Shrinkage Methods and Sparsity

Agniv Chakraborty, Avijit Saha,
Archishman Mukherjee, Tuhin Saha

Statistical Methods 4 Project
Guided by Prof. Arnab Chakraborty
Applied Statistics Unit

Outline

1. Introduction
2. Cross Validation
3. Linear Regression: Ordinary Least Squares
4. Disadvantages of OLS
5. Penalised Regression
6. Ridge
7. LASSO
8. Lasso vs Ridge
9. Why Lasso is more sparse than Ridge?
10. Naive Elastic Net
11. Example of a Simulated Multi-collinear Dataset
12. Example of a Simulated Overfitting Dataset
13. Example of a Real Dataset
14. Conclusion
15. References

Introduction

Why Shrinkage?

Classically, the problem of regression has involved more data points/subjects than covariates, which were assumed to be not very correlated with each other. However, with the modern data explosion, we see that in many cases the classic least-squares solution is not adequate. In this project, we are going to go through the motivation and mathematical foundation behind **shrinkage methods**. Essentially, what we do here is "shrink" the regression coefficient estimates towards zero. The main idea behind this is that we can significantly reduce variance as well as number of significant covariates. The two most well-known shrinkage methods are Ridge Regression and Lasso Regression, which fall under Penalised Linear Regression.

An extra tool

However, before we go into all this, we will look at cross-validation, a tool we will use frequently in this project to justify and evaluate our various models.

Cross Validation

The purpose of the method

Cross-validation is a method that uses different portions of the data to test and train a model on different iterations. This is mainly used in situations where the goal is the prediction from new data, and we want to estimate how accurately our predictive model will perform in practice. Here we consider k-fold cross-validation.

Cross Validation

The purpose of the method

Cross-validation is a method that uses different portions of the data to test and train a model on different iterations. This is mainly used in situations where the goal is the prediction from new data, and we want to estimate how accurately our predictive model will perform in practice. Here we consider k-fold cross-validation.

Procedure

The entire data is randomly divided into k groups of roughly equal size. Then we separate any one group as *test* data, and the remaining $k-1$ we group together as *train* data. We fit our model (through linear regression or any other method) on the *train* data, and then test it on the *test* data. We evaluate its performance through some metric, typically the mean-square of residuals for linear regression, and keep that value stored. We repeat this procedure k times (each of the k parts gets to be *test* once, and the rest in *train*), and get k models with k different evaluation metrics. Then we summarise the prediction skill of the overall model by reporting the average, or a maximum of these k scores.

Ordinary Least Squares

Recap

In OLS, we minimise the objective function $\sum_{i=1}^{n} (Y_i - \beta^T X_i)^2 = (Y - X\beta)^T (Y - X\beta)$ with respective to $\beta \in \mathbb{R}^p$. The exact solution (assuming X is full column rank) is:

$$\hat{\beta}_{OLS} = (X^T X)^{-1} X^T Y$$

This solution has various nice properties, as detailed below:

Ordinary Least Squares

Recap

In OLS, we minimise the objective function $\sum_{i=1}^{n}(Y_i - \beta^T X_i)^2 = (Y - X\beta)^T(Y - X\beta)$ with respective to $\beta \in \mathbb{R}^p$. The exact solution (assuming X is full column rank) is:

$$\hat{\beta}_{OLS} = (X^T X)^{-1} X^T Y$$

This solution has various nice properties, as detailed below:

Properties of OLS Solution

If we assume the following model:

$$Y = X\beta + \varepsilon$$

where $\varepsilon \sim N(0, \sigma^2 I_p)$. Under this model, $\hat{\beta}_{MLE} = \hat{\beta}_{OLS}$. Further, $E(\hat{\beta}_{OLS}) = \beta$ (thus $\hat{\beta}$ is an unbiased estimator of β), $Var(\hat{\beta}_{OLS}) = \sigma^2(X^T X)^{-1}$ (here variance means variance-covariance matrix). Throughout X_i are assumed to be non-random.

Disadvantages of OLS: Overfitting

Overfitting

Overfitting is a situation in regression/classification where the model fits the training dataset too closely, hence the moniker “over” fitting. In this situation, apart from simply describing the relationship between response and covariates, the model captures the random error or “noise” in the training data as well. This is a situation not desirable, as it means that the model may not be able to fit additional data or predict future observations reliably.

