Andy Liu
Math189R SP19
Homework 2
Tuesday, June 16, 2020

Feel free to work with other students, but make sure you write up the homework and code on your own (no copying homework *or* code; no pair programming). Feel free to ask students or instructors for help debugging code or whatever else, though.

The starter files can be found under the Resource tab on course website. The graphs for problem 3 generated by the sample solution could be found in the corresponding zipfile. These graphs only serve as references to your implementation. You should generate your own graphs for submission. Please print out all the graphs generated by your own code and submit them together with the written part, and make sure you upload the code to your Github repository.

1 (**Murphy 8.3**) Gradient and Hessian of the log-likelihood for logistic regression.
(a) Let $\sigma(x) = \frac{1}{1+e^{-x}}$ be the sigmoid function. Show that

$$\sigma'(x) = \sigma(x)\left[1 - \sigma(x)\right].$$

(b) Using the previous result and the chain rule of calculus, derive an expression for the gradient of the log likelihood for logistic regression.
(c) The Hessian can be written as $\mathbf{H} = \mathbf{X}^\top \mathbf{S} \mathbf{X}$ where $\mathbf{S} = \text{diag}(\mu_1(1-\mu_1), \ldots, \mu_n(1-\mu_n))$. Derive this and show that $\mathbf{H} \succeq 0$ ($A \succeq 0$ means that A is positive semidefinite).

Hint: Use the **negative** log-likelihood of logistic regression for this problem.

(a) Note that $\sigma'(x)$, by the chain rule, is equal to $(-e^{-x})(\frac{-1}{(1+e^{-x})^2})$. However, rearranging this expression reveals that $\sigma'(x) = \frac{1}{1+e^{-x}} \cdot \frac{e^{-x}}{1+e^{-x}} = \sigma(x) \cdot (1 - \sigma(x))$, as desired.

(b) Note that the negative log-likelihood of logistic regression is given by the equation $-\sum y_i \log \sigma(\theta^T x_i) + (1 - y_i)\log(1 - \sigma(\theta^T x_i))$. Taking the gradient of this is equal summing the gradient of the inner expression for each y_i. Using part (a) as well as the chain rule, we can conclude that the gradient of the negative log likelihood is

$$-\sum \frac{x_i y_i}{\sigma(\theta^T x_i)}(\sigma(\theta^T x_i))(1-\sigma(\theta^T x_i)) + x_i(1-y_i)(\frac{1}{1-\sigma(\theta^T x_i})))(1-\sigma(\theta^T x_i))(-\sigma(\theta^T x_i))$$

$$= -\sum(y_i(1-\sigma(\theta^T x_i)) - (1-y_i)(\sigma(\theta^T x_i)))x_i$$

$$= -\sum(y_i - \sigma(\theta^T x_i))x_i$$

$$= \sum(\sigma(\theta^T x_i) - y_i)x_i$$

$$= \mathbf{x}^T(\sigma(\theta^T \mathbf{x}) - \mathbf{y})$$

.

(c) Note that, by definition, the Hessian H_θ is equal to $\nabla_\theta(\nabla_\theta \text{nll}(\theta)^T)$. Substituting in our expression derived in part (b), we find that this is equal to $\nabla_\theta(\mathbf{x}^T(\sigma(\theta^T\mathbf{x}) - y))^T$. Since $(ab)^T = b^T a^T$, this simplifies to $\nabla_\theta(\sigma(\theta^T\mathbf{x})^T\mathbf{x} - \mathbf{y}^T\mathbf{x})$, or $\nabla_\theta\sigma(\mathbf{x}^T\theta)\mathbf{x} - \mathbf{y}^T\mathbf{x}$. Since $\nabla_\theta\mathbf{y}^T\mathbf{x}$

■

2 (**Murphy 2.11**) Derive the normalization constant (Z) for a one dimensional zero-mean Gaussian

$$\mathbb{P}(x;\sigma^2) = \frac{1}{Z}\exp\left(-\frac{x^2}{2\sigma^2}\right)$$

such that $\mathbb{P}(x;\sigma^2)$ becomes a valid density.

