STAT 6910-001 – Principles of ML – Homework #3

Due: 5:00 PM 10/11/19

1. **Logistic Regression as ERM (6 pts).** Consider training data $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$ for binary classification and assume $y_i \in \{-1, 1\}$. Show that if $L(y, t) = \log(1 + \exp(-yt))$,then

$$\frac{1}{n} \sum_{i=1}^{n} L(y_i, \boldsymbol{w}^T \boldsymbol{x}_i + b)$$

is proportional to the negative log-likelihood for logistic regression. Therefore ERM with the logistic loss is equivalent to the maximum likelihood approach to logistic regression.
Clarification: In the above expression, y is assumed to be -1 or 1. In the notes, we had $y \in \{0, 1\}$. So all you need to do is rewrite the negative log-likelihood for logistic regression using the ± 1 label convention and simplify that formula until it looks like the formula above.

2. **Convexity and Optimization (27 pts).** Let $f : \mathbb{R}^d \to \mathbb{R}$.

   (a) (8 pts) Show that if f is strictly convex, then f has at most one global minimizer. Do not assume that the function is differentiable.

   (b) (7 pts) Use the Hessian to give a simple proof that the sum of two convex functions is convex. You may assume that the two functions are twice continuously differentiable.

   (c) (8 pts) Consider the function $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T A \boldsymbol{x} + \boldsymbol{b}^T \boldsymbol{x} + c$ where A is a symmetric $d \times d$ matrix. Derive the Hessian of f. Under what conditions on A is f convex? Strictly convex?

   (d) (4 pts) Let $J(\boldsymbol{\theta})$ be a twice continuously differentiable function. Derive the update step for Newton's method from the second order approximation of $J(\boldsymbol{\theta})$ (see lecture slides for equations for both the update step and the second order approximation).

3. **ERM and Stochastic Gradient Descent (12 pts)**. Given training data $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$, define the empirical risk for either a regression or classification problem as

$$\hat{R}(f_{\boldsymbol{\theta}}) = \frac{1}{n} \sum_{i=1}^{n} L\left(y_i, f_{\boldsymbol{\theta}}(\boldsymbol{x}_i)\right).$$

Write pseudocode describing how you would implement stochastic gradient descent to minimize $\hat{R}(f_{\boldsymbol{\theta}})$ with respect to θ. Assume a fixed mini-batch size of m and assume that the step size α is fixed for each epoch.

4. **Handwritten digit classification with logistic regression (28 pts).** Download the file `mnist_49_3000.mat` from the Homework 3 assignment. This is a Matlab data file that contains a subset of the MNIST handwritten digit dataset, which is a well-known benchmark dataset for classification. This subset contains examples of the digits 4 and 9.
The data file contains variables $\boldsymbol{x}$ and y, with the former containing the image of the digit (reshaped into column vector form) and the latter containing the corresponding label ($y \in \{-1, 1\}$). To visualize an image, you will need to reshape the column vector into a square image. You should be able to find methods for loading the data file and for reshaping the vector in your preferred language through a Google search. If you're struggling to find something that works, you may ask for suggestions on Piazza.
Implement Newton's method to find a minimizer of the regularized negative log likelihood for logistic regression: $J(\boldsymbol{\theta}) = -\ell(\boldsymbol{\theta}) + \lambda \|\boldsymbol{\theta}\|^2$. Make sure you don't forget the offset. Try setting $\lambda = 10$. Use the first 2000 examples as training data and the last 1000 as test data.

(a) (8 pts) Report the test error, your termination criterion (you may choose), how you initialized $\boldsymbol{\theta}_0$, and the value of the objective function at the optimum.

(b) (14 pts) Generate a figure displaying 20 images in a 4×5 array. These images should be the 20 misclassified images for which the logistic regression classifier was most confident about its prediction. You will have to define a notion of confidence in a reasonable way and explain how you define it. In the title of each subplot, indicate the true label of the image. What you should expect to see is a bunch of 4s that look kind of like 9s and vice versa.

(c) (6 pts) Include your well-organized, clearly commented code.

Note that the labels in the data are ± 1, whereas the lecture slides at times assume that the labels are 0 and 1. Also, note that it is possible to "vectorize" Newton's method such that you implement it without any additional loops besides the main loop. If you want to do this, you can look up the implementation for iterative reweighted least squares (IRLS) in the Elements of Statistical Learning book. Note that they may have different notation than what we've used in class. However, if you just want to use an additional loop to calculate the Hessian at each iteration, it doesn't take too long.

5. **Linear Regression (8 pts).** Download the file `bodyfat_data.mat` from Canvas. This contains variables X and y for a regression problem. The input variables correspond to abdomen circumference and hip circumference in centimeters, and y corresponds to % body fat.
Use the first 150 examples for training and the remainder for estimating the mean squared error.
Using regularized least squares regression with $\lambda = 10$, report your estimated parameters, test error (mean squared error), and the predicted response at the input $\boldsymbol{x} = [100, 100]^T$.
Do not use built in methods for regression. You do not need to include your code for this problem.

6. **Convex Losses (14 pts).** We say that a loss is convex if for each fixed y, $L(y, t)$ is a convex function of t.

   (a) (7 pts) Show that the logistic loss is convex.

   (b) (7 pts) Show that if L is a general, convex loss, then

$$\hat{R}(\boldsymbol{w}, b) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, \boldsymbol{w}^T \boldsymbol{x}_i + b)$$

   is a convex function of $\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{w} \\ b \end{bmatrix}$.
   Hints: Note that the fact that $L(y, t)$ is a convex function of t does NOT guarantee that $L(y, f(\boldsymbol{\theta}))$ is a convex function of $\boldsymbol{\theta}$ for all functions f. Also, L may not be differentiable everywhere in general and so you cannot show this using differentiation. You will need to use some other approach we covered in class.

7. **Surrogate losses (5 pts)**. In class, we discussed the fact that minimizing the 0-1 loss for classification is intractable, motivating the use of surrogate losses. Typically the least squares loss function is used in regression. Can we also use this loss as a surrogate loss for classification? Discuss why or why not. Can you think of any other surrogate losses for the 0-1 loss? If you can't think of any on your own, you may need to do some searching on the Internet.