Disadvantages of OLS: Overfitting

Overfitting

Overfitting is a situation in regression/classification where the model fits the training dataset too closely, hence the moniker "over" fitting. In this situation, apart from simply describing the relationship between response and covariates, the model captures the random error or "noise" in the training data as well. This is a situation not desirable, as it means that the model may not be able to fit additional data or predict future observations reliably.

Recognising Overfitting

As in k-fold cross-validation, we divide the total data into a *test* part and a *train* part. We fit our model to the *train* set through OLS as per usual, then test the model against the *test* dataset. If the model performs significantly better for the *train* set compared to the *test* set, that is overfitting. This can be quantified through the mean square of residuals, or multiple R-squared, among many other metrics. We demonstrate this through a simulated example.

Disadvantages of OLS: Multicollinearity

Multicollinearity

Multicollinearity refers to the situation when there is a strong correlation between certain covariates. As a result, it becomes redundant to include both covariates, and the corresponding coefficient estimates become highly sensitive to change upon a small change in the data. For example, $3X_1 + 5X_2$ and $7X_1 + X_2$ both describe very similar things Mathematically, this means that the distance between the vector Y and its orthogonal projection on $\mathcal{C}(X)$ is very less. In this case, X^TX becomes "close" to being singular, as a result $(X^TX)^{-1}$ has large entries. Even if we use a generalised inverse when $p > n$, these problems still persist. Since $\hat{\beta}$ and their variance are "directly proportional" to this large matrix, thus this explains why $\hat{\beta}$ has a tendency of exploding and being highly sensitive and far away from the true parameter. We demonstrate this with the help of simulated data.

Penalised Regression

Bias vs Variance Tradeoff

In regression, we always strive to estimate the true population parameter β as closely as possible. A common way to measure the ability of an estimator is its MSE (Mean Squared Error): $E((\hat{\beta}-\beta)^2) = \text{Bias}^2(\hat{\beta}) + \text{Var}(\hat{\beta})$. As we have seen, in OLS the bias term is zero, but the variance may blow up. If we are willing to accept a little bias, we might be able to control the β_j sufficiently so that the variance is reduced enough that the MSE ultimately decreases. This is called the **Bias-Variance Tradeoff**. It is near-impossible to globally minimise MSE across the parameter space, so instead, we look for the best in a family of solutions. We add a "penalty" to our usual loss function and then try to minimise it. This penalty is typically controlled by a tuning parameter λ, or equivalently a metric for the constraint k. We will see this in more detail in the coming slides.

Penalised Regression

Common Loss Functions (including the penalty)

- Best Subset Model: $\sum_{i=1}^{n} (Y_i - \beta^T X_i)^2 + \lambda \sum_{j=1}^{p} \mathbb{I}(\beta_j \neq 0)$
- LASSO Regression: $\sum_{i=1}^{n} (Y_i - \beta^T X_i)^2 + \lambda \sum_{j=1}^{p} |\beta_j|$
- Ridge Regression: $\sum_{i=1}^{n} (Y_i - \beta^T X_i)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$
- Naive Elastic Net: $\sum_{i=1}^{n} (Y_i - \beta^T X_i)^2 + \lambda \left(\alpha \sum_{j=1}^{p} |\beta_j| + (1 - \alpha) \sum_{j=1}^{p} \beta_j^2 \right)$

The Ridge Regression

Regularisation

To keep the regression results free of the scaling factor, by convention:

- **X** is assumed to be standardized (mean 0, unit variance)
- **Y** is assumed to be centered (mean 0)

This means that the intercept term β_0 is useless, so we ignore it from this point onwards.

The Ridge Regression

Regularisation

To keep the regression results free of the scaling factor, by convention:

- **X** is assumed to be standardized (mean 0, unit variance)
- **Y** is assumed to be centered (mean 0)

This means that the intercept term β_0 is useless, so we ignore it from this point onwards.

