Linear regression model in R, R+DBMS, Spark and

Naïve Bayes in R

By:-

Navleen Kaur

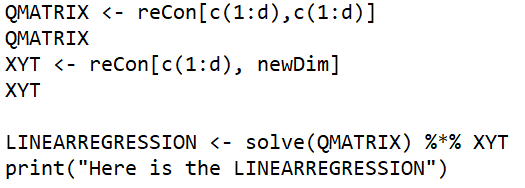
PS Id- 1641012

LR for R+DBMS

For computing linear regression(Beta), I have first computed the value of gamma and then solved beta by exploiting the gamma.

For example, Here we are provided with a very large Diabetes dataset. The steps to compute linear regression using formula is as follows:

1. I have applied a query to have gamma operator on this dataset.
2. The result for the gamma operator is a table which consists of I, j and v values.
3. This gamma operator is been modified into the matrix form by using a function in R.
4. After getting a gamma matrix, we can further transform this matrix to get the value of Q and XY transpose.
5. After getting this value, we can calculate the value for beta which the linear regression.



A screenshot of a computer

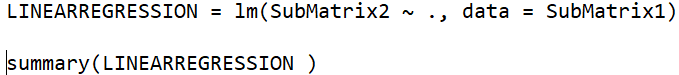
Description generated with high confidence

LR in R alone

To compute linear regression model in R, I have followed the following steps:

1. I have data in csv file format with us.
2. I have calculated the dimensions of this csv file data.
3. Then, I have extracted all the columns except the last one which contains class 0 and 1.
4. After that I have applied built in linear regression function, lm() ,to compute the linear regression model.

Now, this model has been computed for two datasets. First one is for diabetes dataset and the other one is for credit card dataset.



A screenshot of a computer

Description generated with very high confidence

A picture containing electronics

Description generated with high confidence

Its been observed above that credit card dataset is much bigger than the diabetes dataset, so to compute the linear regression model for diabetes dataset takes less time as compared to the credit card dataset.

Also, the difference between R+DBMS and R alone for dataset is as follows:-

1. R alone is fastest when computing on small datasets but it is slowest when working on largest datasets.
2. Its computation is very slow when we have lots of variables stored , often requiring a lot of RAM.
3. R+DBMS is better than R for large datasets.

LR in Spark

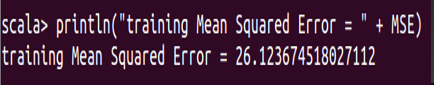
For computing a linear regression model in spark, I have followed the following steps:-

1.I have the dataset which will be first converted to a RDD.

val data = sc.textFile("/home/navleen/Downloads/diabetes.data")

2. Then I have called the labeled point function.

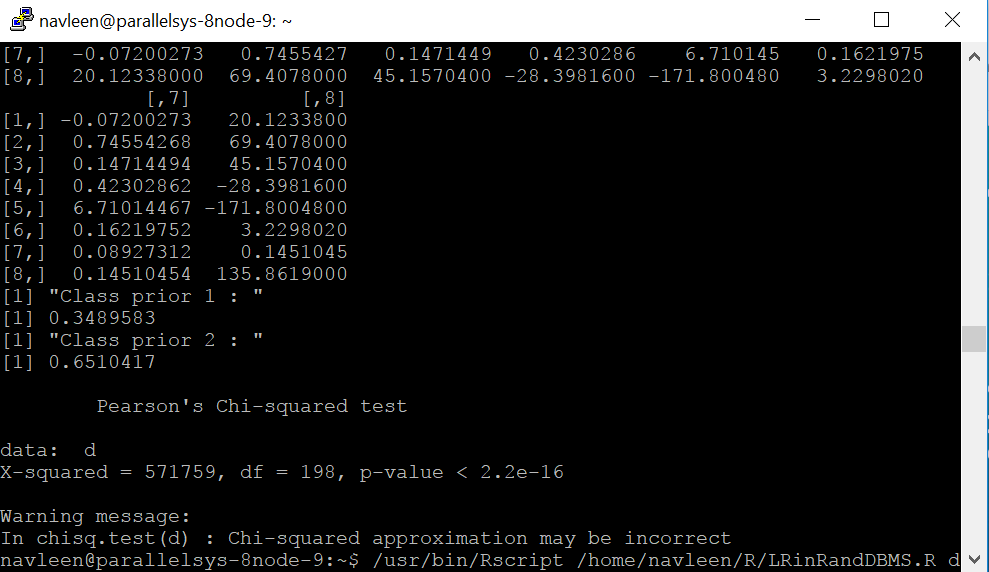
3. Further, iterations and step size has been increased and mean square error has been calculated.



A screenshot of a cell phone

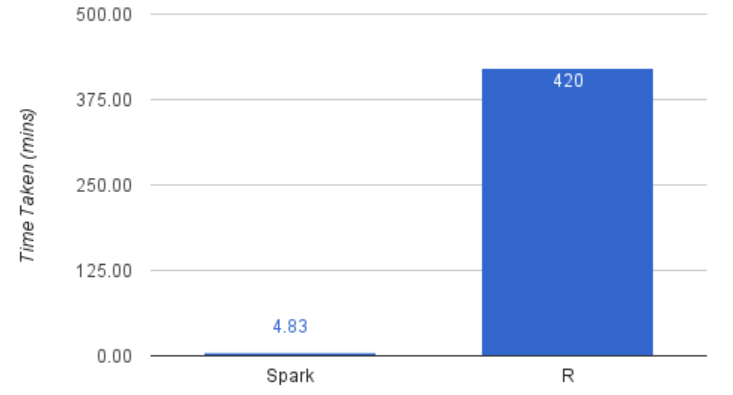
Description generated with very high confidence

Naïve Bayes in R

* Computing a model- to get the mean per class and to compute the variance per class.
* Scoring a test dataset
* It involves:
* Classify sub-tables based on class value(0,1)
* Apply gamma function on each table
* Form gamma matrix
* Calculate the value of q and l
* Calculate mean and standard deviation
* Calculate class priors
* Calculate chi-square , feature selection method

Also, we can contrast R and spark in terms of their performance, R will face computation problem and will become slow when we perform it on big datasets.

Whereas, Spark does not face any computation problem for large datasets. It will keep on working for big datasets as well.



Apart from naïve bayes classifier, we also have different types of classifiers which can be used to compute our classification problem. Different types of classifiers apart from naïve bayes are as follows:-

->Logistic regression

-> Decision trees

Decision trees are very flexible, easy to understand and easy to debug. They will work with classification problems and regression problems. One of the coolest thing about decision trees is that they only need a table of data and they will build a classifier directly from that data without needing any up front design work to take place.

Naïve Bayes requires you to build a classification by hand We cannot just toss a bunch of data at it and have it pick the best features it will use to classify whereas decision trees will choose the best features for us from tabular data.

Naïve bayes is used a lot in robotics and computer vision and does quite well with those tasks. Decision trees perform poorly in those situations