Andhra Pradesh State Skill Development Corporation

Skill AP
APSSDC

eSkill AP
Learn Anytime Anywhere
Andhra Pradesh State Skill Development Corporation

Machine Learning

Regularized Linear Models

Skill AP
Learn Anytime Anywhere
Andhra Pradesh State Skill Development Corporation

CHAPTER 6
Regularized Linear Models

Skill AP
Learn Anytime Anywhere
Andhra Pradesh State Skill Development Corporation

Introduction to Regularized LinearModels :

As Linear Regression model is probably the simplest and the most commonly used prediction model. But there are cases that the **classical linear regression model doesn’t handle well**:

- When there is Multicollinearity. Multicollinearity is the phenomenon that one (or more) of the independent variable(s) can be expressed as the linear combination of other independent variables. In fact, this problem almost exists everywhere in the real world.
- When the number of independent variables is larger than the number of observations. When this happens, the OLS estimates are not valid mainly because there are infinite solutions to our estimators.

Thus we need to check an alternative way to solve such problems. One such widely used technique is “Regularization”. To define in a simple way Regularization is a process of introducing additional information in order to solve an ill-posed problem or to prevent overfitting .One way to regularize is to add a constraint to the loss function:

Regularized Loss = Loss Function + Constraint

Thus we have L1 and L2 are the most common types of regularization. These update the general cost function by adding another term known as the regularization term.

Thus **Ridge Regression Regularization is given by the L2 Norm of weight vector** as
The below formula should come

Cost function $J(\theta) = MSE(\theta) + \alpha*1/2*\sum(i=1 \text{ to } n)\theta^2$
here $\theta = (X\hat{}T.X+ \alpha A)\hat{}-1.X\hat{}T.y$ A is the n * n identity matrix ,θ is the value of that minimizes the cost function
y is the vector of target values containing y(1) to y(m),whereas X is the training set.

Lasso Regression Regularization is given by the L1 Norm of the weight vector as
Cost function $J(\theta) = MSE(\theta) + \alpha*\sum(i=1 \text{ to } n)|\theta(i)|$
Ridge Regression adds penalty equivalent to square of the magnitude of coefficients whereas Lasso Regression adds penalty equivalent to absolute value of the magnitude of coefficients

The goal of Supervised Learning is to learn or derive a target function which can best determine the target variable from the set of variables. So **Constraining the model** to make it simpler and **reduce the risk of Overfitting** is called Regularization

Understanding Regularization:
Regularization involves the use of a shrinkage penalty in order to reduce the residual sum of squares (RSS). This is done by selecting a value for a tuning parameter called “lambda”. Tuning parameters are used in machine learning algorithms to control the behaviour of the models that are developed.
The lambda is multiplied by the normalized coefficients of the model and added to the RSS.
Below is an equation of what was just said

$$RSS + \lambda(normalized\ coefficients)$$

The benefits of regularization are at least three-fold. First, regularization is highly computationally efficient. Instead of fitting k-1 models when k is the number of variables available (for example, 50 variables would lead 49 models!), with regularization only one model is developed for each value of lambda you specify.

Second, regularization helps to deal with the bias-variance headache of model development. When small changes are made to data, such as switching from the training to testing data, there

Skill AP Learn Anytime Anywhere Andhra Pradesh State Skill Development Corporation

can be wild changes in the estimates. Regularization can often smooth this problem out substantially.

Ridge Regression simple example:

Ridge regression involves the normalization of the squared weights or as shown in the equation below

$$RSS + \lambda(normalized\ coefficients\^{}2)$$

This is also referred to as the L2-norm. As lambda increase in value, the coefficients in the model are shrunk towards 0 but never reach 0. This is how the error is shrunk. The higher the lambda the lower the value of the coefficients as they are reduced more and more thus reducing the RSS.

The benefit is that predictive accuracy is often increased. However, interpreting and communicating your results can become difficult because no variables are removed from the model. Instead, the variables are reduced near to zero. This can be especially tough if you have dozens of variables remaining in your model to try to explain.

Data Collection:

Data collection is the most important part to build machine learning models. Even If the model is good, it won't learn anything unless the data is valid.
If there is millions of data in a model, some portion is negligible but when we will work with very few portions of data then it will lead us to the garbage.
Dataset can be biased, no matter how big the data is. For example: the game Tay. The game can communicate with people, but after some time it was not unsupervised the game started to do racial behavior. Which can lead us to unfair discrimination.

By this we can learn that collecting data is the most important thing in Machine Learning. So from where we can collect data?

There are many free resources to collect data, such as: Wikipedia, Social Medias. Here we can find any kind of data such an example is: if we want to have a list of dog breeds then we can find all the names in Wikipedia. On the basis of the name we can collect the image of all the dogs we want from many free image resources.

But sometimes it does not work like this, when we want to find the quality of a thing or something like that. For an example: if we want to show the quality of a dog image and ask which image is soothing for which one is beautiful. Then we need the same image of different quality images like that dog. For this we have to collect many more images like that dog in different quality.
In contrast, it is very difficult to find this data like this on the internet. So, to collect them we have to conduct surveys. Here we can collect data according to our choice. Conducting surveys are the easiest way to gather many data at a time.

