Regularization in Liner and Logistic Regression

Shrikant Patro

School of computer science and engineering, VIT Chennai, Tamil Nadu, India 600 127

Email: Shrikantjagannath.2018@vitstudent.ac.in

Abstract

Regression is one of the popular supervised modeling technique. It is broadly classified into liner regression and logistic regression. Linear regression is used to model the relationship between the dependent variable and multiple independent variable. The case one simple variable is called simple linear regression. Logistic regression is a part of predictive data analysis. The logistic regression is used for the describing the relationship between the one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variable. Fitting the data is very important activity performed before prediction in regression. This paper proposes different mechanism to deal with the fitting problem. This paper also include, detailed description of L1L1 LASSO, L2L2 (Ridge) norm and Elastic-Net for solving problem with linear and logistic Regression.

Introduction

The dependency explained by the continuous dependent variable and exactly one independent variable is called simple linear regression. There can be more than one independent variable. The case of logistic regression is modified version of linear with small change such that dependent variable is binary or class identifying. In Statistical modeling, regression analysis is a set of statistical process for estimating the relationship among the variables. It is also helps us to understand which among the independent variable depend on the dependent variable. Explore more on their relationship.

The ultimate goal behind identifying the relationship between finding the relationship between the variable is to draw straight line between the planes defined by the variable. It can be two dimensional or multidimensional. Better we fit the line between the data points plotted in the plane we can predict better unknown point on y- axis with respect to other axis. So, for model building and prediction we can consider one or more independent variable. If we consider more than one independent variable for the prediction with respect to dependent variable then better prediction can be made. But, in order to find better fit for the known data we tend to over fit the model. Overfitting model are poor in generalization and end up predicting badly about the unseen data during prediction. Overfitting models have this peculiar characteristics of inflating the coefficients of the variables. Regularization is a technique to penalize the loss function by adding a multiple of an L1L1 (LASSO) or an L2L2 (Ridge) norm of the estimated parameter vector of regression. The upcoming methodology section will deal with different mechanism to deal with the overfitting problem in detail. This section will including the detail description of:

(a)L1L1 (LASSO).

(b) L2L2 (Ridge).

(c) Elastic net.

Methodology

Regularization

In linear regression we can specify the number of variable being used as independent variable as deciding factor for dependent variable .The number of the variable increase the magnitude is controlled with the help of Regularization. The coefficient possessed by this variable will play major role. The variable that appear with higher coefficient will drive the total sales more than the variable with less coefficient.

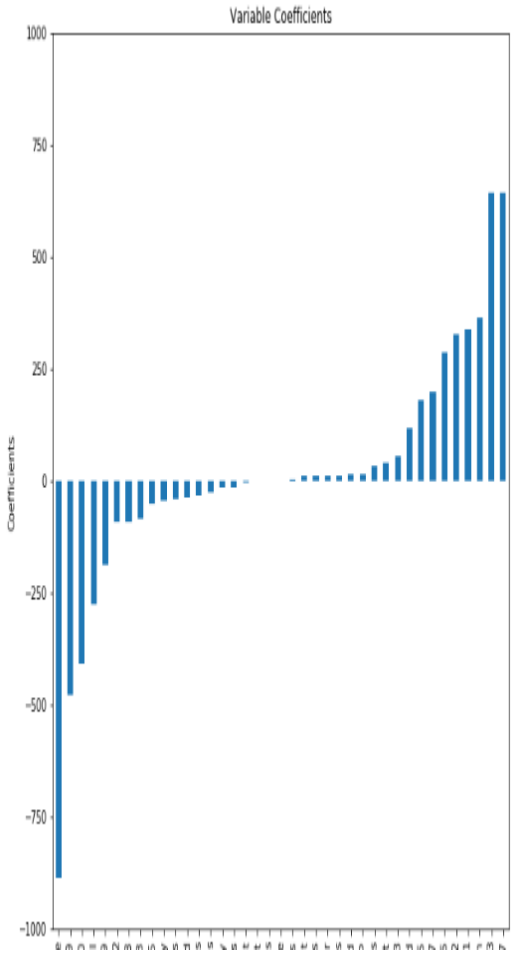

Fig. 1

In this figure, Graph represent the number of the variable possessed by example and their Coefficient. The variable at the end is having higher coefficient and contribute more in magnitude.

