Shanni Lam
Math189R SU20
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) Differentiate directly. $\sigma^{'}(x) = \frac{d}{dx}\frac{1}{1+e^{-x}} = \frac{d}{dx}(1+e^{-x})^{-1} =$
$-1 * -e^{-x}(1+e^{-x})^{-2} = (\frac{1}{1+e^{-x}})(\frac{e^{-x}}{1+e^{-x}}) =$
$\sigma(x)(\frac{e^{-x}}{1+e^{-x}}) = \sigma(x)(\frac{1+e^{-x}-1}{1+e^{-x}}) =$
$\sigma(x)(1 - \frac{1}{1+e^{-x}}) = \sigma(x)[1-\sigma(x)]$, as desired.

b) Recall the negative log likelihood equation for logistic regression is

$$NLL(\boldsymbol{\theta}) = -\sum_i y_i log\sigma(\boldsymbol{\theta}^T x_i) + (1-y_i)log(1-\sigma(\boldsymbol{\theta}^T x_i))$$

Take the gradient with respect to $\boldsymbol{\theta}$.

$\nabla_{\boldsymbol{\theta}} NLL(\boldsymbol{\theta}) = -\sum_i y_i \frac{1}{\sigma(\boldsymbol{\theta}^T x_i)}\sigma^{'}(\boldsymbol{\theta}^T x_i) + (1-y_i)\frac{1}{1-\sigma(\boldsymbol{\theta}^T x_i)}(-\sigma^{'}(\boldsymbol{\theta}^T x_i))$
$= -\sum_i y_i \frac{1}{\sigma(\boldsymbol{\theta}^T x_i)}\sigma(\boldsymbol{\theta}^T x_i)[1-\sigma(\boldsymbol{\theta}^T x_i)] + (1-y_i)\frac{1}{1-\sigma(\boldsymbol{\theta}^T x_i)}(-\sigma(\boldsymbol{\theta}^T x_i)[1-\sigma(\boldsymbol{\theta}^T x_i)])$ since $\sigma'(x) = \sigma(x)[1-\sigma(x)]$. This will result in cancellation of terms.
$= -\sum_i y_i(1-\sigma(\boldsymbol{\theta}^T x_i))x_i - (1-y_i)\sigma(\boldsymbol{\theta}^T x_i)x_i$

$= -\sum_i y_i x_i - y_i \sigma(\boldsymbol{\theta}^T x_i) x_i - \sigma(\boldsymbol{\theta}^T x_i) x_i + y_i \sigma(\boldsymbol{\theta}^T x_i) x_i$ by the distributive property
$= \sum_i (\sigma(\boldsymbol{\theta}^T x_i) - y_i) x_i$ by cancellation of terms and factoring
$= \sum_i (\mu_i - y_i) x_i = X^T(\boldsymbol{\mu} - \mathbf{y})$

where $\mu_i = \sigma(\boldsymbol{\theta}^T x_i)$ and x_i is the ith column of matrix $\boldsymbol{X}$.

c) We will use the expression for gradient of the log likelihood for logistic regression in order to find the Hessian Matrix.

$H = \nabla_{\boldsymbol{\theta}}(\nabla_{\boldsymbol{\theta}} NLL(\boldsymbol{\theta}))^T = \nabla_{\boldsymbol{\theta}}[\boldsymbol{X}^T(\boldsymbol{\mu} - \mathbf{y})]^T = \nabla_{\boldsymbol{\theta}}((\boldsymbol{\mu}^T - \mathbf{y}^T)\boldsymbol{X}) = \nabla_{\boldsymbol{\theta}}(\boldsymbol{\mu}^T \boldsymbol{X} - \mathbf{y}^T \boldsymbol{X})$
$= \nabla_{\boldsymbol{\theta}} \boldsymbol{\mu}^T \boldsymbol{X} = \nabla_{\boldsymbol{\theta}}[\sigma(\boldsymbol{X\theta})^T] X = \boldsymbol{X}^T \sigma'(\boldsymbol{\theta}^T \boldsymbol{X})^T \boldsymbol{X} = \boldsymbol{X}^T (\boldsymbol{\mu}(\mathbf{1} - \boldsymbol{\mu}))^T \boldsymbol{X} = \boldsymbol{X}^T S \boldsymbol{X}$

where $S = diag(\mu_1(1-\mu_1), \cdots, \mu_n(1-\mu_n)) = (\boldsymbol{\mu}(\mathbf{1} - \boldsymbol{\mu}))^T$. $H_{\boldsymbol{\theta}}$ is positive semidefinite if S is positive semidefinite, which occurs if the eigenvalues of S are positive. Since S is a diagonal matrix, its eigenvalues are its diagonal entries. So, we must show that the diagonal entries or eigenvalues $\boldsymbol{\mu_i}(1 - \boldsymbol{\mu_i}) = \sigma(\boldsymbol{\theta}^T x_i)(1 - \sigma(\boldsymbol{\theta}^T x_i)) \geq 0$. Recall that $\boldsymbol{\mu_i} = \sigma(\boldsymbol{\theta}^T x_i)$.

