Linear Regression Regularization

Linear Regression Model -

Lab- 1- Estimating mileage based on features of a second hand car

Description – Sample data is available at
https://archive.ics.uci.edu/ml/datasets/Auto+MPG

The dataset has 9 attributes listed below that define the quality

1. mpg: continuous
2. cylinders: multi-valued discrete
3. displacement: continuous
4. horsepower: continuous
5. weight: continuous
6. acceleration: continuous
7. model year: multi-valued discrete
8. origin: multi-valued discrete
9. car name: string (unique for each instance)

Sol : Ridge_Lasso_Regression.ipynb

Regularising Linear Models (Shrinkage methods)

When we have too many parameters and exposed to curse of dimensionality, we resort to dimensionality reduction techniques such as transforming to PCA and eliminating the PCA with least magnitude of eigen values. This can be a laborious process before we find the right number principal components. Instead, we can employ the shrinkage methods.

Shrinkage methods attempt to shrink the coefficients of the attributes and lead us towards simpler yet effective models. The two shrinkage methods are :

1. Ridge regression is similar to the linear regression where the objective is to find the best fit surface. The difference is in the way the best coefficients are found. Unlike linear regression where the optimization function is SSE, here it is slightly different

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2$$

Linear Regression cost function

$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2$$

Ridge Regression with additional term in the cost function

1. The term λ is like a penalty term used to penalize large magnitude coefficients β_j when it is set to a high number, coefficients are suppressed significantly. When it is set to 0, the cost function becomes same as linear regression cost function

Regularising Linear Models (Shrinkage methods)

Why should we be interested in shrinking the coefficients? How does it help?

When we have large number of dimensions and few data points, the models are likely to become complex, overfit and prone to variance errors. When you print out the coefficients of the attributes of such complex model, you will notice that the magnitude of the different coefficients become large

Large coefficients indicate a case where for a unit change in the input variable, the magnitude of change in the target column is very large.

Coeff for simple linear regression model of 10 dimensions

1. The coefficient for cyl is 2.5059518049385052
2. The coefficient for disp is 2.5357082860560483
3. The coefficient for hp is -1.7889335736325294
4. The coefficient for wt is -5.551819873098725
5. The coefficient for acc is 0.11485734803440854
6. The coefficient for yr is 2.931846548211609
7. The coefficient for car_type is 2.977869737601944
8. The coefficient for origin_america is -0.5832955290166003
9. The coefficient for origin_asia is 0.3474931380432235
10. The coefficient for origin_europe is 0.3774164680868855

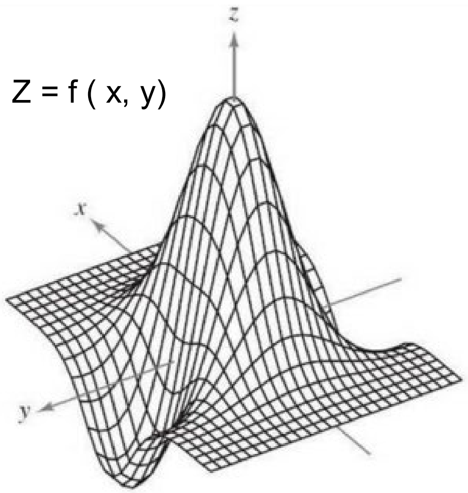
Ref: Ridge_Lasso_Regression.ipynb

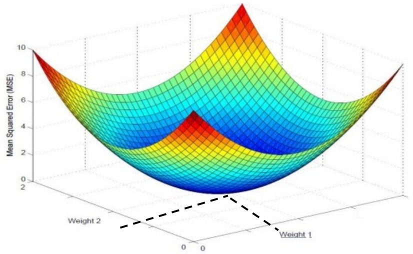
Coeff with polynomial features shooting up to 57 from 10

