**Problem Statement: -**

An Analytics Company has been tasked by a crucial job of finding out what factors does affect a startup company and will it be profitable to do so or not. For this, they have collected some historical data and would like to applying supervised predictive learning algorithm such as Lasso Ridge Regression on it and provide brief insights about their data. Predict Profit, given different attributes for various startup companies.

Steps

In this guide, we will follow the following steps:

*Step 1 - Loading the required libraries and modules.*

*Step 2 - Loading the data and performing basic data checks.*

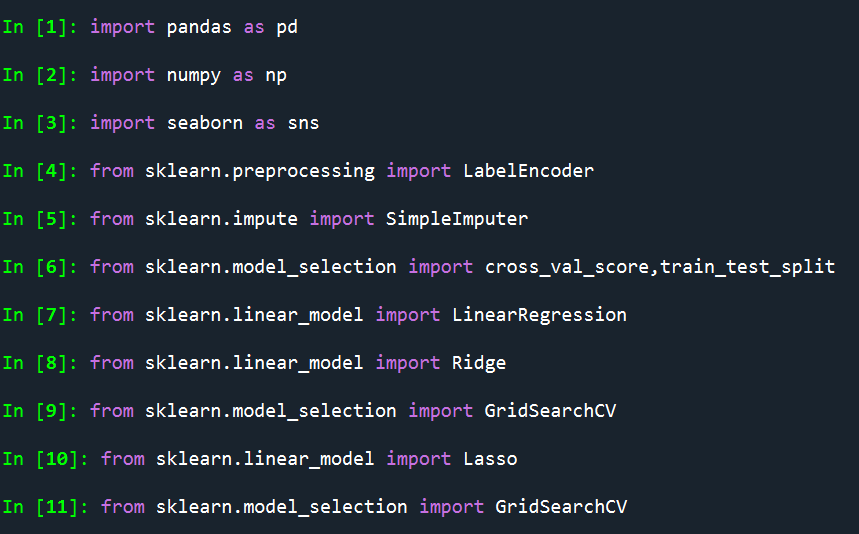
Step 3 - Creating arrays for the features and the response variable.

*Step 4 - Creating the training and test datasets.*

*Step 5 - Build, Predict and Evaluate the regression model.* We will be repeating Step 5 for the various regression models.

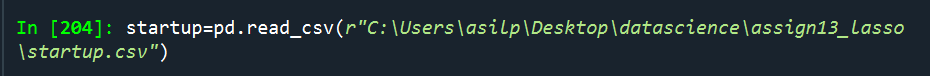
The following sections will cover these steps.

# Step1-- Loading the required libraries and modules.

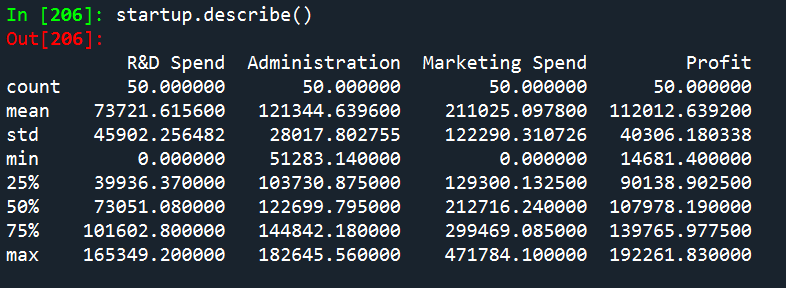
****

Step2- *Loading the data and performing basic data checks*

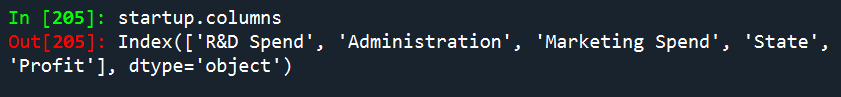
The required data is loaded as



Data is evaluated by using describe as

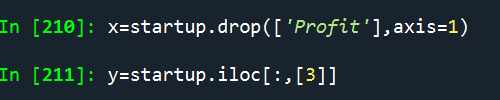


And columns are listed as



Step3- Creating arrays for the features and the response variable.

