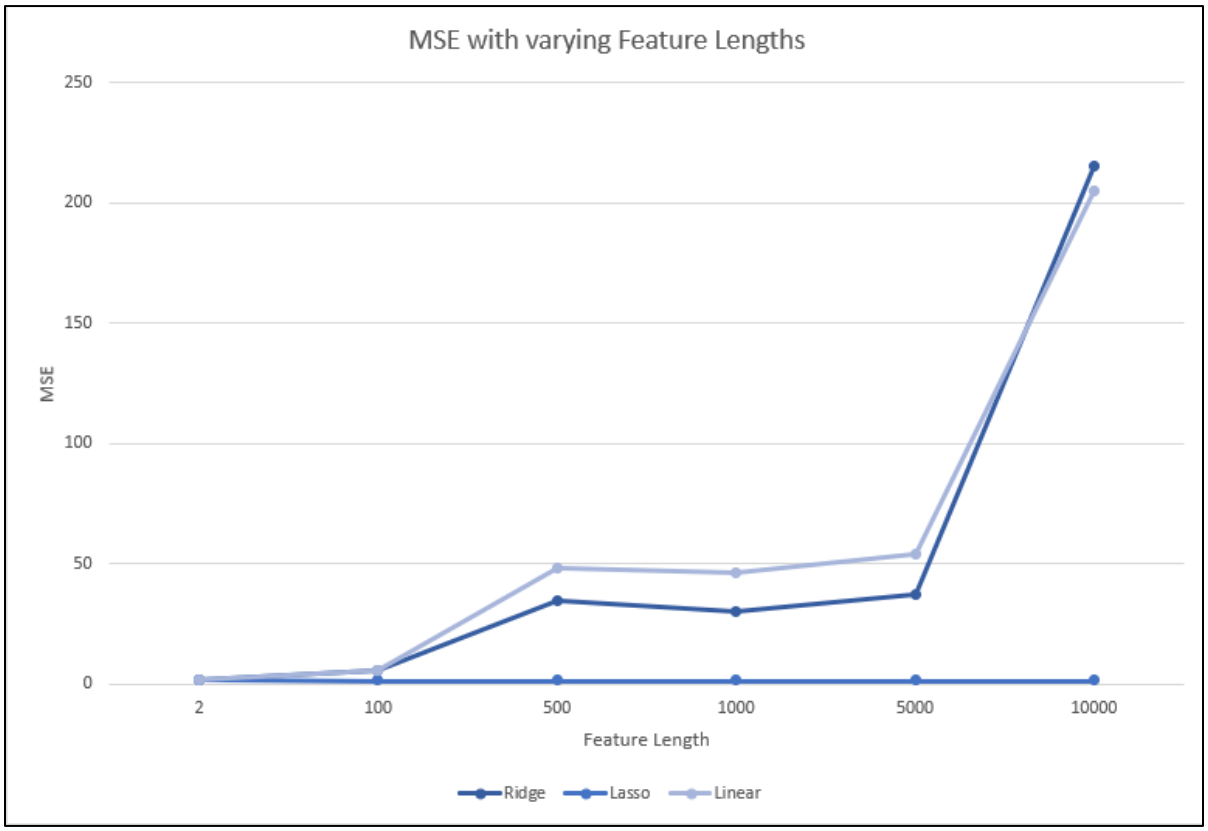


Figure 9: *Graph depicting MSE with varying feature lengths.*

Not only did Lasso regression not lose accuracy with an increase of 10,000 features, but it even improved the regression algorithms performance by a MSE of 0.02531.

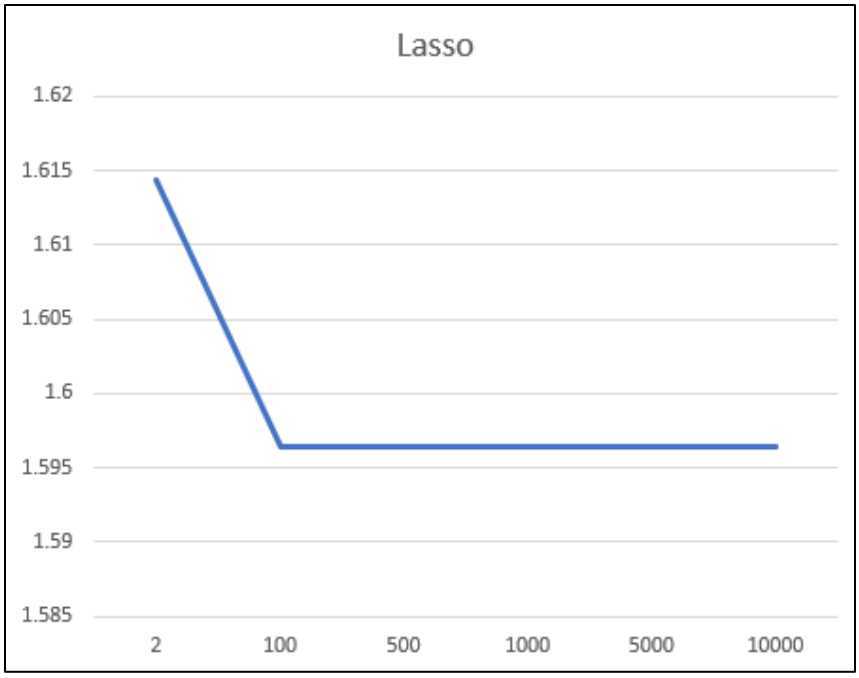


Figure 10: *Graph depicting MSE with varying feature lengths for Lasso Regression.*