We know that the sigmoid function $\sigma(x)$ returns values between 0 and 1 exclusive, which implies $\sigma(\boldsymbol{\theta}^T x_i))(1 - \sigma(\boldsymbol{\theta}^T x_i))) \geq 0$. Therefore, the Hessian matrix H is positive semidefinite, as desired.

■

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.

A probability density function integrates to 1, so $\int_{\mathbb{R}} \mathbb{P}(x;\sigma^2)dx = \int_{\mathbb{R}} \frac{1}{Z} exp(-\frac{x^2}{2\sigma^2})dx = \frac{1}{Z}\int_{\mathbb{R}} exp(-\frac{x^2}{2\sigma^2})dx = 1$. Then, $Z = \int_{\mathbb{R}} exp(-\frac{x^2}{2\sigma^2})dx$.

Using integralcalculator.com with the integral's limits being set from $-\infty$ to ∞, since the time to manually compute this integral is not really worth it, I found that $Z = \sqrt{2\pi}\sigma$.

■

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 $||\boldsymbol{\theta} b^\star||_2$ where $\boldsymbol{\theta} b$ is your weight vector. What is the final RMSE on the test set with the optimal $\lambda^\star$?

(continued on the following pages)

a) We are given the maximum a posteriori problem $max_w \mathbb{P}(\mathbf{w}|D) = max_w \mathbb{P}(D|\mathbf{w})\mathbb{P}(\mathbf{w})$ in the logarithmic form: $argmax_w \sum_{i=1}^{N} log\mathcal{N}(y_i|w_0 + \mathbf{w}^T x_i, \sigma^2) + \sum_{j=1}^{D} log\mathcal{N}(w_j|0,\tau^2)$

Recall the probability distribution $\mathcal{N}(x|\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} exp(-\frac{(x-\mu)^2}{2\sigma^2})$. Use this as a substitution in the above term. Then, we get

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

Using logarithm properties like splitting products in logarithm arguments, the previous quantity equals

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

Combining summations with like terms and separating summations when they have different terms:

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

Because the constant $-(N+D) log\sqrt{2\pi}\sigma$ does not affect the optimal value $\mathbf{w}^*$, scale the problem up by $2\sigma^2$ without changing $\mathbf{w}^*$. Since maximizing a function is equivalent to minimizing its negative, we can optimize this problem instead:

$argmin_w \sum_{i=1}^{N} (y_i - w_0 - \mathbf{w}_i^T)^@ + \frac{\sigma^2}{\tau^2} \sum_{j=1}^{D} w_j^2.$

Let $\lambda = \frac{\sigma^2}{\tau^2}$. Then, we have $argmin_w \sum_{i=1}^{N} (y_i - w_0 - \mathbf{w}^T x_i)^2 + \lambda \sum_{j=1}^{D} w_j^2$, which equals $argmin_w \sum_{i=1}^{N} (y_i - (w_0 + \mathbf{w}^T x_i))^2 + \lambda ||\mathbf{w}||_2^2$, the ridge regression problem, as desired.

b) First, decompose the norm $||A\mathbf{x} - \mathbf{b}||_2^2 + ||\Gamma\mathbf{x}||_2^2$ into $(A\mathbf{x} - \mathbf{b})^T (A\mathbf{x} - \mathbf{b}) + (\Gamma\mathbf{x})^T (\Gamma\mathbf{x})$. We will now minimize this.

