HW3: Regularized Multi-Class Logistic Regression

Irina Gaynanova

Introduction

In this Homework, you will be asked to implement Newton's method for multi-class logistic regression as described in class. Specifically, we will aim to minimize the following objective function

$$f(\beta) = \left[-\sum_{i=1}^{n} \left\{ \sum_{k=0}^{K-1} 1(y_i = k) \log p_{i,k} \right\} + \frac{\lambda}{2} \sum_{k=0}^{K-1} \sum_{j=1}^{p} \beta_{k,j}^2 \right], \quad \text{where} \quad p_{i,k} = \frac{e^{x_i^\top \beta_k}}{\sum_{l=0}^{K-1} e^{x_i^\top \beta_l}}.$$

with some $\lambda > 0$ over $\beta = (\beta_0, \dots, \beta_{K-1}) \in \mathbb{R}^{p \times K}$ based on supplied training data $X \in \mathbb{R}^{n \times p}$ (with 1st column being all ones) and $Y \in \{0, \dots, K-1\}^n$.

For this assignment, we will always use original data without centering or scaling (for convenience of grading).

The formula for Newton's update with learning rate $\eta > 0$ has been given in the lecture notes as

$$\beta_k^{(t+1)} = \beta_k^{(t)} - \eta (X^\top W_k X + \lambda I)^{-1} [X^\top \{P_k(X; \beta) - 1(Y = k)\} + \lambda \beta_k^{(t)}],$$

where W_k is diagonal with elements $w_{ki} = p_{i,k}(1 - p_{i,k})$, and

$$p_{i,k} = \frac{e^{x_i^\top \beta_k}}{\sum_{l=0}^{K-1} e^{x_i^\top \beta_l}}.$$

Functions Instructions

You will be asked to implement the following function

```
LRMultiClass(X, Y, Xt, Yt)
```

The **FunctionsLR.R** contains wrapper with more instructions for implementation. Here X, Y are for training data, and Xt, Yt are for testing data. You are welcome to create as many additional functions as you want, but I will explicitly test LRMultiClass function.

Within the function, in addition to performing β updates, you will be asked to calculate the objective value at each iteration as well as the classification error (in %) on both training and test data. For the classification error calculation, use the logistic regression class assignment (discussed in class), and then evaluate percentage error based on true supplied labels. More instructions are provided within the comments.

Additional considerations

As with the k-means algorithm, you are not allowed to use any external libraries and you want to use vectorized code whenever possible. In this assignment, you should pay particular attention to

- calculating $(X^\top W_k X + \lambda I)^{-1}$ as the matrix W_k may be too large to explicitly store in memory
- calculating $p_{i,k}$ for all i and k
- calculating class assignment for each i

Data instructions

You are given two datasets to test your code:

- letter-train.txt, letter-test.txt - you will use these data of 16 features to assign one of the 26 labels (corresponding to different letters in the alphabet, comes from images)
- the file Test_letter.R contains some code to load the data and extract covariates/assignments. You need to complete it with addition of intercept columns, run your algorithm and check the performance based on provided graphical plots. I encourage you to perform additional tests on your function to make sure it behaves correctly with wrong input/different values of default parameters/etc. You will also be asked to time the function on your computer (I will retime on mine anyway but it should give you some idea). More comments are provided within the code.

Grading criteria

Your assingment will be graded based on

- correctness (50%)

Take advantage of objective function values over iterations and error rates as a way to indirectly check the correctness of your function. For the letter data, I get around 20% error rate on training data, and around 25% error rate on testing data after 50 iterations.

- speed (30%)

My code takes around 4.5 seconds on my laptop for the letter data with 50 iterations, $\lambda = 1$ and $\eta = 0.1$. You will get full credit if your code runs at most 5 times slower than mine. You will get +5 points if your **completely correct** code is significantly faster than my code ((your median time)/(my median time) $<$ 0.9).

- code style/documentation (10%)

You need to comment different parts of the code so it's clear what they do, have good identation, readable code with names that make sense in consistent with recommended R style.

- version control/commit practices (10%)

I expect you to start early on this assignment (at least 2 days before it is due), and work gradually. You want to commit often, have logically organized commits with short description that makes sense.