PROGRAMMING ASSIGNMENT 3
HOMEWORK 5, FOR CREDIT
MA5755
DUE APR 2, 2024

Stochastic Gradient Descent for Logistic Regression. In this project we implement binary logistic regression. The primary goal in this exericse is to experiment with the gradient and stochastic gradient descent methods which will be used to determine the optimal parameters $\mathbf{w}$. To that end, you need a function to return gradient in (10.21) which should be of the form

```
def fullGradient(w,X,y)
```

where `w` is the vector with the parameters, and `X`, `y` are the training data. Likewise, for stochastic gradient you need

```
g = randGradient(w,X,y,n)
```

where `n` is the randomly selected component. You should also write a routine of the form

```
fdescent(X,y,rho,job,nEpoch)
```

where `job` selects either the full gradient or the stochastic gradient method. Further, `rho` is a predetermined learning rate schedule. You may want to try $\rho_t = 1/\sqrt{t}$ as a first start. In the stochastic gradient method choose for `n` a random permutation of the indices 1 to N where N is the length of the training set. Finally `nEpoch` indicates how many times a new random permutation is determined in the course of the descent method. Thus the total number of iterations is `nEpoch*N`. For full gradient, no permutations are needed, but the total number of iterations should also be `nEpoch*N`.

The function `fdescent` must return the final `w`-vector and the history of the functional, that is, a vector `f` such that `f[t]` contains the value of the objective function in the `t`-th iteration, given by (10.13).

Training and Test set. We test the above methodology on the Smarket data, a popular data set in machine learning. The attached data is from the Kaggle website. It contains percentage returns for the S&P 500 stock index for every day between 2001 and 2005. For each date, it records the percentage returns for each of the five previous trading days, (Lag1 through Lag5). It also records Volume (=the number of shares traded on the previous day, in billions), Today (the percentage return on the date in question) and Direction (whether the market was Up or Down on this date). The goal here is to predict whether the marked will go up or down on a given day.

Thus fit a logistic regression model in order to predict Direction using Lag1 through Lag5 and Volume. Don't forget to include a bias term (reflected by a columns of ones in the matix X. To learn the optimal parameters use the gradient descent methods that you implemented in the first part.

Testing the Method. To get a better understanding of the data, compute the correlation matrix, leaving out the first column (i.e., the year). Which variables show a significant correlation?

Use the rows corresponding to the years 2001 to 2004 as the training set and the year 2005 as test set. Run your code for both full and stochastic descent and print out the histories `f` of the objective function in the same plot. The values of nEpoch=2 or 3 should be sufficient to get good answers. Which methods gives better results and which method uses fewer floating point operations?

Write another subroutine that creates a table that contains for each test point the probabilies and classifiers of both optimization methods, and the actual direction from the test set. How many days were misclassified?

Is this a get rich quick scheme? If you have done everything right you will have few misclassified points. This seems to suggest that we now have a way to predict the stockmarket and make a lot of money by investing in the S&P on days our classifier says it goes up. The problem is that the volume and the return variables are only available *after the fact*, meaning the volume is only known at the end of the day for which we want to predict the direction.

To make our model work as a predictor of the direction of the following day you simply have to move the last column one unit up. Then the n-th row contains the direction of the stock market for the $n+1$-st day. This way the first entry in this column is lost and the last row with no entry in the last column must be deleted, but this is inconsequential. Run the classification again with this modified data set. How many days are you misclassifying with the modified data set?

Submission Instructions. A complete solution consists of an executable code that displays the correlation matrix, the figure with the iteration histories, the table and the number of misclassified items with both methods and the original and modified data set. Further you should write the answers to the questions as comments in the code. Please follow these instructions exactly to facilitate the grading.

(1) Prepare your solution in google colab.
(2) When you are done click the *Share* button.
(3) Under *General Access* select *Anyone with link* and click *Copy Link*.
(4) Go to your email. In the area where you compose the message right click *Paste* and then the link will appear.
(5) In the subject line write *Homework 3* and your roll number.
(6) Send this email to ma5755dav@gmail.com