Implementation of LASSO with Cyclic Gradient Descent

Narotam Singh : 2015CSB1065 and **Nittin Singh : 2015CSB1067**
Indian Institute of Technology Ropar
Punjab

Abstract

This report summarises our machine learning project that aims to look at the working of the LASSO operator over the linear regression without any regularization. We showcase here the implementation of LASSO using : Cyclic coordinate descent. Here we will be implementing the LASSO algorithm from scratch and then we will compare results with the linear regression. Later we will try a small modification over lasso that is relaxed lasso and compare its results as well.

Introduction

Overfitting have always terrorized people familiar with machine learning because it can ruin our models. The workaround around overfitting is the famous regularization method for both classification and regression problems, however, we shall only discuss regression here. Cost function least squares error without regularization is written as :

$$Cost = \sum_{n=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{m}\beta_i x_i)^2 \quad (1)$$

where x is an instance denoted by $\{x_1\ x_2\ ..\ x_m\}$. The cost function is penalised by adding a p-norm of weights learned multiplied by a constant called learning parameter :

$$Cost = \sum_{n=1}^{N}(y_i - \beta_0 - \sum_{j=1}^{m}\beta_i x_i)^2 + \lambda(\sum_{j=1}^{m}\beta^p)^{1/p} \quad (2)$$

In most cases, ridge regression or L-2 norm regularization (p=2) and L1-norm (p=1) regularization is used. We may want to use p¡1 but that results in a non-convex problem, so to keep life simple we consider p=1. When we use least squares error as the cost function while training then the L1-regularisation is called LASSO regularisation. Lasso regularization is special because it performs both variable selection as well as parameter shrinkage. It zeroes out the attributes thus creating a sparse instance unlike the ridge regularizer which never zeroes out the attributes. This speciality is used to fit high dimensional data to the model while also optimising the model by preventing overfitting. So lasso is important to fit hgih dimensional data and prevent overfitting. We may also use parameter subset selection for training to reduce the dimensions but it is very random and the resultant variance can be very high if we change the dataset. LASSO on the other hand does this automatically in the course of learning the model. These details and their importance shall be made clear further in the report when we discuss the results by exploring more about LASSO.

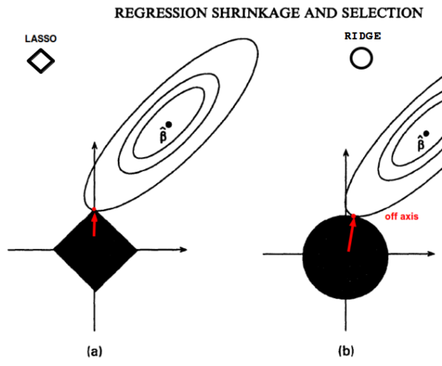

Figure 1: (a) Lasso and (b) Ridge regression

Related Works

The LASSO operator technique was first proposed by Tibshirani. In his paper [1], he proposed his technique with the support of mathematical solutions and statistical graphs, contours and compared the results of LASSO with ordinary least squares error linear regression, LASSO with cross-validation, subset selection, ridge regression, etc. In his yet another work [2], he proposed a slighly different lasso for Cox models and compared it with stepwise selection method. [3] proposed the least angle regression algorithm and showed how it can be used to solve the LASSO problem with slight modification. [4] implemented coordinate descent algorithms for lasso penalization. Chris Hans [5] discussed the various aspects of Lasso using the Bayesian treatment. Bayesian insight is also given by Tibshirani in [1]. Giving cipher weight to certain parameters is whats LASSO is doing. This type of need is encountered in datasets with attributes which are more than the number of instances. Such datasets are called large-p-small-n datasets. [6] gives the random forest approach to tackle the problems arising from such datasets.

Data-Set

The data set that we have used here is the 'Abalone' [7] data set provided by the UCI online repository. The data set consists of various attributes of a snail species 'abalone' which are then used to predict the no. of rings on their shells. These rings have a direct connection with their ages.