Ridge Constraint

We wish to minimise $\sum_{i=1}^{n}(Y_i - \beta^T X_i)^2$ subject to $||\beta||_2^2 = \sum_{j=1}^{p} \beta_i^2 \leq k$ for some fixed $k > 0$. This is $\iff$ to minimising the loss function $\sum_{i=1}^{n}(Y_i - \beta^T X_i)^2 + \lambda \left(\sum_{j=1}^{p} \beta_i^2 \right)$ for some fixed $\lambda > 0$. If we increase k, that means more 'freedom' for the β_j's to explode, thus it is closer to the OLS solution, which corresponds to λ being very small and close to 0.

Conversely, if λ is very large, all the $\hat{\beta}_j$ become close to 0 (the null model).

The Ridge Regression

Solution

In matrix terms, we need to minimise $(Y - X\beta)^T(Y - X\beta) + \lambda\beta^T\beta$. Differentiating with respect to β, we have:

$$-2X^T(Y - X\beta) + 2\lambda\beta = 0 \implies \boxed{\hat{\beta}_\lambda = (X^TX + \lambda I_p)^{-1}X^TY}$$

Clearly, this is not an unbiased estimator because of λ. This solves the multicollinearity problem as well since λ will automatically be chosen in a way that $(X^TX + \lambda I)$ becomes invertible. This solution can be obtained as the usual OLS solution of an extended dataset like: $\displaystyle\sum_{i=1}^{n}(Y_i - \beta^TX_i)^2 + \sum_{j=1}^{p}(0 - \sqrt{\lambda}\beta_i)^2$. In practical situations, we utilise **Singular Value Decomposition** of $X = UDV^T$ where U, V are orthogonal and D is diagonal.

The LASSO Regression

Lasso Constraint

We wish to minimise $\sum_{i=1}^{n}(Y_i - \beta^T X_i)^2$ subject to $||\beta||_1 = \sum_{j=1}^{p}|\beta_i| \leq k$ for some fixed $k > 0$. This is $\iff$ to minimising the loss function $\sum_{i=1}^{n}(Y_i - \beta^T X_i)^2 + \lambda\left(\sum_{j=1}^{p}|\beta_i|\right)$ for some fixed $\lambda > 0$. Again we have that λ and k have a loose inverse relation.

The LASSO Regression

Lasso Constraint

We wish to minimise $\sum_{i=1}^{n} (Y_i - \beta^T X_i)^2$ subject to $||\beta||_1 = \sum_{j=1}^{p} |\beta_i| \leq k$ for some fixed $k > 0$. This is $\iff$ to minimising the loss function $\sum_{i=1}^{n} (Y_i - \beta^T X_i)^2 + \lambda \left(\sum_{j=1}^{p} |\beta_i| \right)$ for some fixed $\lambda > 0$. Again we have that λ and k have a loose inverse relation.

Solution

Unfortunately, there exists no closed-form solution for $\hat{\beta}$ in LASSO. However, we can still comment on the solution. Like ridge, increasing λ (or decreasing t) after a certain point sends the solution to the null model. However, LASSO has a tendency of reducing coefficient estimates to 0 much faster than Ridge, which will be discussed later.

Lasso vs Ridge

Sparsity

Solution of a linear regression is said to be **Sparse** if the solution $\hat{\beta}$ has $\hat{\beta}_j = 0$ for many components j $\in \{1, 2, 3, \ldots, p\}$

Lasso vs Ridge

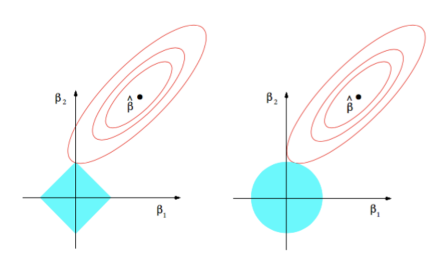
Sparsity

Solution of a linear regression is said to be **Sparse** if the solution $\hat{\beta}$ has $\hat{\beta}_j = 0$ for many components j $\in \{1, 2, 3, \ldots, p\}$

When sparsity is desirable?

- Sparsity corresponds to performing variable selection in the constructed linear model
- Sparsity provides a level of interpretability (beyond sheer accuracy)

Why Lasso is more sparse than Ridge?