(a) L2L2 (Ridge)

```
from sklearn.linear_model import Ridge

## training the model

ridgeReg = Ridge(alpha=0.05, normalize=True)

ridgeReg.fit(x_train,y_train)

pred = ridgeReg.predict(x_cv)

calculating mse

mse = np.mean((pred_cv - y_cv)**2)

mse 1348171.96 ## calculating score ridgeReg.score(x_cv,y_cv) 0.5691
```

Figure 2

The model produced by the Ridge is slightly better than the model produced generalized regression. Alpha in this is the hyper parameter. It is by default set 0.05.
If we increase the value of the alpha the value of the coefficient decrease, where the value reduces close to zero but not the absolute zero. So produce, the appropriate result we must choose the value of alpha and calculate R^2.

The ultimate goal behind the above implementation is to reduce the cost

function, such that the value predicted are closer to the desired function. The cost function for the Ridge Regression is given by

$$\min\left(\left|\left|Y - X(\theta)\right|\right|_2^2 + \lambda||\theta||_2^2\right)$$

Here if you notice, we come across an extra term, which is known as the penalty term. λ given here, is actually denoted by alpha parameter in the ridge function. So by changing the values of alpha, we are basically controlling the penalty term. Higher the values of alpha, bigger is the penalty and therefore the magnitude of coefficients are reduced.

(b) L1L1 (LASSO)

In statistics and machine learning, **lasso** (least absolute shrinkage and selection operator; also **Lasso** or **LASSO**) is a **regression** analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces. Using General regression technique for prediction will increase the magnitude along with the number of variable being used is increase. This increase in magnitude is controlled with the help of LASSO regression.

LASSO (Least Absolute Shrinkage Selector Operator), is quite similar to ridge, but let's understand the difference them by implementing it in our big mart problem.

```
from sklearn.linear_model import Lasso

lassoReg = Lasso(alpha=0.3, normalize=True)

lassoReg.fit(x_train,y_train)

pred = lassoReg.predict(x_cv)

 # calculating mse

mse = np.mean((pred_cv - y_cv)**2)

 mse

1346205.82

lassoReg.score(x_cv,y_cv)

0.5720
```

Figure3

LASSO increases both mse and R^2. It is also performing better than ridge even though, the value of alpha is 0.05. So, the same performance is achieved in ridge using alpha 5.
The improvement is observed due to one additional factor called feature selection used in the LASSO and absent in Ridge.