Find the gradient of f with respect to $\mathbf{x}$ and set it equal to 0:

$\nabla_x f = \nabla_x [(A\mathbf{x} - \mathbf{b})^T (A\mathbf{x} - \mathbf{b}) + (\Gamma\mathbf{x})^T (\Gamma\mathbf{x})] =$
$\nabla_x [(\mathbf{x}^T A^T - \mathbf{b}^T)(A\mathbf{x} - \mathbf{b}) + \mathbf{x}^T \Gamma^T \Gamma \mathbf{x}]$
$= \nabla_x [\mathbf{x}^T A^T A\mathbf{x} - \mathbf{x}^T A^T \mathbf{b} - \mathbf{b}^T A\mathbf{x} + \mathbf{b}^T \mathbf{b} + \mathbf{x}^T \Gamma^T \Gamma \mathbf{x}]$
$= \nabla_x [\mathbf{x}^T A^T A\mathbf{x} - 2\mathbf{x}^T A^T \mathbf{b} + \mathbf{b}^T \mathbf{b} + \mathbf{x}^T \Gamma^T \Gamma \mathbf{x}]$
$= 2A^T A\mathbf{x} - 2A^T \mathbf{b} + 2\Gamma^T \Gamma \mathbf{x}.$

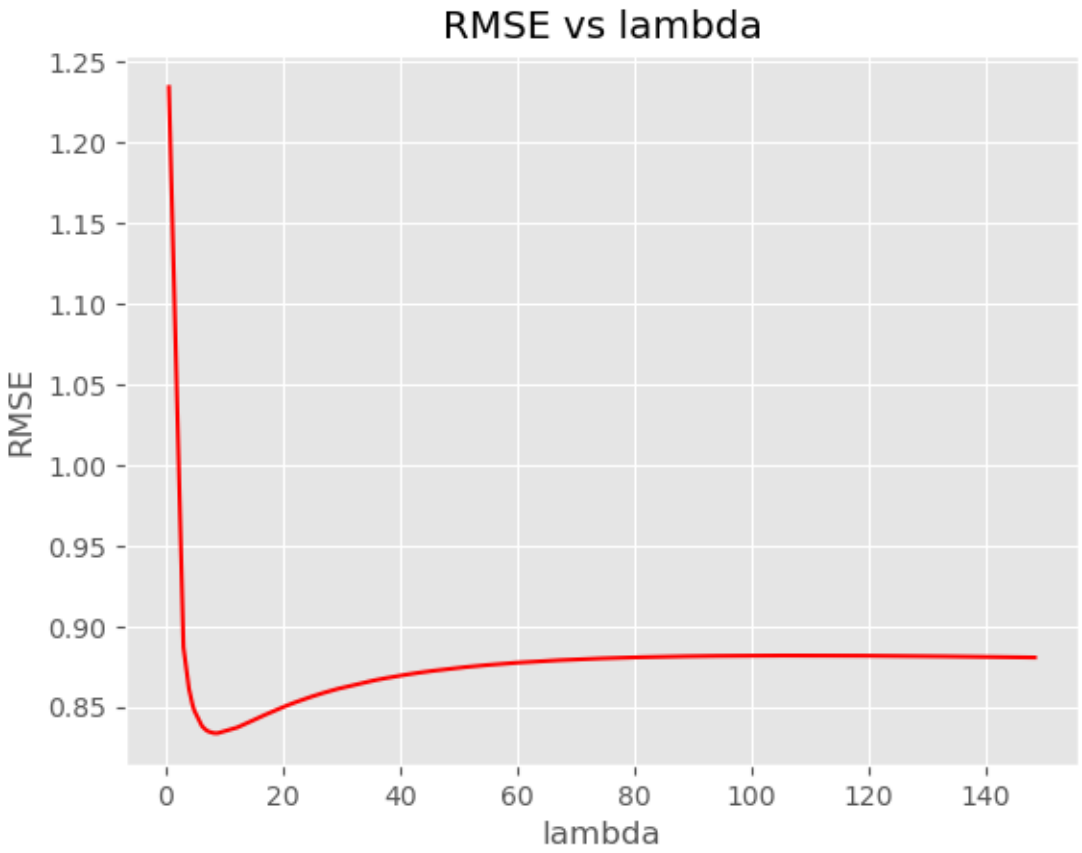
$0 = 2A^T A\mathbf{x} - 2A^T \mathbf{b} + 2\Gamma^T \Gamma \mathbf{x}$ can be simplified into $A^T \mathbf{b} = \mathbf{x}(A^T A + \Gamma^T \Gamma)$, which implies that $x^* = (A^T A + \Gamma^T \Gamma)^{-1} A^T \mathbf{b}$.