The ellipses centred about $\hat{\beta_{OLS}}$ represent curves of constant RSS. As the ellipses expand away from $\hat{\beta_{OLS}}$, the RSS increases. The blue-shaded region represents the constrained regions of $\hat{\beta}_L$ and $\hat{\beta}_R$. The lasso and ridge coefficients are given by the first point an ellipse contacts the constraint region. Since ridge regression has a circular constraint with no sharp points, this intersection will generally not occur on an axis, and so the ridge regression coefficient estimates will be usually non-zero. However, the lasso constraint has "corners" at each of the axes, and so the ellipse will often intersect the constraint region at an axis. When this occurs, one of the coefficients will be zero and in higher dimensions, many of the coefficients may be zero simultaneously.

Naive Elastic Net

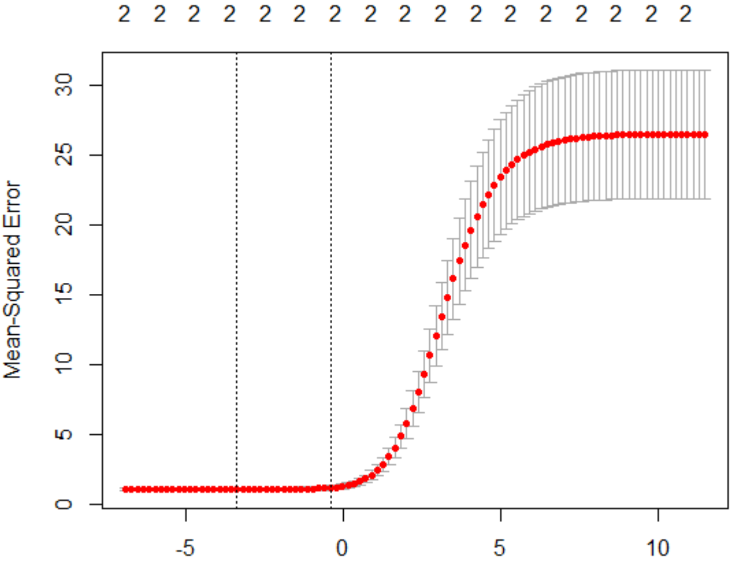
Best of both worlds

Since their proposal, both Ridge and LASSO have been criticised for various reasons. For example, we already know that Lasso can make coefficients disappear, thus making the interpretability a lot easier, as compared to Ridge where all the coefficients are present and are only made close to zero. On the other hand, Lasso is more time-consuming as compared to Ridge, simply due to the objective functions being based on L1 and L2 norms, respectively. Sometimes Lasso may be too selective and ignore moderately significant covariates. So Naive Elastic Net is a compromise between the two. In this case, the loss function is simply a convex combination of that of the Ridge and LASSO penalty functions:

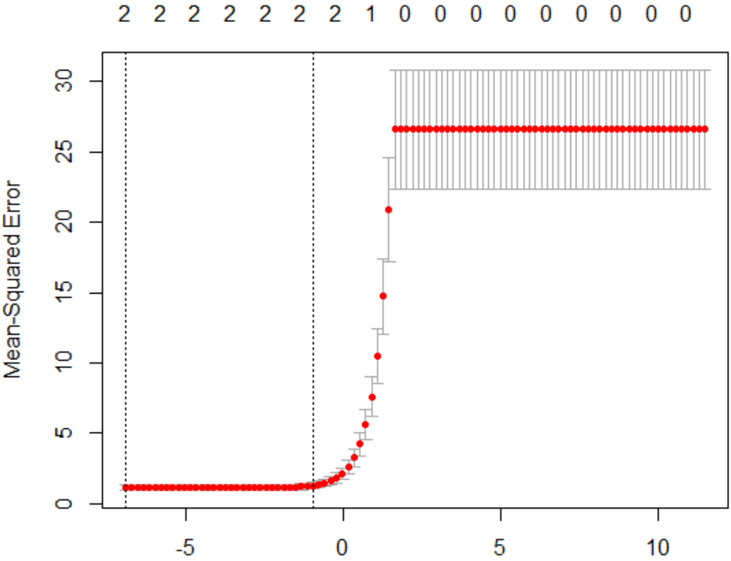
$$\alpha||\beta||_1 + (1-\alpha)||\beta||_2^2$$

So here we have an additional layer for cross-validation, the optimal values for both λ and α need to be found instead of just λ as before.