Modelling :

Model Selection: We can think of the process of configuring and training the model as a model selection process. Each iteration we have a new model that we could choose to use or to modify. Even the choice of machine learning algorithm is part of that model selection process. Of all the possible models that exist for a problem, a given algorithm and algorithm configuration on the chosen training dataset will provide a finally selected model.

Inductive Bias: Bias is the limit imposed on the selected model. All models are biased which introduces error in the model, and by definition all models have error (they are generalizations from observations). Biases are introduced by the generalizations made in the model including the configuration of the model and the selection of the algorithm to generate the model. A machine learning method can create a model with a low or a high bias and tactics can be used to reduce the bias of a highly biased model.

Model Variance: Variance is how sensitive the model is to the data on which it was trained. A machine learning method can have a high or a low variance when creating a model on a dataset. A tactic to reduce the variance of a model is to run it multiple times on a dataset with different initial conditions and take the average accuracy as the model's performance.

Bias-Variance Tradeoff: Model selection can be thought of as the trade-off of the bias and variance. A low bias model will have a high variance and will need to be trained for a long time or many times to get a usable model. A high bias model will have a low variance and will train quickly, but suffer poor and limited performance

Evaluation:

Machine Learning model evaluation for Regression Models is to Calculate RootMean Square Error and Coefficient of Determination.

Lasso Regression Example

Lasso Regression is a popular type of regularized linear regression that includes an L1 penalty. This has the effect of shrinking the coefficients for those input variables that do not contribute much to the prediction task. This penalty allows some coefficient values to go to the value of zero, allowing input variables to be effectively removed from the model, providing a type of automatic feature selection.

In this tutorial, you will discover how to develop and evaluate Lasso Regression models in Python.

After completing this tutorial, you will know:

- Lasso Regression is an extension of linear regression that adds a regularization penalty to the loss function during training.
- How to evaluate a Lasso Regression model and use a final model to make predictions for new data.
- How to configure the Lasso Regression model for a new dataset via grid search and automatically

Understanding Lasso Regression:

Linear regression refers to a model that assumes a linear relationship between input variables and the target variable.

With a single input variable, this relationship is a line, and with higher dimensions, this relationship can be thought of as a hyperplane that connects the input variables to the target variable. The coefficients of the model are found via an optimization process that seeks to minimize the sum squared error between the predictions (yhat) and the expected target values (y).

loss = sum i=0 to n (y_i – yhat_i)^2

A problem with linear regression is that estimated coefficients of the model can become large, making the model sensitive to inputs and possibly unstable. This is particularly true for problems with few observations (samples) or more samples (n) than input predictors (p) or variables (so-called p >> n problems).

One approach to address the stability of regression models is to change the loss function to include additional costs for a model that has large coefficients. Linear regression models that use these modified loss functions during training are referred to collectively as penalized linear regression.

A popular penalty is to penalize a model based on the sum of the absolute coefficient values. This is called the L1 penalty. An L1 penalty minimizes the size of all coefficients and allows some coefficients to be minimized to the value zero, which removes the predictor from the model.

l1_penalty = sum j=0 to p abs(beta_j)
An L1 penalty minimizes the size of all coefficients and allows any coefficient to go to the value of zero, effectively removing input features from the model.

Implementation of Lasso Regression-Data Collection

Modeling and Feature Selection

Building foundation to implement Lasso Regression using Python

Sum of squares function

- Firstly, let us have a look at the Sum of square of errors function, that is defined as

$$E = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

- It is also important to note that the first requirement that should be fulfilled for any data set that we want to use for making machine learning models is that the data points should be random in nature and data size should be large.
- But this requirement is not fulfilled sometimes. That is, in some cases, number of features/dimensions(D) is greater than the number of samples/observations(N). Thus, the data set becomes fatty(D >> N) in nature instead of skinny(D << N).
- One thing to be noted is that even completely random noise can also improve R squared. But, this is very unwanted. We don't want to let noise or unwanted features alter our outputs. This can be achieved by means of regularization.
- In case of L1 regularization, few weights, corresponding to the most important features, are kept non-zero and other/ most of them are kept equal to zero.

Gaussian distribution and probabilities

- For any data set which is random in nature, it should follow Gaussian distribution.
- Any Gaussian distribution is defined by its mean, µ and variance, σ^2 and is represented by $N(\mu, \sigma^2)$, i.e., $X \sim N(\mu, \sigma^2)$ where X is the input matrix.

- For any point xi, the probability of xi is given by the expression

$$P(x_i) = \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}}$$

.

- Also, because the occurrence of each xi is independent to the occurrence of other, the joint probability of each of them is given by

$$p(x_1, x_2, \ldots, x_N) = \prod_{i=1}^{N} \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}}$$

Likelihood function

- Also, linear regression is the solution which gives the maximum likelihood to the line of best fit.
- Now, the question arises, what is likelihood? We define Likelihood as the probability of data X given a parameter of interest, in our case, it's μ. So, we define likelihood function

$$P(X|\mu) = |p(x_1, x_2, \ldots, x_N) = \prod_{i=1}^{N} \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}}$$

as .

