Homework 3 (Oct. 30th)

Deadline: Wednesday, November 16th, at 11:59pm.

Submission: Read the submission instruction carefully! There are 4 questions in this assignment. You need to submit two files through Quercus for this assignment.

- The first file should be a PDF file titled `hw3_writeup.pdf` containing your answers to Questions 1 – 4, as well as R code and R outputs requested for Questions 3 and 4. You can produce the file however you like (e.g. LaTeX, Microsoft Word, scanner), as long as it is readable.
- The second file should be your completed R code, named as `penalized_logistic_regression.R`. You need to ensure that this file has the exact name as indicated. DO NOT set or modify the working directory within this file.

Neatness Point: You will be deducted one point if we have a hard time reading your solutions or understanding the structure of your code.

Late Submission: 10% of the total possible marks will be deducted for each day late, up to a maximum of 3 days. After that, no submissions will be accepted.

- **Problem 1 (3 pts)**
  Consider the classification problem with the label of Y belong to $\mathcal{C} := \{1, 2, \ldots, K\}$ and any realization x of $X \in \mathbb{R}^p$. Let f be any classifier that maps any $x \in \mathbb{R}^p$ to a label in $\mathcal{C}$.

  1. (**2 pts**) Prove that the best function f^* (i.e. the Bayes classifier)

  $$f^* := \underset{f:\mathbb{R}^p \to \mathcal{C}}{\operatorname{argmin}} \ \mathbb{E}\Big[1\{Y \neq f(X)\} \mid X = x\Big]$$

  satisfies

  $$f^*(x) = \underset{k \in \mathcal{C}}{\operatorname{argmax}} \ \mathbb{P}(Y = k \mid X = x). \tag{0.1}$$

  2. (**1 pt**) Argue that the Bayes error equals to

  $$\mathbb{E}\Big[1\{Y \neq f^*(X)\} \mid X = x\Big] = 1 - \max_{k \in \mathcal{C}} \ \mathbb{P}(Y = k \mid X = x).$$

- **Problem 2 (3 pts)**

  Consider a classification problem. Assume that the response variable Y can only take value in $\mathcal{C} = \{1, 2, 3\}$. For a fixed x_0, assume that the conditional probability of Y given $X = x_0$ follows

  $$\mathbb{P}(Y = 1 \mid X = x_0) = 0.6; \quad \mathbb{P}(Y = 2 \mid X = x_0) = 0.3; \quad \mathbb{P}(Y = 3 \mid X = x_0) = 0.1.$$

  Consider a naive classifier $\hat{f}$, called random guessing, which randomly picks one label from $\mathcal{C} = \{1, 2, 3\}$ with equal probability.

  1. (**2 pts**) Compute the expected test error rate of $\hat{f}$ at $X = x_0$.

  2. (**1 pt**) Compute the Bayes error rate at $X = x_0$ and compare it with that of $\hat{f}$.

- **Problem 3 (21 pts)**
  In this problem, you will implement logistic regression by completing the provided code in `penalized_logistic_regression.R` & `hw3_starter.R` and experiment with the completed code.

  Throughout this homework, you will be working with a subset of hand-written digits, 2's and 3's, represented as 16×16 pixel arrays. We show the example digits in Figure 1. The pixel intensities are between 0 and 1, and were read into the vectors in a raster-scan manner. You are given one training set: `train` which contains 300 examples of each class. You can access and load this training set by using functions

```
source("hw3_starter/utils.R")
data_train <- Load_data("hw3_starter/data/train.csv")
x_train <- train$x
y_train <- train$y
```

  `y_train` contains the labels of these 300 images while `x_train` are the 256 pixel values. You are also given a validation set that you should use for tuning and a test set that you should use for reporting the final performance. Optionally, the code for visualizing the dataset is located at `utils.py`.