Note that since the integral of any probability density taken over all reals must be equal to 1, Z must be equal to $\int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}}\mathrm{dx}$. Note that Z is then also equal to $\int_{-\infty}^{\infty} e^{-\frac{y^2}{2\sigma^2}}\mathrm{dy}$. Thus, $Z^2 = Z \cdot Z = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\frac{x^2+y^2}{2\sigma^2}}\mathrm{dx\ dy}$.

Note that we can convert this integral to polar coordinates, where $0 \leq r \leq \infty, 0 \leq \theta \leq 2\pi$, $x^2 + y^2 = r^2$, and $\mathrm{dx\ dy} = r\mathrm{dr\ d}\theta$. Thus, $Z^2 = \int_0^{2\pi}\int_0^{\infty} re^{-\frac{r^2}{2\sigma^2}}\mathrm{dr\ d}\theta = 2\pi\int_0^{\infty} re^{-\frac{r^2}{2\sigma^2}}\mathrm{dr} = 2\pi\sigma^2\int_0^{\infty} re^{-\frac{r^2}{2}} = 2\pi\sigma^2$.

Thus, $Z^2 = 2\pi\sigma^2 \Rightarrow Z = \sigma\sqrt{2\pi}$, as desired. ■

3 (**regression**). In this problem, we will use the online news popularity dataset to set up a model for linear regression. In the starter code, we have already parsed the data for you. However, you might need internet connection to access the data and therefore successfully run the starter code.

We split the csv file into a training and test set with the first two thirds of the data in the training set and the rest for testing. Of the testing data, we split the first half into a 'validation set' (used to optimize hyperparameters while leaving your testing data pristine) and the remaining half as your test set. We will use this data for the remainder of the problem. The goal of this data is to predict the **log** number of shares a news article will have given the other features.

(a) (**math**) Show that the maximum a posteriori problem for linear regression with a zero-mean Gaussian prior $\mathbb{P}(\mathbf{w}) = \prod_j \mathcal{N}(w_j|0, \tau^2)$ on the weights,

$$\arg\max_{\mathbf{w}} \sum_{i=1}^{N} \log \mathcal{N}(y_i|w_0 + \mathbf{w}^\top \mathbf{x}_i, \sigma^2) + \sum_{j=1}^{D} \log \mathcal{N}(w_j|0, \tau^2)$$

is equivalent to the ridge regression problem

$$\arg\min \frac{1}{N} \sum_{i=1}^{N} (y_i - (w_0 + \mathbf{w}^\top \mathbf{x}_i))^2 + \lambda ||\mathbf{w}||_2^2$$

with $\lambda = \sigma^2/\tau^2$.

(b) (**math**) Find a closed form solution $\mathbf{x}^\star$ to the ridge regression problem:

$$\text{minimize: } ||A\mathbf{x} - \mathbf{b}||_2^2 + ||\Gamma \mathbf{x}||_2^2.$$

(c) (**implementation**) Attempt to predict the log shares using ridge regression from the previous problem solution. Make sure you include a bias term and *don't regularize the bias term*. Find the optimal regularization parameter λ from the validation set. Plot both λ versus the validation RMSE (you should have tried at least 150 parameter settings randomly chosen between 0.0 and 150.0 because the dataset is small) and λ versus $||\boldsymbol{\theta}^\star||_2$ where $\boldsymbol{\theta}$ is your weight vector. What is the final RMSE on the test set with the optimal $\lambda^\star$?

(continued on the following pages)

3 (continued)

(d) (**math**) Consider regularized linear regression where we pull the basis term out of the feature vectors. That is, instead of computing $\hat{\mathbf{y}} = \boldsymbol{\theta}^\top \mathbf{x}$ with $\mathbf{x}_0 = 1$, we compute $\hat{\mathbf{y}} = \boldsymbol{\theta}^\top \mathbf{x} + b$. This corresponds to solving the optimization problem

$$\text{minimize: } ||A\mathbf{x} + b\mathbf{1} - \mathbf{y}||_2^2 + ||\Gamma \mathbf{x}||_2^2.$$

Solve for the optimal $\mathbf{x}^\star$ explicitly. Use this close form to compute the bias term for the previous problem (with the same regularization strategy). Make sure it is the same.