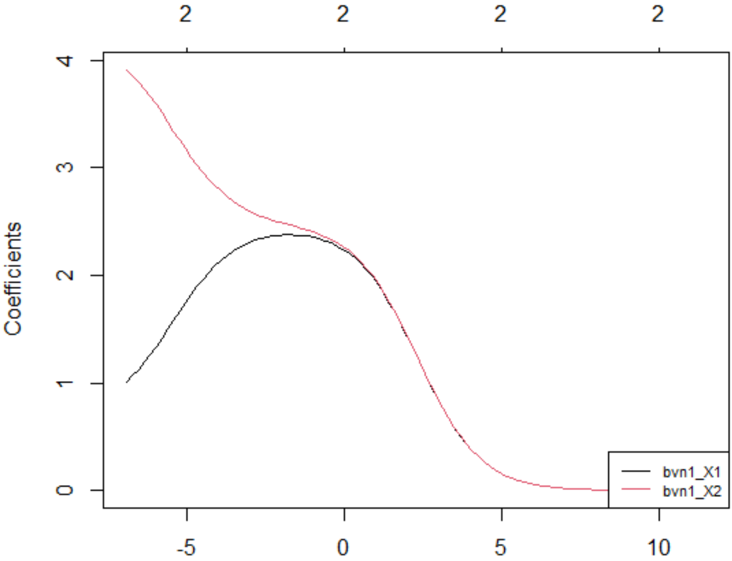
MSE vs log(λ) plot - Ridge (Simulated Multi-collinear Dataset)



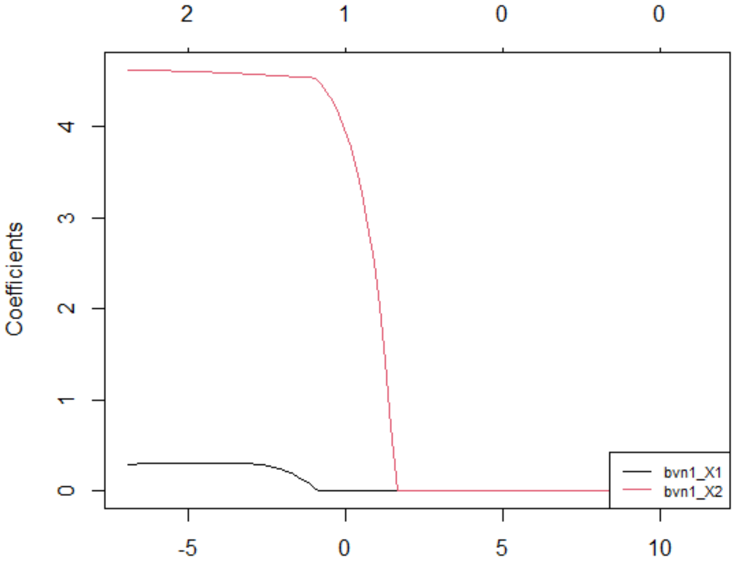
MSE vs log(λ) plot - Lasso (Simulated Multi-collinear Dataset)



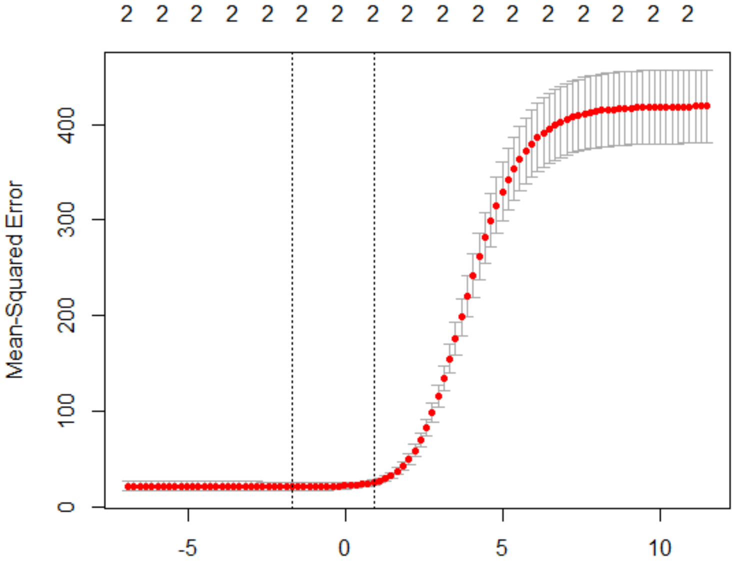
Coefficient vs log(λ) plot - Ridge (Simulated Multi-collinear Dataset)