Methodology

The method that we have employed here to implement the LASSO is the 'Cyclic Coordinate Descent'. The objevctive of the lasso problem is

$$minimize_{(\beta \epsilon R^p)}\{\frac{1}{2N}\sum_{i=1}^{N}(y_i - \sum_{j=1}^{p} x_{ij})^2 + \lambda \sum_{j=1}^{p} abs(\beta_j)\}$$

Our assumptions involve

- $\sum_{i=1}^{N} y_i = 0$
- $\sum_{i=1}^{N} x_{ij} = 0$
- $\frac{1}{N}\sum_{i=1}^{N} x_{ij}^2 = 1$

Algorithm of Cyclic Coordinate Descent:

- choose any of the components say $\{j\}$
- now optimize the error function with respect to the $\{\beta_j\}$ the update equation for this is

$$\beta_j(new) = \begin{cases} \frac{1}{N}\langle X_j, r_j\rangle - \lambda & if \frac{1}{N}\langle X_j, r_j\rangle > \lambda \\ 0 & if \frac{1}{N}\langle X_j, r_j\rangle <= abs(\lambda) \\ \frac{1}{N}\langle X_j, r_j\rangle + \lambda & if \frac{1}{N}\langle X_j, r_j\rangle < \lambda \end{cases}$$

  - repeat this for all the dimensions in any order.
  - also called as the *soft-thresholding* of the weights $\{\beta_j\}$
  - these equations can be derived with the notion of subgradient as at zero gradient of *Absloute()* function is not defined, we can consider its derivative at zeros $\epsilon[-1, 1]$
- repeat previous step for a no. of cycles till error is stagnant.

Experiments/Procedure

- The data-set is taken and standardized.
  - Standardization involves subtraction of mean (centering of data) and division of independent attributes by the sum of their squares over all instances.
- This standardized data is taken as input and for different values of λ graph is plotted between the $calculated\_weightsv/s\frac{||\beta_{calculated}||_2}{||\beta_{ridge}||_2}$ Then we have plotted the same graph but this time we took the 'ridge-regression' with l_2 norm.

- Now in order to choose the best values of lambda we have performed 10-fold cross validation and plotted error graph along with the lambda values.

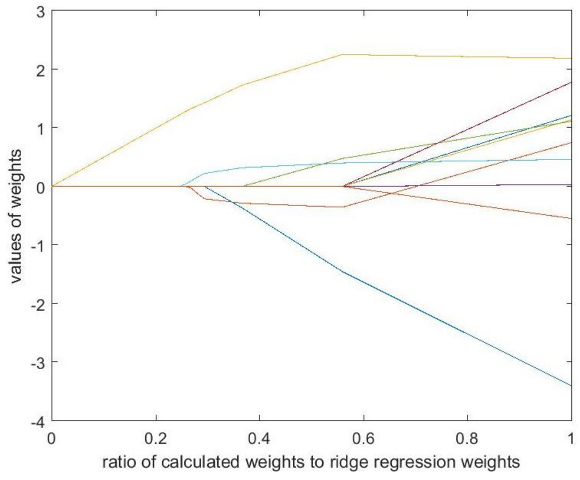

Figure 2: Graph between Coefficients v/s $\frac{||\beta_{calculated}||_2}{||\beta_{ridge}||_2}$

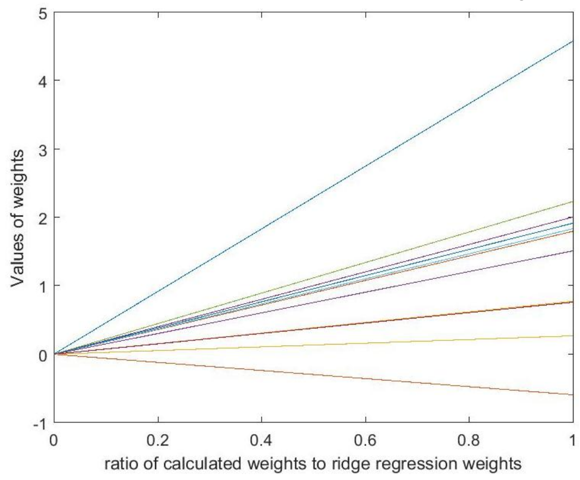

Figure 3: Graph between Coefficients v/s $\frac{||\beta_{calculated}||_2}{||\beta_{ridge}||_2}$