(e) (**implementation**) We can also compute the solution to the least squares problem using gradient descent. Consider the same bias-relocated objective

$$\text{minimize: } f = ||A\mathbf{x} + b\mathbf{1} - \mathbf{y}||_2^2 + ||\Gamma \mathbf{x}||_2^2.$$

Compute the gradients and run gradient descent. Plot the ℓ_2 norm between the optimal $(\mathbf{x}^\star, b^\star)$ vector you computed in closed form and the iterates generated by gradient descent. Hint: your plot should move down and to the left and approach zero as the number of iterations increases. If it doesn't, try decreasing the learning rate.

(a) First, we substitute in our probability distribution $\mathcal{N}(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp(-\frac{(x-\mu)^2}{2\sigma^2})$ to our arg max formula, which yields

$$\arg\max \sum_{i=1}^{N} \log \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{(y_i - w_0 - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2}) + \sum_{j=1}^{D} \log \frac{1}{\sigma\sqrt{2\pi}} \exp(-\frac{w_j^2}{2\tau^2})$$

Simplifying using exponential rules, this becomes

$$\arg\max \sum_{i=1}^{N} (-\frac{(y_i - w_0 - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2} - \log\sigma\sqrt{2\pi}) + \sum_{j=1}^{D} (-\frac{w_j^2}{2\tau^2} - \log\sigma\sqrt{2\pi}).$$

$$= \arg\min((N+D)\log(\sigma\sqrt{2\pi}) + \sum_{i=1}^{N} \frac{(y_i - w_0 - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2} + \sum_{j=1}^{D} \frac{w_j^2}{2\tau^2})$$

Ignoring the constant value of $(N+D)\log\sigma\sqrt{2\pi}$, and dividing out by $2\sigma^2$, this is equivalent to $\arg\min \sum_{i=1}^{N} (y_i - w_0 - \mathbf{w}^T\mathbf{x}_i)^2 + \frac{\sigma^2}{\tau^2} \sum_{j=1}^{D} w_j^2$. If we let $\lambda = \frac{\sigma^2}{\tau^2}$, this is equivalent to $\arg\min \sum_{i=1}^{N} (y_i - w_0 - \mathbf{w}^T\mathbf{x}_i)^2 + \lambda||\mathbf{w}||_2^2$, as desired.

(b) It suffices to take the gradient of $||A\mathbf{x} - \mathbf{b}||_2^2 + ||\Gamma\mathbf{x}||_2^2$ with respect to $\mathbf{x}$ and set it equal to 0. Note that the expression that we seek to minimize is equal to $(A\mathbf{x} - \mathbf{b})^T(A\mathbf{x} - \mathbf{b}) + (\Gamma\mathbf{x})^T\Gamma\mathbf{x} = (\mathbf{x}^TA^T - \mathbf{b}^T)(A\mathbf{x} - \mathbf{b}) + \mathbf{x}^T\Gamma^T\Gamma\mathbf{x} = \mathbf{x}^TA^TA\mathbf{x} - 2\mathbf{x}^TA^T\mathbf{b} + \mathbf{b}^T\mathbf{b} + \mathbf{x}^T\Gamma^T\Gamma\mathbf{x}$. Taking the gradient of this expression with respect to $\mathbf{x}$ yields $2A^TA\mathbf{x} - 2A^T\mathbf{b} + 2\Gamma^T\Gamma\mathbf{x}$.

This gradient must be equal to 0, from which we can conclude that $2A^TA\mathbf{x} - 2\Gamma^T\Gamma\mathbf{x} + 2A^T\mathbf{b} = 0 \Rightarrow A^TA\mathbf{x} + A^T\mathbf{b} = \Gamma^T\Gamma\mathbf{x}$. We can solve for $\mathbf{x}$ by rearranging this equation to become $\Gamma^T\Gamma\mathbf{x} + A^TA\mathbf{x} = A^T\mathbf{b}$

$$\Rightarrow \mathbf{x}(\Gamma^T\Gamma + A^TA) = A^T\mathbf{b} \Rightarrow \boxed{\mathbf{x}^\star = (A^T\mathbf{b})(\Gamma^T\Gamma + A^TA)^{-1}}.$$

(c) My code (in GitHub) computed an optimal regularization parameter of 9.1707, a validation set RMSE of 0.8341, and a test set RMSE of 0.8628. All necessary graphs are also in the GitHub folder.