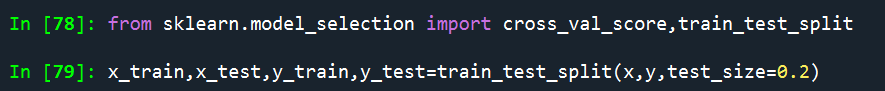
X is taken as feature variable and y is taken as response variable



Step 4 - Creating the Training and Test Datasets

We will build our model on the training set and evaluate its performance on the test set. This is called the holdout-validation method.

Split the data into training and test dataset, with the 'test\_size' argument specifying the percentage of data to be kept in the test data.

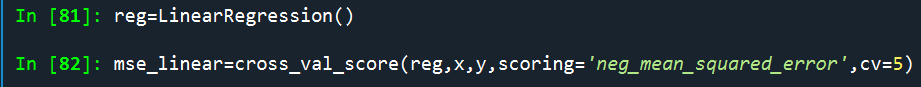


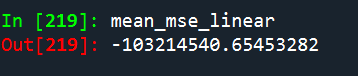
Step 5 - Build, Predict and Evaluate the Regression Model

In this step, we will be implementing the various linear regression models using the scikit-learn library.

Linear Regression

The simplest form of regression is the linear regression, which assumes that the predictors have a linear relationship with the target variable. The input variables are assumed to have a Gaussian distribution. Another assumption is that the predictors are not highly correlated with each other (a problem called multi-collinearity).





As discussed above, linear regression works by selecting coefficients for each independent variable that minimizes a loss function. However, if the coefficients are too large, it can lead to model over-fitting on the training dataset. Such a model will not generalize well on the unseen data. To overcome this shortcoming, we do regularization which penalizes large coefficients. The following sections of the guide will discuss the various regularization algorithms.

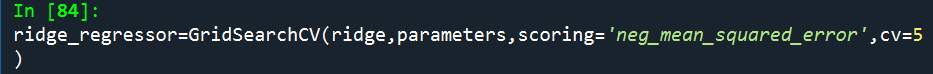
# Ridge Regression

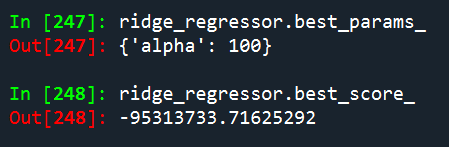
Ridge regression is an extension of linear regression where the loss function is modified to minimize the complexity of the model. This modification is done by adding a penalty parameter that is equivalent to the square of the magnitude of the coefficients.

Loss function = OLS + alpha \* summation (squared coefficient values)

In the above loss function, alpha is the parameter we need to select. A low alpha value can lead to over-fitting, whereas a high alpha value can lead to under-fitting.

GridSearchCV is used to tune the hyper parameters , the parameters obtained are





The above output gives the best parameters used and best score for ridge regression model

# Lasso Regression

Lasso regression, or the Least Absolute Shrinkage and Selection Operator, is also a modification of linear regression. In Lasso, the loss function is modified to minimize the complexity of the model by limiting the sum of the absolute values of the model coefficients (also called the l1-norm).

The loss function for Lasso Regression can be expressed as below:

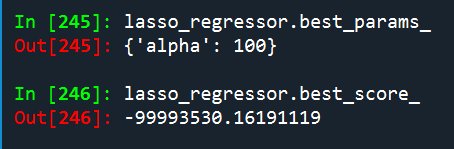
Loss function = OLS + alpha \* summation (absolute values of the magnitude of the coefficients)

In the above loss function, alpha is the penalty parameter we need to select. Using an l1 norm constraint forces some weight values to zero to allow other coefficients to take non-zero values.

In scikit-learn, a lasso regression model is constructed by using the Lasso class

GridSearchCV is used to tune the hyper parameters , the parameters obtained are



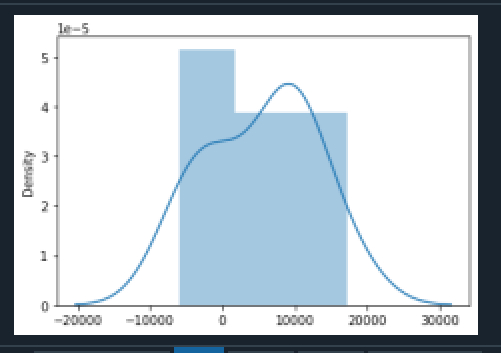


The above output gives the best parameters used and best score for lasso regression model

The Ridge regression model representations on distantplot







The Lasso regression model representations on distantplot