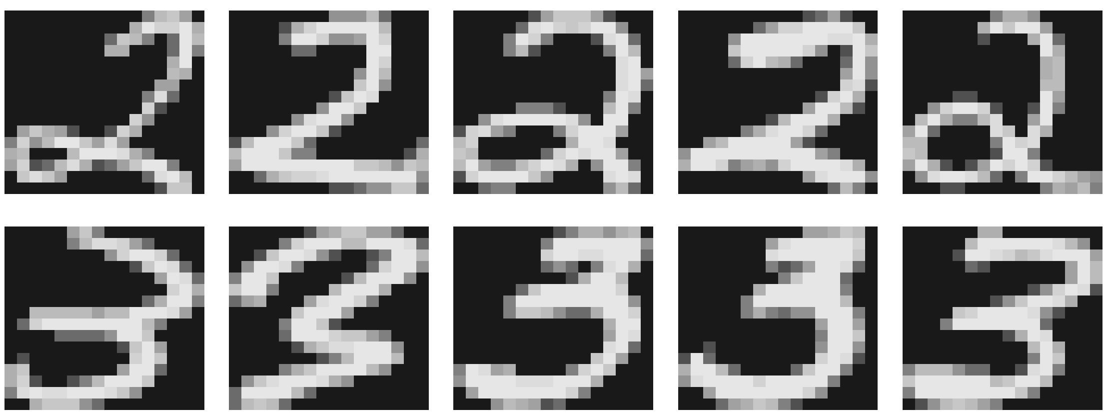

Figure 1: Example digits. Top and bottom show digits of 2s and 3s, respectively.

  You need to implement the penalized logistic regression model by minimizing the cost

$$\mathcal{J}(\boldsymbol{\beta}, \beta_0) \; := \; -\frac{1}{n}\sum_{i=1}^{n}\Big\{y_i \log\big[p(\boldsymbol{x}_i; \boldsymbol{\beta}, \beta_0)\big] + (1-y_i)\log\big[1 - p(\boldsymbol{x}_i; \boldsymbol{\beta}, \beta_0)\big]\Big\} + \frac{\lambda}{2}\|\boldsymbol{\beta}\|_2^2$$

  over $(\boldsymbol{\beta}, \beta_0) \in (\mathbb{R}^p, \mathbb{R})$, where

$$p(\boldsymbol{x}_i; \boldsymbol{\beta}, \beta_0) = \frac{e^{\beta_0 + \boldsymbol{x}_i^\top \boldsymbol{\beta}}}{1 + e^{\beta_0 + \boldsymbol{x}_i^\top \boldsymbol{\beta}}}.$$

  Here n is the total number of data points, p is the number of features in $\boldsymbol{x}_i$, $\lambda \geq 0$ is the regularization parameter and $\boldsymbol{\beta}$ and β_0 are the parameters to optimize over. Note that we should only penalize the coefficient parameters $\boldsymbol{\beta}$ and not the intercept term β_0.

1. (**2 pts**) Verify that the gradients of $\mathcal{J}(\boldsymbol{\beta}, \beta_0)$ at any $(\bar{\boldsymbol{\beta}}, \bar{\beta}_0)$ have the following expression,

$$\frac{\partial \mathcal{J}(\boldsymbol{\beta}, \beta_0)}{\partial \boldsymbol{\beta}}\Big|_{\bar{\boldsymbol{\beta}}, \bar{\beta}_0} = \frac{1}{n}\sum_{i=1}^{n}\left[-y_i + \frac{e^{\bar{\beta}_0 + \boldsymbol{x}_i^\top \bar{\boldsymbol{\beta}}}}{1 + e^{\bar{\beta}_0 + \boldsymbol{x}_i^\top \bar{\boldsymbol{\beta}}}}\right]\boldsymbol{x}_i + \lambda \bar{\boldsymbol{\beta}},$$

$$\frac{\partial \mathcal{J}(\boldsymbol{\beta}, \beta_0)}{\partial \beta_0}\Big|_{\bar{\boldsymbol{\beta}}, \bar{\beta}_0} = \frac{1}{n}\sum_{i=1}^{n}\left[-y_i + \frac{e^{\bar{\beta}_0 + \boldsymbol{x}_i^\top \bar{\boldsymbol{\beta}}}}{1 + e^{\bar{\beta}_0 + \boldsymbol{x}_i^\top \bar{\boldsymbol{\beta}}}}\right].$$

2. (**4 pts**) Implement the functions `Evaluate`, `Predict_logis`, `Comp_gradient` and `Comp_loss` located at `penalized_logistic_regression.R`. While implementing the functions, remember to vectorize the operations; you should not have any `for-loops` in these functions. Include your code in the report.