-9.67853872e-13 -1.06672046e+12 -4.45865268e+00 -2.24519565e+00 -
2.96922206e+00 -1.56882955e+00 3.00019063e+00 -1.42031640e+12 -
5.46189566e+11 3.62350196e+12 -2.88818173e+12 -1.16772461e+00 -
1.43814087e+00 -7.49492645e-03 2.59439087e+00 -1.92409515e+00 -
3.41759793e+12 -6.27534905e+12 -2.44065576e+12 -2.32961194e+12
3.97766113e-01 1.94046021e-01 -4.26086426e-01 3.58203125e+00 -
2.05296326e+00 -7.51019934e+11 -6.18967069e+11 -5.90805593e+11
2.47863770e-01 -6.68518066e-01 -1.92150879e+00 -7.37030029e-01 -
1.01183732e+11 -8.33924574e+10 -7.95983063e+10 -1.70394897e-01
5.25512695e-01 -3.33097839e+00 1.56301740e+12 1.28818991e+12
1.22958044e+12 5.80200195e-01 1.55352783e+00 3.64527008e+11
3.00431724e+11 2.86762821e+11 3.97644043e-01 8.58604718e+10
7.07635073e+10 6.75439422e+10 -7.25449332e+11 1.00689540e+12
9.61084146e+11 2.18532428e+11 -4.81675252e+12 2.63818648e+12

Very large coefficients!

Regularising Linear Models (Shrinkage methods)





$$\sum_{i=1}^{n}\left(y_i-\beta_0-\sum_{j=1}^{p}\beta_j x_{ij}\right)^2 = 0$$

1. Curse of dimensionality results in large magnitude coefficients which results in a complex undulated surface / model.

1. This complex surface has the data points occupying the peaks and the valleys

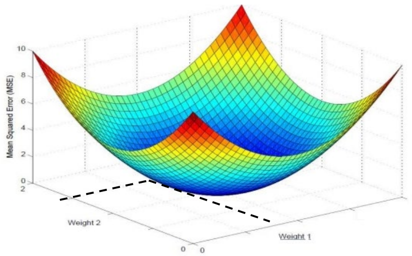
1. The model gives near 100% accuracy in training but poor result in testing and the testing scores also vary a lot from one sample to another.

1. The model is supposed to have absorbed the noise in the data distribution!

1. Large magnitudes of the coefficient give the least SSE and at times SSE = 0! A model that fits the training set 100%!

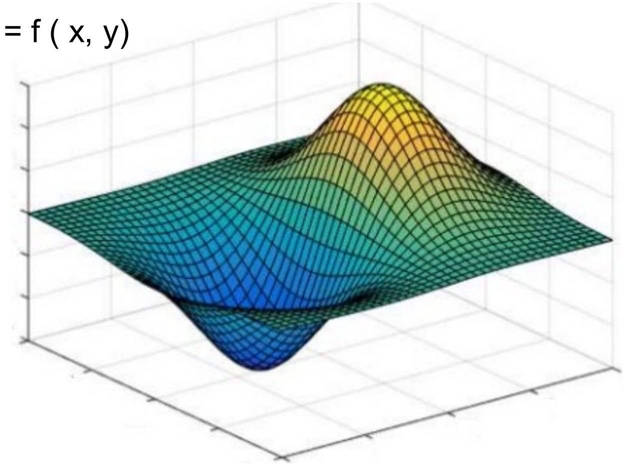
1. Such models do not generalize

Regularising Linear Models (Shrinkage methods)



$$\sum_{i=1}^{n}\left(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij}\right)^2 + \lambda\sum_{j=1}^{p}\beta_j^2$$

Z = f (x, y)



1. In Ridge Regression, the algorithm while trying to find the best combination of coefficients which minimize the SSE on the training data, is constrained by the penalty term

1. The penalty term is akin to cost of magnitude of the coefficients. Higher the magnitude, more the cost. Thus to minimize the cost, the coefficient are suppressed