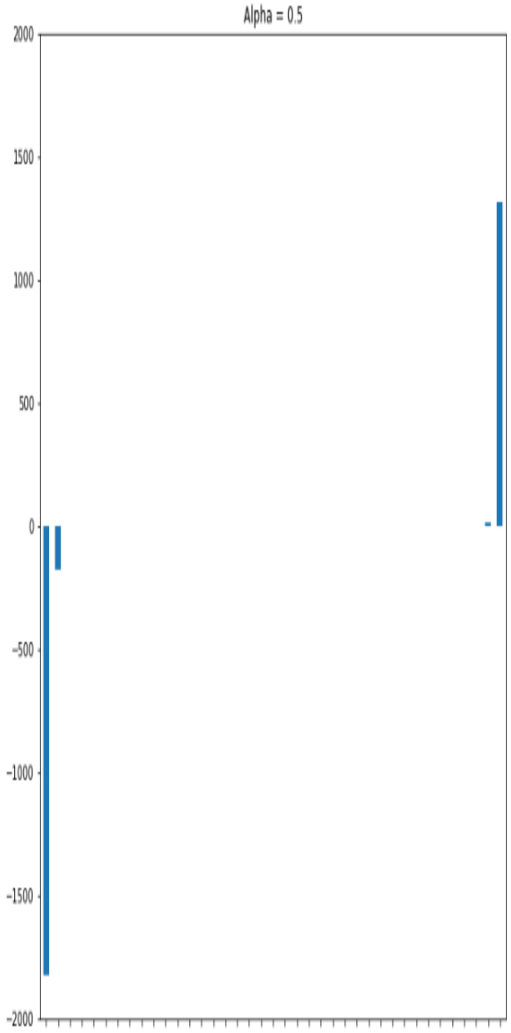

Figure 4

Mathematical formulae to represent LASSO:

$$\min\left(\left|\left|Y - X\theta\right|\right|_2^2 + \lambda||\theta||_1\right)$$

It is suggested to use LASSO when there are more number of feature because, it automatically does feature selection.

(c) Elastic Net

```
from sklearn.linear_model import ElasticNet

ENreg = ElasticNet(alpha=1, l1_ratio=0.5, normalize=False)

ENreg.fit(x_train,y_train)

pred_cv = ENreg.predict(x_cv)

#calculating mse

mse = np.mean((pred_cv - y_cv)**2)

mse 1773750.73

ENreg.score(x_cv,y_cv)

0.4504
```

Figure 5

So we get the value of R-Square, which is very less than both Ridge and Lasso. The reason behind this downfall is basically we didn't have a large set of features. Elastic regression generally works well when we have a big dataset. Note, here we had two parameters alpha and l1_ratio. First let's discuss, what happens in elastic net, and how it is different from ridge and lasso.

Elastic net is basically a combination of both L1 and L2 regularization. So if you know elastic net, you can implement both Ridge and Lasso by tuning the parameters. So it uses both L1 and L2 penalty term, therefore its equation look like as follows:

$$\min\left(\left|\left|Y - X\theta\right|\right|_2^2 + \lambda_1||\theta||_1 + \lambda_2||\theta||_2^2\right)$$

Let' say, we have a bunch of correlated independent variables in a dataset, then elastic net will simply form a group consisting of these correlated variables. Now if any one of the variable of this group is a strong predictor (meaning having a strong relationship with dependent variable), then we will include the entire group in the model building, because omitting other variables (like what we did in lasso) might result in losing some information in terms of interpretation ability, leading to a poor model performance. So, if you look at the code above, we need to define alpha and l1_ratio while defining the model. Alpha and l1_ratio are the parameters which you can set accordingly if you wish to control the L1 and L2 penalty separately. Actually, we have

Alpha = a + b and l1_ratio = a / (a+b)

Where, a and b weights assigned to L1 and L2 term respectively. So when we change the values of alpha and l1_ratio, a and b are set aaccordingly such that they control tradeoff between L1 and L2 as:

a * (L1 term) + b* (L2 term)

Let alpha (or a+b) = 1, and now consider the following cases:

- If l1_ratio =1, therefore if we look at the formula of l1_ratio, we can see that l1_ratio can only be equal to 1 if a=1, which implies b=0. Therefore, it will be a lasso penalty.
- Similarly if l1_ratio = 0, implies a=0. Then the penalty will be a ridge penalty.
- For l1_ratio between 0 and 1, the penalty is the combination of ridge and lasso.

So let us adjust alpha and l1_ratio, and try to understand from the plots of coefficient given below.