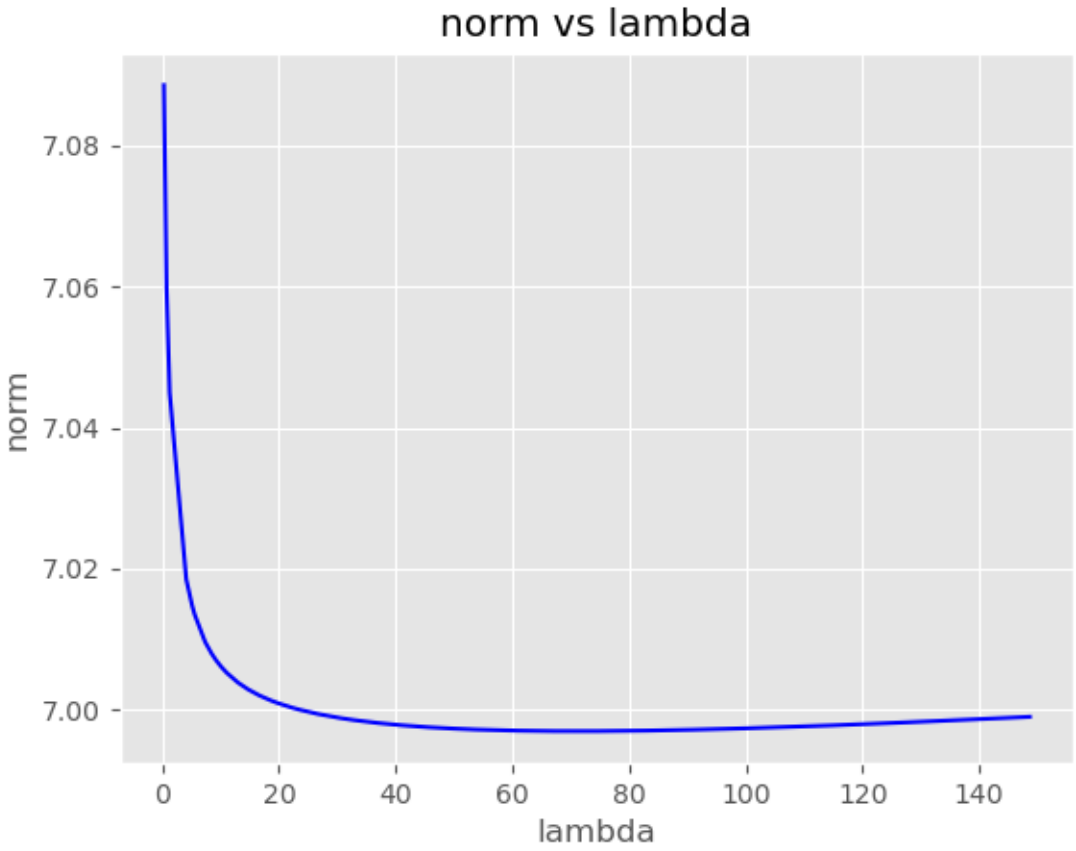
Recall the normal equation $\theta = (X^T X)^{-1} X^T \mathbf{y}$. Here, $\mathbf{x}^* = \theta, \mathbf{b} = \mathbf{y}, X^T = A^T, X^T X = A^T A + \Gamma^T \Gamma$.
The closed form solution is $\mathbf{x}^* = (A^T A + \Gamma^T \Gamma)^{-1} A^T \mathbf{b}$.

Letting $\Gamma = \sqrt(\lambda)$, we can minimize $f = ||A\mathbf{x} - \mathbf{b}||_2^2 + \lambda \mathbf{x}^T \mathbf{x}$.

Then, using the above closed form, the optimal solution is $\mathbf{x}^* = (A^T A + \lambda I)^{-1} A^T \mathbf{b}$.

c) The optimal regularization parameter $\lambda^* = 8.8922$ with a validation set RMSE of 0.8340 and a test set RMSE of 0.8628. Below are the plots:


RMSE vs lambda
RMSE
lambda
1.25
1.20
1.15
1.10
1.05
1.00
0.95
0.90
0.85
0
20
40
60
80
100
120
140


norm vs lambda
7.08
7.06
7.04
7.02
7.00
norm
0
20
40
60
80
100
120
140
lambda

■

3 (continued)

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

d) First, decompose the norm $f = ||A\mathbf{x} + b1 - \mathbf{y}||_2^2 + ||\Gamma\mathbf{x}||_2^2$ into $(A\mathbf{x} + b\mathbf{1} - \mathbf{y})^T(A\mathbf{x} + b\mathbf{1} - \mathbf{y}) + (\Gamma\mathbf{x})^T(\Gamma\mathbf{x})$.