- Linear regression maximizes this function for the sake of finding the line of best fit. We do this by finding the value of μ which maximizes this function and we can say that it is very likely that our data has come from a population that has μ as mean.
- For solving this, first we take the natural log of the likelihood function(L), then differentiate L wrt μ and then equate this to zero.

$$\ln(P(X|\mu)) = \ln(p(x_1, x_2, \ldots, x_N)) = ln \prod_{i=1}^{N} \frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}}$$

$$= \sum_{i=1}^{N} \ln\left(\frac{1}{2\pi\sigma^2} e^{-\frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}}\right) = \sum_{i=1}^{N} \ln\left(\frac{1}{2\pi\sigma^2}\right) - \sum_{i=1}^{N} \left| \frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2} \right.$$

$$\frac{\partial \ln(P(X|\mu))}{\partial\mu} = \frac{\partial \sum_{i=1}^{N} \ln\left(\frac{1}{2\pi\sigma^2}\right)}{\partial\mu} - \frac{\partial \sum_{i=1}^{N} \frac{1}{2}\frac{(x_i-\mu)^2}{\sigma^2}}{\partial\mu} = 0 + \sum_{i=1}^{N} \frac{(x_i-\mu)}{\sigma^2} = \sum_{i=1}^{N} \frac{(x_i-\mu)}{\sigma^2}$$

Skill AP Learn Anytime Anywhere Andhra Pradesh State Skill Development Corporation

$$\frac{\partial \ln(P(X|\mu))}{\partial \mu} = \sum_{i=1}^{N} \frac{(x_i - \mu)}{\sigma^2} = 0 \Rightarrow \mu = \frac{\sum_{i=1}^{N} x_i}{N}$$

Hence. this value of μ maximizes the likelihood function.

Maximize likelihood and minimizing error function

- One thing to note here is that maximizing likelihood function L is equivalent to minimizing error function E. Also, y is Gaussian distributed with mean transpose(w)*X and variance sigma-square or $y \sim N(w^T x,\ \sigma^2)$ or

  $$y \ = \ w^T x + \varepsilon, \text{ where } \varepsilon \sim N(0,\ \sigma^2)$$

  where ε is Gaussian distributed noise with zero mean and sigma-square variance.
- This is equivalent to saying that in linear regression, errors are Gaussian and the trend is linear.

Why do we need regularization?

- Now, let's understand why the need for introduction to regularization was there. The answer is outliers! In the presence of outliers, the linear regression gets the line of best fit which has some diversion from the real trend. This is because it follows the method of least squares and in order to minimize the error, it makes the trend line bent towards the outliers. This makes the prediction less accurate and far from what could be in the absence of outliers. To handle this problem, we introduce the method of Regularization.

The concept of Penalty

- L1 regularization uses L1 norm as a penalty term.

$$J_{\text{LASSO}} = \sum_{n\ =\ 1}^{N} (y_n - \hat{y}_n)^2 + \lambda ||w||_1$$

Likelihood and Prior probabilities

- Plain squared error maximizes likelihood as shown above. But now, since we have two terms in the cost function, we no longer do this. We now have two probabilities, one is likelihood probability and other one is prior. Following formula gives Likelihood:

$$P(Y|X, w) = \prod_{n=1}^{N} \frac{1}{2\pi\sigma^2} \exp(-\frac{1}{2\sigma^2}(y_n - w^T x_n)^2)$$

and formula for prior is:

Skill AP Learn Anytime Anywhere Andhra Pradesh State Skill Development Corporation

$$\text{Prior: } \mathrm{P(w)} = \frac{\lambda}{2} exp(- \lambda | w |)$$

- We call P(w) as prior because it represents our prior beliefs about w. Thus, now, J is proportional to -ln(P(Y|X, w))-ln(P(w)). Also, by Bayes rule, we get, P(w|Y,X) is proportional to P(Y|X,w)*P(w). We call P(w|Y,X) as the Posterior probability.
- The method of maximizing P(w|Y,X) is called Maximizing A Posterior or MAP.

$$\text{Also, } \mathrm{J} = (\mathrm{Y} - \mathrm{Xw})^{\mathrm{T}}(\mathrm{Y} - \mathrm{Xw}) + \lambda |\mathrm{w}|$$

$$\& \qquad \frac{\partial \mathrm{J}}{\partial w} = -2\mathrm{X}^{\mathrm{T}}\mathrm{Y} + 2\mathrm{X}^{\mathrm{T}}\mathrm{Xw} + \lambda \mathrm{sign}(w) = 0$$

$$where \ \mathrm{sign}(w) = 1 \ if \ \mathrm{x} > 0, \ -1 \ if \ \mathrm{x} < 0 \text{ and } 0 \text{ if } \mathrm{x} = 0.$$

Skill AP Learn Anytime Anywhere Andhra Pradesh State Skill Development Corporation