(d) Similarly to part (b), we can just expand our function and set its gradient equal to 0 before solving for the optimal $x^\star$.

Note that $||A\mathbf{x} + b\mathbf{1} - \mathbf{y}||_2^2 + ||\Gamma\mathbf{x}||_2^2 = (A\mathbf{x} + b\mathbf{1} - \mathbf{y})^T(A\mathbf{x} + b\mathbf{1} - \mathbf{y}) + (\Gamma\mathbf{x})^T(\Gamma\mathbf{x}) = (\mathbf{x}^TA^T + b\mathbf{1}^T - \mathbf{y}^T)(A\mathbf{x} + b\mathbf{1} - \mathbf{y}) + \mathbf{x}^T\Gamma^T\Gamma\mathbf{x} = \mathbf{x}^TA^TA\mathbf{x} - 2\mathbf{y}^TA\mathbf{x} + 2b\mathbf{1}^TA\mathbf{x} - 2b\mathbf{1}^T\mathbf{y} + b^2n + \mathbf{y}^T\mathbf{y} + \mathbf{x}^T\Gamma^T\Gamma\mathbf{x}$.

First, we take the gradient of this function with respect to $\mathbf{x}$. By standard gradient properties, we can compute that this is equal to $2A^TA\mathbf{x} - 2\mathbf{y}^TA + 2b\mathbf{1}^TA + 2\Gamma^T\Gamma\mathbf{x}$. Setting this equal to zero (as we are searching for the minimum) yields $A^TA\mathbf{x} + b\mathbf{1}^TA + \Gamma^T\Gamma\mathbf{x} = \mathbf{y}^TA$.

Then, we can take the gradient of this function with respect to $\mathbf{b}$. This is equal to $2\mathbf{1}^TA\mathbf{x} - 2\mathbf{1}^Ty + 2bn$. Setting this equal to zero, we find that $2bn = 2\mathbf{1}^Ty - 2\mathbf{1}^TA\mathbf{x} \Rightarrow b^\star = \frac{2\mathbf{1}^Ty - 2\mathbf{1}^TA\mathbf{x}}{2n} = \boxed{\frac{\mathbf{1}^Ty - \mathbf{1}^TA\mathbf{x}}{n}}$.

Now, we can substitute this $\mathbf{b}$ value back into our first equation, $A^TA\mathbf{x} + b\mathbf{1}^TA + \Gamma^T\Gamma\mathbf{x} = \mathbf{y}^TA$, to find that $A^TA\mathbf{x} + \frac{\mathbf{1}^Ty - \mathbf{1}^TA\mathbf{x}}{n}\mathbf{1}^TA + \Gamma^T\Gamma\mathbf{x} = \mathbf{y}^TA$. Simplifying, this becomes $A^TA\mathbf{x} + \frac{1}{n}((\mathbf{1}^Ty)(\mathbf{1}^TA) - (\mathbf{1}^TA\mathbf{x})(\mathbf{1}^TA)) + \Gamma^T\Gamma\mathbf{x} = \mathbf{y}^TA \Rightarrow A^TA\mathbf{x} - \frac{1}{n}(\mathbf{1}^TA\mathbf{x})(\mathbf{1}^TA) + \Gamma^T\Gamma\mathbf{x} = \mathbf{y}^TA - \frac{1}{n}(\mathbf{1}^Ty\mathbf{1}^TA)$.

Note that all terms in the left-hand side contain a factor of $\mathbf{x}$, so it suffices to multiply both sides by the inverse of the non-$\mathbf{x}$ terms to find that

$$\mathbf{x}^\star = (A^TA - \frac{1}{n}(\mathbf{1}^TA)(\mathbf{1}^TA) + \Gamma^T\Gamma)^{-1}(\mathbf{y}^TA - \frac{1}{n}(\mathbf{1}^Ty\mathbf{1}^TA)).$$

My code computed a difference in bias of $6.9573 \cdot 10^{-10}$ and a difference in weights of $9.1962 \cdot 10^{-10}$, which is insignificant.

(e) My code (in GitHub) computed a difference in bias of $1.5385 \cdot 10^{-1}$ and a difference in weights of $7.9392 \cdot 10^{-1}$. All necessary graphs are also in the GitHub.

■