EECS 553 HW 2

Due Thursday, September 12, by 11:59 PM Eastern Time via Gradescope

When you upload your solutions to Gradescope, please indicate which pages of your solution are associated with each problem.

1. Full Rank Matrices (3 points)

For any $p \times q$ matrix $\boldsymbol{B}$, with $q \leq p$, argue that $\boldsymbol{B}^T\boldsymbol{B}$ is invertible iff $\boldsymbol{B}$ has full rank, i.e., the columns of $\boldsymbol{B}$ are linearly independent.

2. Weighted Least Squares (3 points)

Consider linear regression and let $c_1, \ldots, c_n > 0$ be known weights. Determine the solution of

$$\min_{\boldsymbol{w},b} \sum_{i=1}^{n} c_i(y_i - \boldsymbol{w}^T\boldsymbol{x}_i - b)^2.$$

Express your solution in terms of the matrix $\boldsymbol{C} = \text{diag}(c_1, \ldots, c_n)$, an appropriate data matrix $\boldsymbol{X}$, and other notation as needed. To make everyone's solution consistent, please use the approach that estimates $\boldsymbol{w}$ and b together, as opposed to the approach that first eliminates b (see lecture notes on linear regression).

Hint: As a sanity check, your solution should reduce to the ordinary least squares solution when C is the identity matrix.

3. LDA and Logistic Regression (3 points)

Show that in binary classification, the LDA assumption implies the logistic regression assumption. That is, if the class-conditional densities are multivariate Gaussian with common covariance, then $\eta(\boldsymbol{x})$ has the form of $\eta(\boldsymbol{x}) = \frac{1}{1+e^{-(\boldsymbol{w}^T\boldsymbol{x}+b)}}$ for certain $\boldsymbol{w}$ and b. Give formulas for $\boldsymbol{w}$ and b in terms of the LDA model parameters.

4. Logistic regression objective function (3 points each)

Consider the regularized logistic regression objective (penalized negative log-likelihood)

$$J(\boldsymbol{\theta}) = -\ell(\boldsymbol{\theta}) + \lambda\|\boldsymbol{w}\|^2$$

where

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n}\left[y_i \log\left(\frac{1}{1+e^{-\boldsymbol{\theta}^T\tilde{\boldsymbol{x}}_i}}\right) + (1-y_i)\log\left(\frac{e^{-\boldsymbol{\theta}^T\tilde{\boldsymbol{x}}_i}}{1+e^{-\boldsymbol{\theta}^T\tilde{\boldsymbol{x}}_i}}\right)\right].$$

Here $\boldsymbol{\theta} = [b\ w_1\ \cdots\ w_d]^T$, $\tilde{\boldsymbol{x}}_i = [1\ x_{i1}\ \cdots\ x_{id}]^T$, $\boldsymbol{w} = [w_1\ \cdots\ w_d]^T$, and $y_i \in \{0, 1\}$.

(a) Show that if we change the label convention in logistic regression from $y \in \{0, 1\}$ to $y \in \{-1, 1\}$, then

$$-\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log\left(1 + \exp\left(-y_i\boldsymbol{\theta}^T\tilde{\boldsymbol{x}}_i\right)\right).$$

Introduce the notation $\phi(t) = \log(1 + \exp(-t))$ so that, by the previous problem, the logistic regression regularized negative log-likelihood may be written

$$J(\boldsymbol{\theta}) = \sum_{i=1}^{n} \phi(y_i \boldsymbol{\theta}^T \tilde{\boldsymbol{x}}_i) + \lambda \|\boldsymbol{w}\|^2. \tag{1}$$

(b) Calculate the gradient of $J(\boldsymbol{\theta})$ using (1). Your answer may be in the form of a summation. Simplify your result so that it involves only the full vectors $\boldsymbol{x}_i$ (or $\tilde{\boldsymbol{x}}_i$) and not their components.

(c) Calculate the Hessian of $J(\boldsymbol{\theta})$. This will be a $(d+1) \times (d+1)$ matrix. Again your result may be in the form of a summation. Simply your result so that it involves only the full vectors $\boldsymbol{x}_i$ (or $\tilde{\boldsymbol{x}}_i$) and not their components.

(d) Argue that J is a convex function of $\boldsymbol{\theta}$ when $\lambda \geq 0$, and strictly convex when $\lambda > 0$.

5. Logistic Regression for Fashion Classification (3 points each)

Download the files `fashion_mnist_images.npy`, `fashion_mnist_labels.npy` from Canvas under Files → Homework → HW2 → `hw2p5_data.zip`. This is a subset of the "Fashion MNIST" dataset, which contains images of different articles of clothing. This subset contains examples of coats and dresses.

The data file contains variables `x` and `y`, with the former containing images and the latter labels. The images are stored as column vectors. To visualize an image, in Python run

```
import numpy as np
import matplotlib.pyplot as plt

x = np.load("fashion_mnist_images.npy")
y = np.load("fashion_mnist_labels.npy")
d, n= x.shape

i = 0 #Index of the image to be visualized
plt.imshow(np.reshape(x[:,i], (int(np.sqrt(d)),int(np.sqrt(d)))), cmap="Greys")
plt.show()
```

The above code can be found in the file `loadfmnist.py`.

Newton's method finds a critical point of an objective function $J(\boldsymbol{\theta})$ by iterating

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - (\nabla^2 J(\boldsymbol{\theta}_t))^{-1} \nabla J(\boldsymbol{\theta}_t).$$

Implement Newton's method (a.k.a. Newton-Raphson) to find a minimizer of the regularized negative log likelihood $J(\boldsymbol{\theta})$ from the previous problem. Set $\lambda = 1$. Use the first 5000 examples as training data, and the last 1000 as test data. As a termination criterion, let i be the final iteration as soon as $|J(\boldsymbol{\theta}_i) - J(\boldsymbol{\theta}_{i-1})|/J(\boldsymbol{\theta}_{i-1}) \leq \epsilon := 10^{-6}$. Let the initial iterate $\boldsymbol{\theta}_0$ be the zero vector.

(a) Report the test error, the number of iterations run, and the value of the objective function after convergence.

(b) Generate a figure displaying 20 images in a 4 x 5 array. These images should be the 20 misclassified images for which the logistic regression classifier was least confident about its prediction (you will have to define a notion of confidence in a reasonable way – explain what this is). In the title of each subplot, indicate the true label of the image. What you should expect to see is some dresses that look kind of like coats and coats that look kind of like dresses.

(c) Submit your code as a single .py file to the corresponding assignment on *Canvas* designated for this purpose. In addition, please also submit a .pdf version of your code (just print the .py file to pdf) and upload to gradescope. This will make it easier for graders who want to take a quick look at your code without running it. Your code should follow best coding practices including the use of comments, indentation, and descriptive variable names.

Some additional remarks:

- Helpful Python commands and libraries: numpy.log, numpy.exp, numpy.sum, numpy.sign, numpy.zeros, numpy.repeat, str(), matplotlib.pyplot.
- Note that the labels in the data are ± 1, whereas the notes (at times) assume that the labels are 0 and 1.
- It is possible to "vectorize" Newton's method, that is, implement it without any additional loops besides the main loop. If you would prefer to do this, please consult the book by Hastie, Tibshirani, and Friedman and look for the iterative reweighted least squares (IRLS) implementation. Do note that they may have different notation than what was presented in class. That said, if you just use an additional loop to calculate the Hessian at each iteration, and it shouldn't take too long.