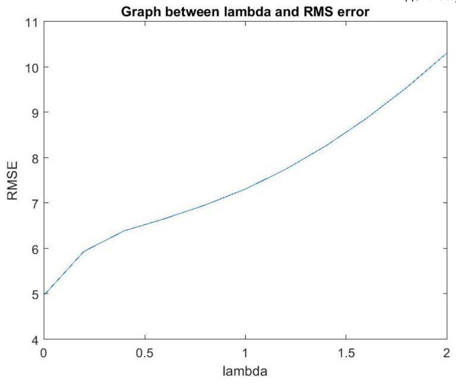

Figure 4: Graph between RMSE v/s lambda

- After choosing the optimal value for lambda we perform the lasso with 'cyclic_descent()', we will get weights to be zero and others as non zero. Now there is another variant called as the *'Relaxed Lasso'* in this approach after we have calculated the results from lasso we take the non-zero variables and perform ridge-regression on those variables. This results in the movement of weights away from the origin, which were earlier shrinked by the lasso.

Results

- In figure 1 we can see for small values of ratio we have very less variables whose values are non-zero. As we move further on the more and more variables become non zero, in fact when we reach one this implies that we have almost same coefficients as in case of ridge regression. This corresponds to the fact that penalty on the $||\beta||_1$ is very low which is same as the ridge regression with no penalty. The sparse nature of lasso solution is very useful as it reduces the no. of unnecessary variables with reasonable accuracy.
- In figure 2 we have plotted the same graph but for the case of ridge regression. The difference is very obvious as the values of coefficients are non-zero. Unlike the lasso solution.
- The figure 3 is for choosing the best value of lambda for our lasso solution. Though the value zero is having the least error but it comes with the solution which is not sparse. In order to have sparse matrix as well as reasonable accuracy we have chosen lambda = 0.3 as our required lambda.

Coefficients	Lasso	Relaxed-Lasso	Ridge-Regression
β_1	0	0	1.39
β_2	-0.029	-0.39	0.94
β_3	0	0	1.32
β_4	0	0	-0.11
β_5	0	0	1.1
β_6	0.096	0.35	0.44
β_7	0	0	2.88
β_8	0	0	-3.74
β_9	0	0	-0.65
β_{10}	1.33	1.52	1.72
RMSE error	2.5849	2.4774	2.2338

Summary

We saw that how the quadratic optimization problem can be solved using cyclic coordinate descent. We saw that how computations on high dimensional data can be avoided by nulling out some attributes and getting approximately the same error. This concept of sparse solutions helps us in extracting out the variables that truly do matter. This helps in compressing necessary details spread over multiple variables into few relevant ones. It might give a bit more error but still we are saving a lot on the computation side. Further we concluded that we are still able to optimize over simple lasso with relaxed lasso having same sparsity but less shrinked weights and thus more accuracy.

Future Work

In the future, we would like to conduct more experiments on regularization preferably on ridge regression. In this project we focused on only Cyclic Coordinate Descent , so experimenting with other procedures as well as with other cost functions along with L1-norm might be another future prospect. We could combine L1 and L2 norm to get the mixed effects of lasso and ridge. Another implementation of lasso involves the modification of LARS (least angles regression) which we can try implementing from scratch.

References

[1] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[2] Robert Tibshirani. The lasso method for variable selection in the cox model. *Statistics in medicine*, 16(4):385–395, 1997.

[3] Bradley Efron, Trevor Hastie, Iain Johnstone, Robert Tibshirani, et al. Least angle regression. *The Annals of statistics*, 32(2):407–499, 2004.

[4] Tong Tong Wu and Kenneth Lange. Coordinate descent algorithms for lasso penalized regression. *The Annals of Applied Statistics*, pages 224–244, 2008.

[5] Chris Hans. Bayesian lasso regression. *Biometrika*, 96(4):835–845, 2009.

[6] Wei-Yin Loh. Variable selection for classification and regression in large p, small n problems. In *Probability approximations and beyond*, pages 135–159. Springer, 2012.

[7] M. Lichman. UCI machine learning repository, 2013.