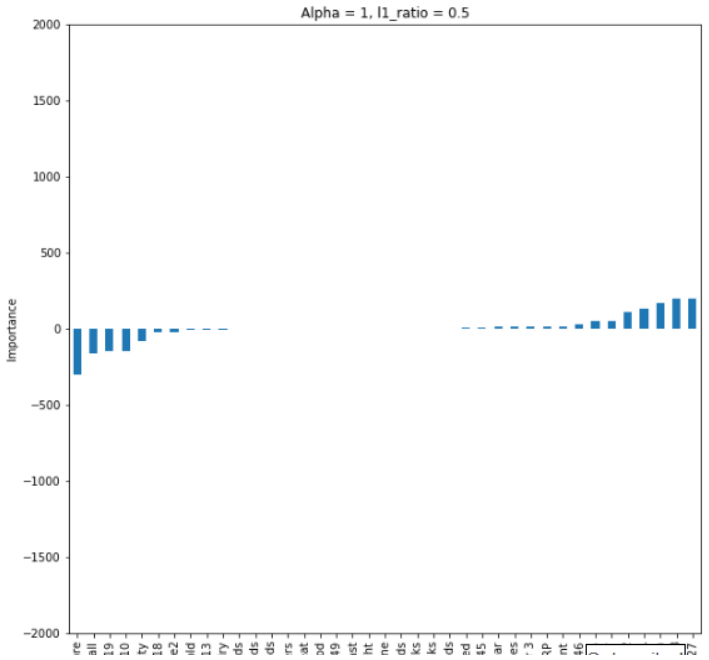

Figure 6

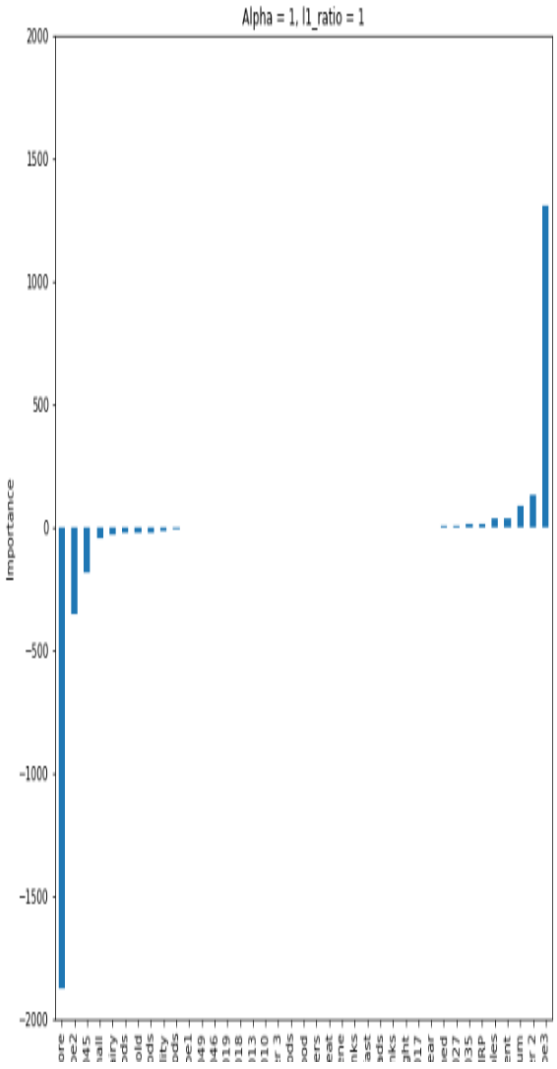

Figure7

Conclusion:

Now that you have a basic understanding of ridge and lasso regression, let's think of an example where we have a large dataset, let's say it has 10,000 features. And we know that some of the independent features are correlated with other independent features. If we apply ridge regression to it, it will retain all of the features but will shrink the coefficients. But the problem is that model will still remain complex as there are 10,000 features, thus may lead to poor model performance. Instead of ridge what if we apply lasso regression to this problem. The main problem with lasso regression is when we have correlated variables, it retains only one variable and sets other correlated variables to zero. That will possibly lead to some loss of information resulting in lower accuracy in our model .Actually we have another type of regression, known as elastic net regression, which is basically a hybrid of ridge and lasso regression. Now, you have basic understanding about ridge, lasso and elasticnet regression. But during this, we came across two terms L1 and L2, which are basically two types of regularization. To sum up basically lasso and ridge are the direct application of L1 and L2 regularization respectively.