1. Thus the resulting surface tends to be relatively much more smoother than the unconstrained surface. This means we have settled for a model which will make errors in the training data

1. This is fine as long as the errors can be attributed to the random fluctuations i.e. because the model does not absorb the random fluctuations in the data

1. Such model will perform equally well on unseen data i.e. test data. The model will generalize better than the complex model

Regularising Linear Models (Shrinkage methods)

Impact of Ridge Regression on the coefficients of the 56 attributes

```
Ridge model: [[ 0. 3.73512981 -2.93500874 -2.13974194 -3.56547812 -1.28898893 3.01290805
2.04739082 0.0786974 0.21972225 -0.3302341 -1.46231096 -1.17221896 0.00856067 2.48054694
-1.67596093 0.99537516 -2.29024279 4.7699338 -2.08598898 0.34009408 0.35024058 -0.41761834
3.06970569 -2.21649433 1.86339518 -2.62934278 0.38596397 0.12088534 -0.53440382 -1.88265835
-0.7675926 -0.90146842 0.52416091 0.59678246 -0.26349448 0.5827378 -3.02842915 -0.36548074
0.5956112 -0.15941014 0.49168856 1.45652375 -0.43819158 -0.20964198 0.77665496 0.36489921
 -0.4750838 0.3551047 0.23188557 -1.42941282 2.06831543 -0.34986402 -0.32320394 0.39054656 0.06283411]]
```

Large coefficients have been suppressed, almost close to 0 in many cases.

Ref: Ridge_Lasso_Regression.ipynb

greatlearning
Learning for Life

Regularising Linear Models (Shrinkage methods)

1. Lasso Regression is similar to the Ridge regression with a difference in the penalty term. Unlike Ridge, the penalty term here is raised to power 1. Also known as L1 norm.

$$\sum_{i=1}^{M}\left(y_i - \sum_{j=0}^{p} w_j \times x_{ij}\right)^2 + \lambda \sum_{j=0}^{p} |w_j|$$

1. The term λ continues to be the input parameter which will decide how high penalties would be for the coefficients. Larger the value more diminished the coefficients will be.

1. Unlike Ridge regression, where the coefficients are driven towards zero but may not become zero, Lasso Regression penalty process will make many of the coefficients 0. In other words, literally drop the dimensions

Regularising Linear Models (Shrinkage methods)

Impact of Lasso Regression on the coefficients of the 56 attributes

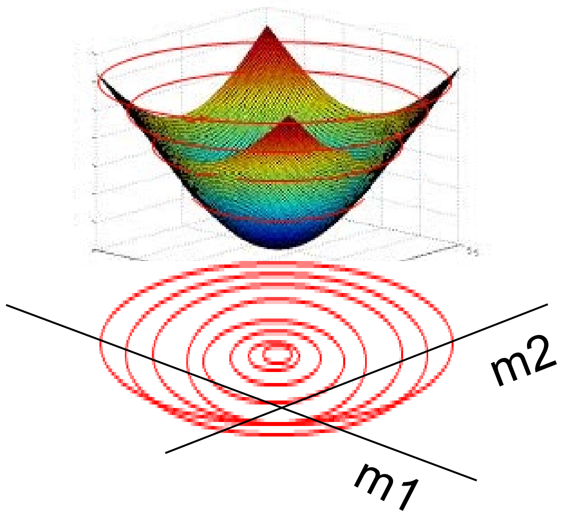
Lasso model: [0. 0.52263805 -0.5402102 -1.99423315 -4.55360385 -0.85285179 2.99044036 0.00711821 -0. 0.76073274 -0. -0. -0.19736449 0. 2.04221833 -1.00014513 0. -0. 4.28412669 -0. 0. 0.31442062 -0. 2.13894094 -1.06760107 0. -0. 0. 0. -0.44991392 -1.55885506 -0. -0.68837902 0. 0.17455864 -0.34653644 0.3313704 -2.84931966 0. -0.34340563 0.00815105 0.47019445 1.25759712 -0.69634581 0. 0.55528147 0.2948979 -0.67289549 0.06490671 0. -1.19639935 1.06711702 0. -0.88034391 0. -0.]