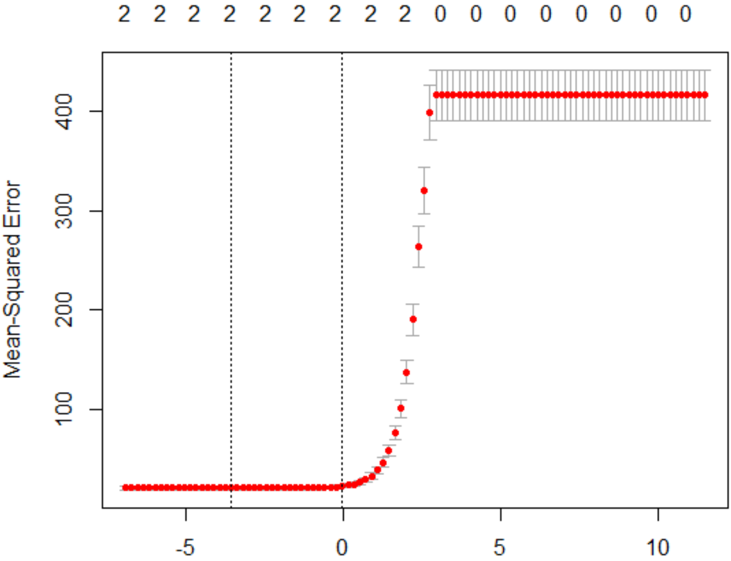
Coefficient vs log(λ) plot - Lasso (Simulated Multi-collinear Dataset)



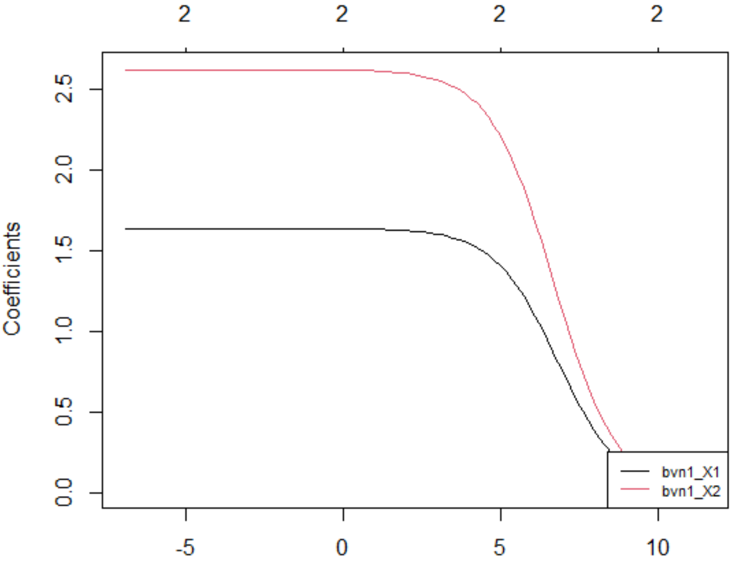
MSE vs log(λ) plot - Ridge (Simulated Overfitting Dataset)



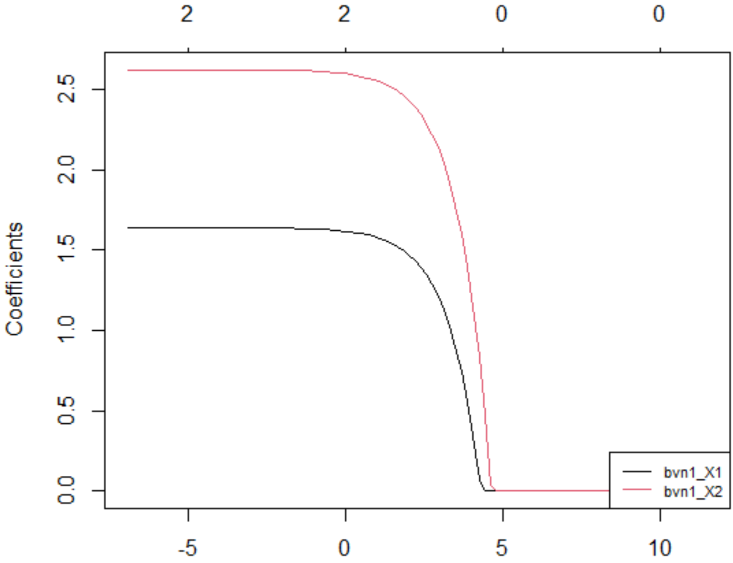
MSE vs log(λ) plot - Ridge (Simulated Overfitting Dataset)