Finally, the α value test case using varying α and 2 features, showed that any α value above 10, worsens Lassos MSE performance within this particular dataset. The Ridge regression change was so minimal with changes in α that it seems changing the α value for this particular feature pick makes no difference in regards to the accuracy of Ridge Models.

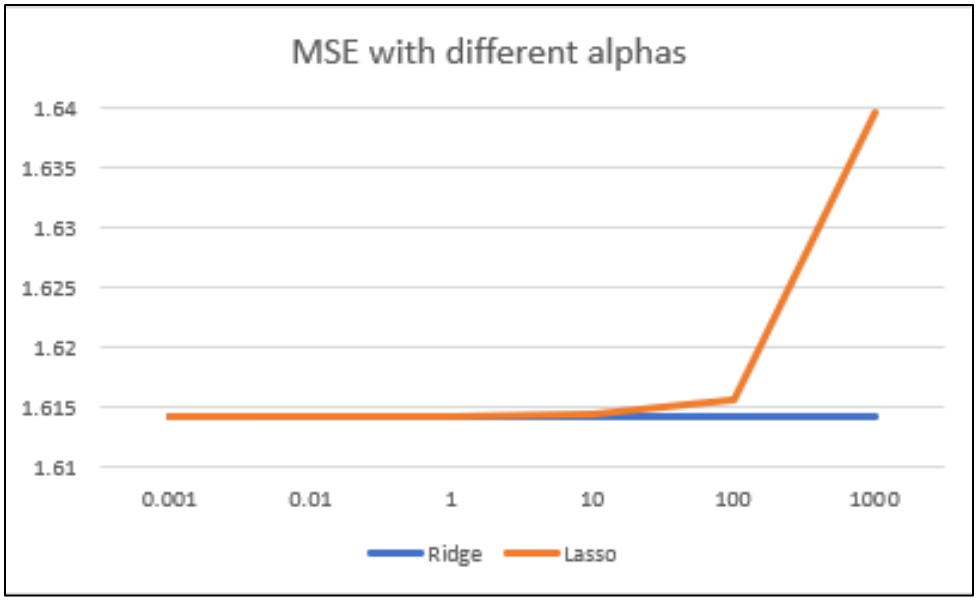


Figure 11: *MSE of Lasso and Ridge using various α-values.*

Discussion

It seems that (specifically for Ridge and Lasso) no matter which regression algorithm is used, the resulting approximation will be mostly similar. There is minimal difference between the performance of Ridge Regression and Lasso Regression with this data. There may be some use in adding additional features to the dataset and using Lasso, as it improved initial performance by a small amount. Although the trade off is that it takes much longer to perform this calculation due to the increased feature length, making it computationally less effective. Overall, the best mean squared error received from both Lasso and Ridge Regression given a 95% training data 5% testing data split, was 1.479 which could possibly be outperformed by other more complex models. Overall, it seems that these two algorithms wouldn't be very useful when approximating a function for this data, as they are built to handle issues that this dataset does not contain.

[why is it useful?]

This information is useful to know so that

1. The reader has a better understanding of the applications of Ridge and Lasso Regression
2. The reader knows that these algorithms applied to this dataset is unfavorable.

If there were more time to work on this, it would have been nice to test other Regression models on this dataset.

References

[1] Shekhar, P., Csatho, B., Schenk, T., Roberts, C., & Patra, A. K. (2021). Alps: A unified framework for modeling time series of land ice changes. *IEEE Transactions on Geoscience and Remote Sensing*, *59*(8), 6466–6481. https://doi.org/10.1109/tgrs.2020.3027190

[2] The problem of multicollinearity. (n.d.). *Understanding Regression Analysis*, 176–180. https://doi.org/10.1007/978-0-585-25657-3_37

[3] *The problem of Multicollinearity | SpringerLink*. (n.d.). Retrieved December 5, 2021, from https://link.springer.com/chapter/10.1007%2F978-0-585-25657-3_37.

[4] Sneiderman, R. (2020, November 6). *From linear regression to ridge regression, the Lasso, and the elastic net*. Medium. Retrieved December 5, 2021, from https://towardsdatascience.com/from-linear-regression-to-ridge-regression-the-lasso-and-the-elastic-net-4eaecaf5f7e6.

[5] *Python: Implementation of polynomial regression*. GeeksforGeeks. (2021, October 4). Retrieved December 5, 2021, from https://www.geeksforgeeks.org/python-implementation-of-polynomial-regression/.