Large coefficients have been suppressed, to 0 in many cases, making those dimensions useless i.e. dropped from the model.

Ref: Ridge_Lasso_Regression.ipynb

Regularising Linear Models (Comparing The Methods)

To compare the Ridge and Lasso, let us first transform our error function (which is a quadratic / convex function) into a contour graph



1. Every ring on the error function represents a combination of coefficients (m1 and m2 in the image) which result in same quantum of error i.e. SSE

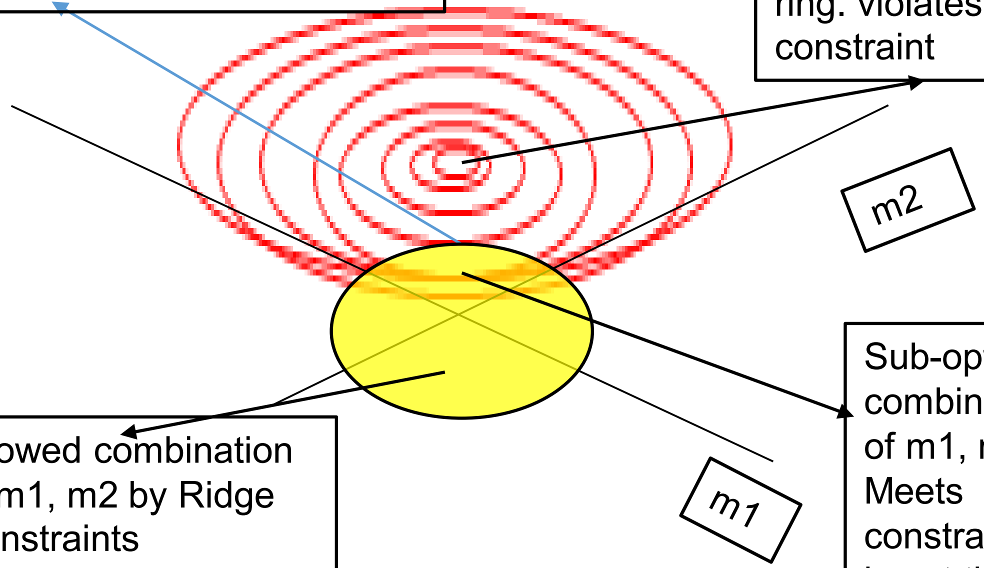
1. Let us convert that to a 2d contour plot. In the contour plot, every ring represents one quantum of error.

1. The innermost ring / bull's eye is the combination of the coefficients that gives the lease SSE

Regularising Linear Models (Ridge Constraint)

Most optimal combination of m1, m2 given the constraints

Lowest SSE error ring. violates the constraint



Allowed combination of m1, m2 by Ridge Constraints

Sub-optimal combination of m1, m2. Meets constraint but is not the minimal possible SSE within constraint