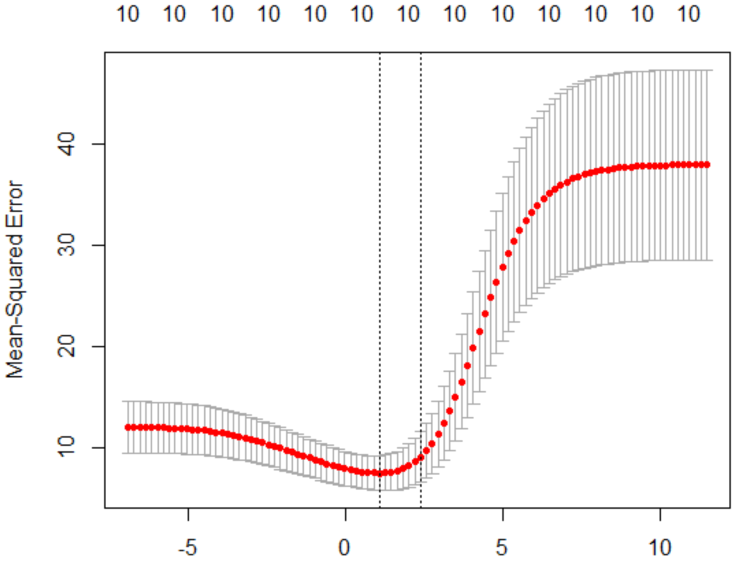
Coefficient vs log(λ) plot - Ridge (Simulated Overfitting Dataset)



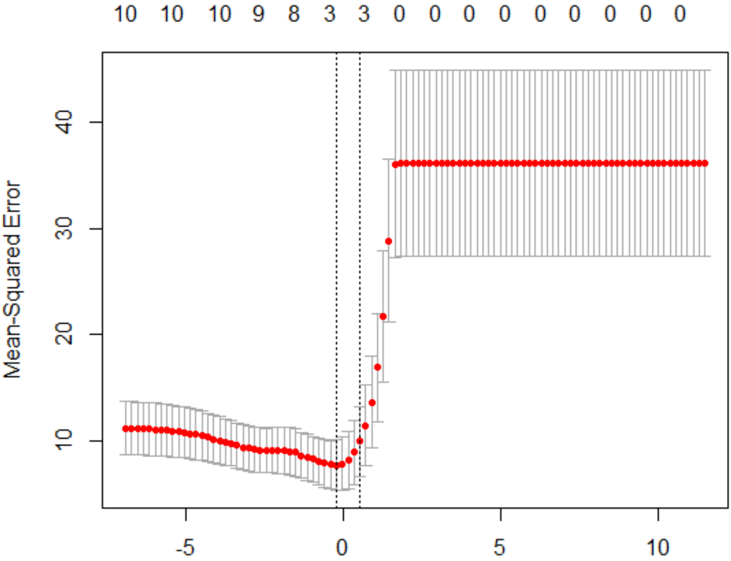
Coefficient vs log(λ) plot - Lasso (Simulated Overfitting Dataset)