   Important note: carefully read the provided code in `penalized_logistic_regression.R`*. You should understand the code and its structure instead of using it as a black box!*

3. (**2 pts**) Complete the missing parts in function `Penalized_Logistic_Reg` located at `penalized_logistic_regression.R`. This function should train the penalized logistic regression model using gradient descent on given training set. You may use the implemented functions from step 2. Include your code in the report.

   For parts 2 and 3, your completed `penalized_logistic_regression.R` *should NOT import other* `R` *packages.*

4. (**4 pts**) Complete the part (a) in `hw3_starter.R`.

   In this part, you need to fix your regularization parameter, `lbd = 0`, and to experiment with the hyperparameters for `stepsize` (the learning rate) and `max_iter` (the number of iterations).

   [Hints: (1) You only need to use the training data for this part. (2) A too small learning rate takes longer to converge. (3) A too large learning rate is also problematic.]

   – In the write-up, report and briefly explain which hyperparameter settings you found worked the best.
   – For this choice of hyperparameters, generate and report a plot that shows how the training loss changes (iteration counter on x-axis and training loss on y-axis).
   – For this choice of hyperparameters, generate and report a plot for the training 0-1 error (iteration counter on x-axis and training error on y-axis).
   – Did the training 0-1 error have the same pattern as the training loss? Is your finding aligned with your expectation? State you reasoning.

5. (**7 pts**) Complete the part (b) in `hw3_starter.R`.

   Using the selected setting of hyperparameters (for learning rate and number of iteration) that you identified in step 4, fit the model by using $\lambda \in \{0, 0.01, 0.05, 0.1, 0.5, 1\}$.

   – (**1 pts**) Does your selected setting of hyperparameters guarantee convergence for all λ's? If not, re-identify hyperparameters for those λ's for which convergence is not guranteed. Report the hyperparameter setting(s) you used for each λ.
   – (**2 pts**) Generate and report one plot that shows how the training 0-1 error changes as you train with different values of λ.
   – (**2 pts**) Generate and report one plot that shows how the validation 0-1 error changes as you train with different values of λ.

- (**2 pts**) Comment on the effects of λ based on these two plots. Which is the best value of λ based on your experiment?

6. (**2 pts**) Complete the part (c) in `hw3_starter.R`.
   Fit the model by using the best value of λ identified in step 5 and report its test 0-1 error. Compare your test error with the model fitted by using `glmnet` with the same λ.

- **Problem 4 (10 pts)**
  In this problem, you will develop a model to predict whether a given car gets high or low gas mileage based on the `Auto` data set.

  1. (**1 pts**) Create a binary variable, **mpg01**, that contains a 1 if **mpg** contains a value above its median, and a 0 if **mpg** contains a value below its median. You can compute the median using the `median()` function.

     Split the data into a training set (70%) and a test set (30%). (Use `set.seed(0)` to ensure reproducibility.)

  2. (**2 pts**) Perform LDA on the training data in order to classify **mpg01** using the variables **cylinders**, **displacement**, **horsepower**, **weight**, **acceleration**, and **year**. What is the test error of the model obtained?

  3. (**2 pts**) Perform QDA on the training data in order to classify **mpg01** using the same variables in part 3. What is the test error of the model obtained?

  4. (**2 pts**) Perform logistic regression on the training data in order to classify **mpg01** using the same variables in part 3. What is the test error of the model obtained?

  5. (**3 pts**) Draw the ROC curves of LDA, QDA and logistic regression on the test data. Compute their AUCs and comment on which classifier you would choose. (You may find the `R` package `pROC` useful.)