1. Yellow circle is the Ridge constraint region representing the ridge penalty (sum of squared coeff)

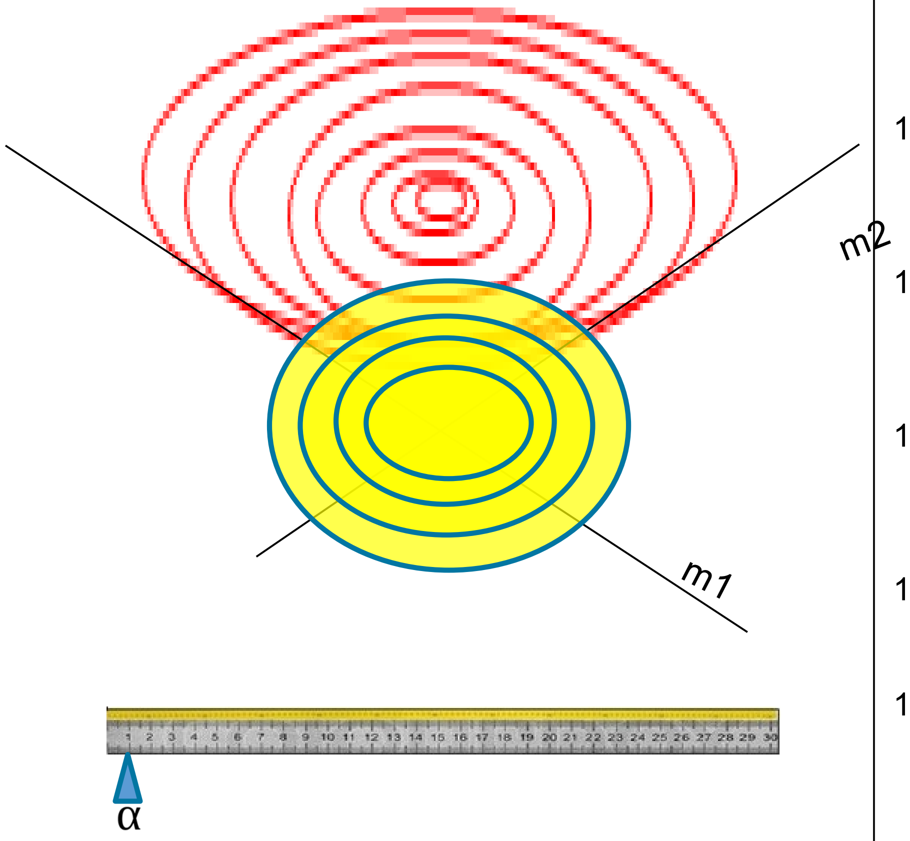
1. Any combination of m1 nd m2 that fall within yellow is a possible solution

1. The most optimal of all solutions is the one which satisfies the constraint and also minimizes the SSE (smallest possible red circle)

1. Thus the optimal solution of m1 and m2 is the one where the yellow circle touches a red circle.

<u>The point to note is that the red rings and yellow circle will never be tangential (touch) on the axes representing the coefficient. Hence Ridge can make coefficients close to zero but never zero.</u> You may notice some coefficients becoming zero but that will be due to roundoff…

Regularising Linear Models (Ridge Constraint)



1. As the lambda value (shown here as alpha) increases, the coefficients have to become smaller and smaller to minimize the penalty term in the cost function i.e. the

1. The larger the lambda, smaller the sum of squared coefficients should b $\lambda \sum_{j=1}^{p} \beta_j^2$ as a result the tighter the constraint region

1. The tighter the constraint region, the larger will be the red circle in the contour diagram that will be tangent to the boundary of the yellow region