Now, we minimize and work with $f = ||A\mathbf{x} + b\mathbf{1} - \mathbf{y}||_2^2 + ||\Gamma\mathbf{x}||_2^2$
$= (A\mathbf{x} + b\mathbf{1} - \mathbf{y})^T(A\mathbf{x} + b\mathbf{1} - \mathbf{y}) + (\Gamma\mathbf{x})^T(\Gamma\mathbf{x})$
$= (\mathbf{x}^T A^T + b\mathbf{1} - \mathbf{y})^T(A\mathbf{x} + b\mathbf{1} - \mathbf{y}) + \mathbf{x}^T\Gamma^T\Gamma\mathbf{x}$
$= \mathbf{x}^T A^T A\mathbf{x} + \mathbf{x}^T A^T b\mathbf{1} - \mathbf{x}^T A^T\mathbf{y} + \mathbf{1}^T b^T A\mathbf{x} + b^2 n - \mathbf{1}^T b^T\mathbf{y} - \mathbf{y}^T A\mathbf{x} - \mathbf{y}^T b\mathbf{1} + \mathbf{y}^T\mathbf{y} + \mathbf{x}^T\Gamma^T\Gamma\mathbf{x}$
$= \mathbf{x}^T A^T A\mathbf{x} + 2b\mathbf{1}^T A\mathbf{x} - 2\mathbf{y}^T A\mathbf{x} - 2b\mathbf{1}^T\mathbf{y} + b^2 n + \mathbf{y}^T\mathbf{y} + \mathbf{x}^T\Gamma^T\Gamma\mathbf{x}$.

Find the gradient of f with respect to $\mathbf{x}$ and b. At optimality, we have $\nabla_x f = 0$, so we have

$\nabla_x f = \nabla_x(||A\mathbf{x} + b1 - \mathbf{y}||_2^2 + ||\Gamma\mathbf{x}||_2^2) = 2A^T A\mathbf{x} + A^T b\mathbf{1} + \mathbf{1}^T b\mathbf{1}A - A^T\mathbf{y} - \mathbf{y}^T A + 2\Gamma^T\Gamma\mathbf{x} =$
$2A^T A\mathbf{x} + 2bA^T\mathbf{1} - 2A^T\mathbf{y} + 2\Gamma^T\Gamma\mathbf{x} = (A^T A + \Gamma^T\Gamma)\mathbf{x} + bA^T\mathbf{1} - A^T\mathbf{y} = 0$

$\nabla_b f = \nabla_b(||A\mathbf{x} + b1 - \mathbf{y}||_2^2 + ||\Gamma\mathbf{x}||_2^2) = x^T A^T\mathbf{1} + \mathbf{1}^T A\mathbf{x} - \mathbf{1}^T\mathbf{y} - \mathbf{y}^T\mathbf{1} + 2by = 2\mathbf{1}^T A\mathbf{x} -$
$2\mathbf{1}^T\mathbf{y} + 2bn = 0$

Isolate for the bias term, so $bn = \mathbf{1}^T\mathbf{y} - \mathbf{1}^T A\mathbf{x}$.
Then, $b^* = \frac{\mathbf{1}^T(\mathbf{y} - A\mathbf{x})}{n}$.

This is reasonable because if the line is flat, meaning $(\mathbf{x} = 0)$, then $b^* = \frac{\mathbf{1}^T\mathbf{y}}{n}$, so the bias term becomes the mean value of the output, which is reasonable.

Plug b^* back as the b in the equation for the gradient of f with respect to x to solve for $\mathbf{x}^*$:

$(A^T A + \Gamma^T\Gamma)\mathbf{x} + (\frac{\mathbf{1}^T(\mathbf{y} - A\mathbf{x})}{n})A^T 1 - A^T\mathbf{y} = 0$