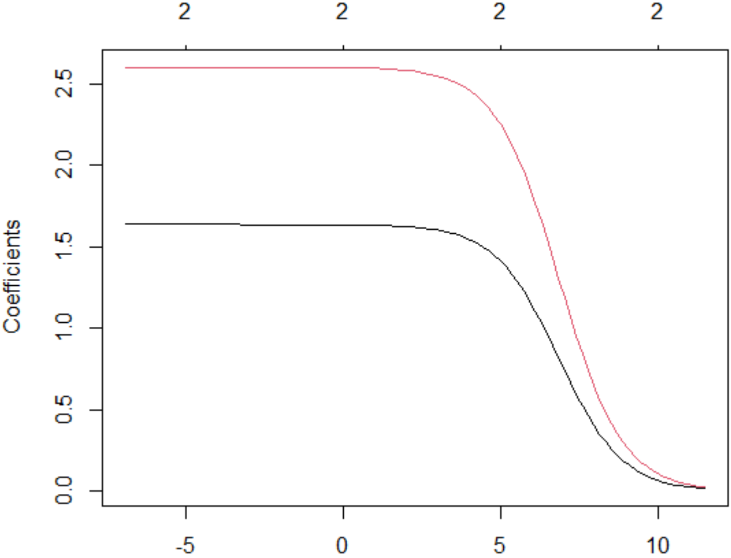
MSE vs log(λ) plot - Ridge (Real Dataset)



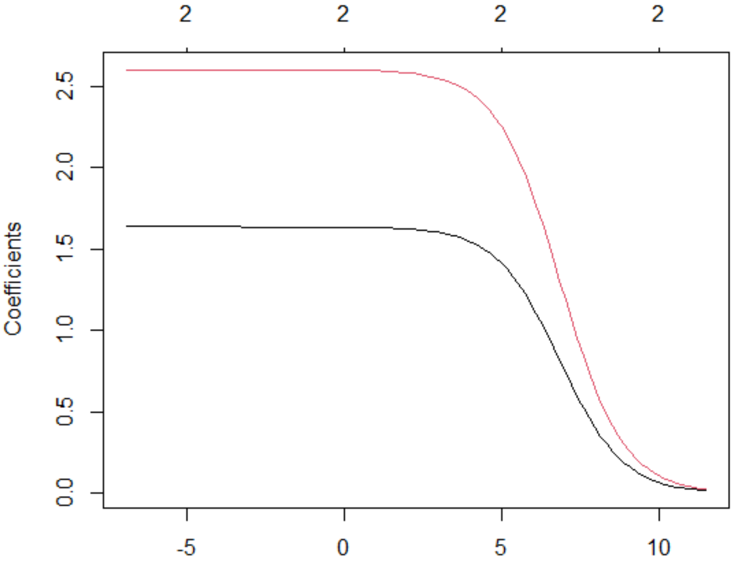
MSE vs log(λ) plot - Lasso (Real Dataset)



Coefficient vs log(λ) plot - Ridge (Real Dataset)



Coefficient vs log(λ) plot - Lasso (Real Dataset)



Conclusion

A standard least squares model tends to have some variance in it, i.e. this model won't generalize well for a data set different from its training data. Regularization, significantly reduces the variance of the model, without a substantial increase in its bias. So the tuning parameter λ, used in the regularization techniques described above, controls the impact on bias and variance. As the value of λ rises, it reduces the value of coefficients, thus reducing the variance. To a point, this increase in λ is beneficial as it is only reducing the variance(hence avoiding overfitting), without losing any essential properties in the data. But after a certain value, the model starts losing important properties, giving rise to bias in the model and thus underfitting. Therefore, the value of λ should be carefully selected.

References

An Introduction to Statistical Learning with Applications in R

Gareth James, Daniela Witten, Trevor Haste, Robert Tibshirani
Springer

Regularization: Ridge Regression and the LASSO

Statistics 305: Autumn Quarter 2006/2007

Sparsity, the Lasso, and Friends

Ryan Tibshirani (with Larry Wasserman)
Statistical Machine Learning, Spring 2017

Motor Trend Car Road Tests

https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/mtcars.html

Ridge Regression and Lasso

Prof. Dirk Neumann
Business Analytics, Albert-Ludwigs-Universitat Freiburg

References

Stat 383C: Statistical Modelling I, Lecture 10 - September 27, 2016

Prof. Purnamrita Sarkar, Carnegie Mellon University

Statistical Learning (BST 263), Lecture 11: Penalized Regression

Prof. Jeffrey W. Miller, Department of Biostatistics
Harvard T.H. Chan School of Public Health

Regression Shrinkage and Selection via the Lasso

Prof. Robert Tibshirani, Journal of the Royal Statistician Society, Series B (Methodological), Volume 58, Issue 1 (1996), 267-288

Regularization in Machine Learning

https://towardsdatascience.com/regularization-in-machine-learning-76441ddcf99a

Thank You

Grazie!