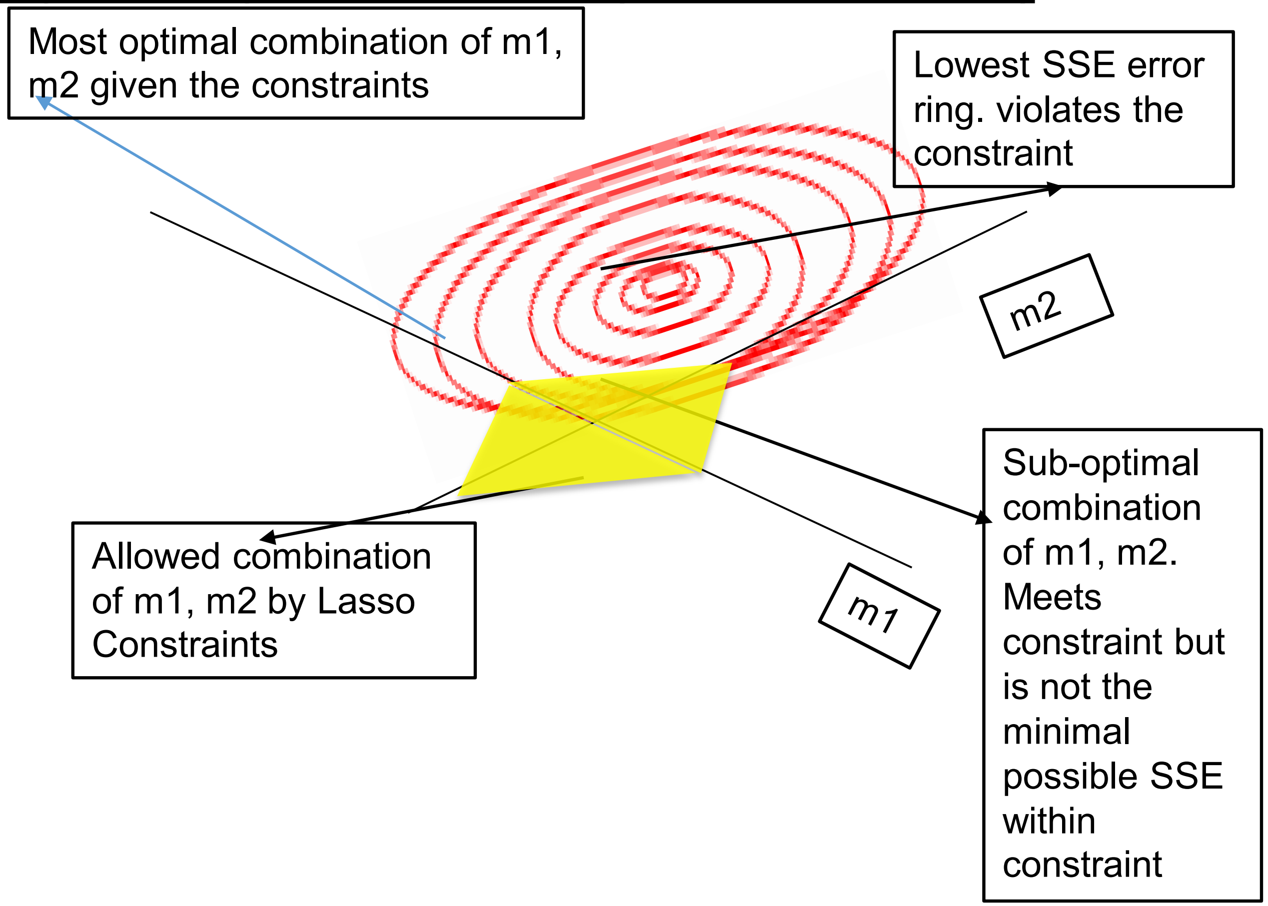
1. Thus, higher the lambda, stronger the shrinkage, the coefficient shrink significantly and hence more smooth the surface / model

1. More smoother the surface, more likely the model is going to perform equally well in production

1. When we move away from a model with sharp peaks and valleys (complex model) to smoother surface (simpler models), we reduce the variance errors but bias errors go up.

1. Using gridsearch, we have to find the right value of lambda which results in right fit, neither too complex nor too simple a model

Regularising Linear Models (Lasso Constraint)



1. Yellow rectangle is the Lasso constraint region representing the Lasso penalty (sum coeff)
1. Any combination of m1 nd m2 that fall within yellow is a possible solution
1. The most optimal of all solutions is the one which satisfies the constraint and also minimizes the SSE (smallest possible red circle)
1. Thus the optimal solution of m1 and m2 is the one where the yellow rectangle touches a red circle.

<u>The beauty of Lasso is, the red circle may touch the constraint region on the attribute axis! In the picture above the circle is touching the yellow rectangle on the m1 axis. But at that point m2 coefficient is 0! Which means, that dimension has been dropped from analysis. Thus Lasso does dimensionality reduction which Ridge does not</u>