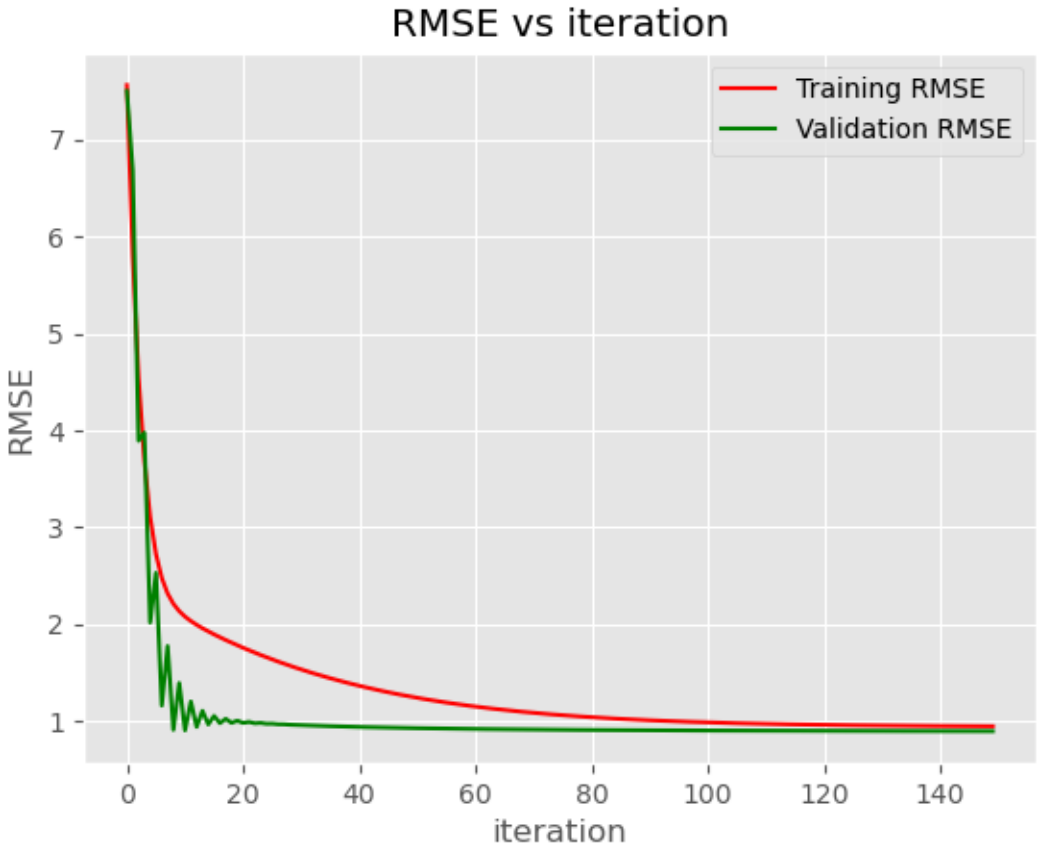
$(A^TA + \Gamma^T\Gamma)\mathbf{x} + \frac{1}{n}A^T11^T\mathbf{y} - \frac{1}{n}A^T\mathbf{1}\mathbf{1}^TA\mathbf{x} - A^T\mathbf{y} = 0$ by the distributive property
$[A^TA + \Gamma^T\Gamma - \frac{1}{n}A^T\mathbf{1}\mathbf{1}^TA]\mathbf{x} = A^T\mathbf{y} - \frac{1}{n}A^T\mathbf{1}\mathbf{1}^T\mathbf{y}$ by factoring and setting terms on the right side
$[A^T(I - \frac{1}{n}\mathbf{1}\mathbf{1}^T)A + \Gamma^T\Gamma]\mathbf{x} = A^T(I - \frac{1}{n}\mathbf{1}\mathbf{1}^T)\mathbf{y}$ by factoring
Recall the normal equation $\theta = (X^TX)^{-1}X^T\mathbf{y}$. Here, $\mathbf{x}^* = \theta, X^T = A^T(I - \frac{1}{n}\mathbf{1}\mathbf{1}^T), X^TX = A^T(I - \frac{1}{n}\mathbf{1}\mathbf{1}^T)A + \Gamma^T\Gamma$.
$\mathbf{x}^* = [A^T(I - \frac{1}{n}\mathbf{1}\mathbf{1}^T)A + \Gamma^T\Gamma]^{-1}A^T(I - \frac{1}{n}11^T)\mathbf{y}$

where I is the identity matrix, $\mathbf{1}$ is a vector of all ones, and $y \in \mathbb{R}$.

We compute the bias term using the code, and we get negligible differences:

Difference in bias: 4.2226E-10
Difference in weights: 5.5357E-10

e) Below is the convergence plot of RMSE vs. iteration, where the training RMSE and validation RMSE are plotted:



Compare the results to bias, and weights obtained from parts c and d. Then, we have:
Difference in bias: 1.5386E-01 Difference in weights: 7.9692E-01

■