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. Using chain rule,

$$\begin{aligned}
\sigma'(x) &= -(1+e^{-x})^{-2}(-e^{-x}) \\
&= \frac{1}{(1+e^{-x}}\frac{e^{-x}}{1+e^{-x}} \\
&= \sigma(x)(1-\sigma(x))
\end{aligned}$$

b. The negative log likelihood equation is

$$NLL(\mathbf{w}) = -\Sigma_{i=1}^{N}[y_i \log \sigma(\mathbf{w}^T x_i) + (1-y_i) log(1-\sigma(\mathbf{w}^T x_i))]$$

Taking the derivative and using the result from a), we get

$$\begin{aligned}
\nabla NLL(\mathbf{w}) &= -\Sigma_{i=1}^{N}[y_i \frac{d}{d\mathbf{w}}(\log \sigma(\mathbf{w}^T x_i)) + \frac{d}{d\mathbf{w}}(1-y_i)\log(1-\sigma(\mathbf{w}b^T x_i))] \\
&= -\Sigma_{i=1}^{N}[y_i \frac{1}{\sigma(\mathbf{w}^T x_i)}\sigma'(\mathbf{w}^T x_i) + (1-y_i)\frac{1}{1-\sigma(\mathbf{w}^T x_i)}(-\sigma'(\mathbf{w}^T x_i))] \\
&= -\Sigma_{i=1}^{N}[y_i(1-\sigma(\mathbf{w}^T x_i))x_i - (1-y_i)\sigma(\mathbf{w}^T x_i))x_i] \\
&= -\Sigma_{i=1}^{N}[y_i x_i - y_i\sigma(\mathbf{w}^T x_i)x_i) - \sigma(\mathbf{w}^T x_i x_i) + y_i\sigma(\mathbf{w}^T x_i))x_i] \\
&= -\Sigma_{i=1}^{N}(y_i x_i - \sigma(\mathbf{w}^T x_i)x_i)
\end{aligned}$$

Let $\mu_i = \sigma(\mathbf{w}^T x_i)$. Then,

$$\begin{aligned}\nabla NLL(\mathbf{w}) &= -\Sigma_{i=1}^{N}(y_i x_i - \mu_i x_i) \\ &= X^T(\boldsymbol{\mu} - \mathbf{y})\end{aligned}$$

c. In order to find the Hessian matrix, we need to the derivative from the gradient found in (b):

$$\begin{aligned}H &= \nabla(\nabla NLL\mathbf{w})^T \\ &= \nabla(X^T\mu - X^T\mathbf{y})^T\end{aligned}$$

Using the matrix transpose distribution rule, $H = \nabla(\mu^T X - \mathbf{y}^T X)$. If we substitute σ back in, since the second term is not dependent on w, we can disregard it:

$$\begin{aligned}H &= \nabla\boldsymbol{\mu}^T X \\ &= \nabla(\sigma(\mathbf{w}^T X))^T X \\ &= \nabla\sigma(X\mathbf{w})^T X \\ &= X^T diag(\boldsymbol{\mu}(1-\boldsymbol{\mu}))X \\ &= X^T S X\end{aligned}$$

By definition, H is positive semi-definite if all of its eigenvalues are non-negative. Since H can be diagonalized as $X^T SX$, the diagonal entries of S are its eigenvalues and we need to show that they are non-negative. Since $\mu_i = \frac{1}{(1+e^{-\mathbf{w}^T x_i})}$, $\mu_i > 0$ and $\mu_i < 1$, thus $1 - \mu_i > 0$. Therefore, the diagonal entries, or in other words the eigenvalues of H, $\mu_i(1-\mu_i) > 0$, so H must be positive semid-efinite.

■

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.

In order for the distribution to be a valid density, $\int_{-\infty}^{\infty} \frac{1}{Z} e^{-\frac{x^2}{2\sigma^2}} dx = 1$. Rewriting this, we get $\int_{-\infty}^{\infty} e^{-\frac{x^2}{2\sigma^2}} dx = Z$. Since this form cannot be integrated easily, the textbook hints on computing Z^2 instead, using polar coordinate.

$$\begin{aligned}
Z^2 &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{-\frac{x^2+y^2}{2\sigma^2}} dxdy \\
&= \int_0^{2\pi}\int_0^{\infty} e^{-\frac{x^2+y^2}{2\sigma^2}} drd\theta \\
&= \int_0^{2\pi}\int_0^{\infty} e^{-\frac{r^2}{2\sigma^2}} rdrd\theta \\
&= \int_0^{2\pi}\int_0^{\infty} \sigma^2 e^{-u} dud\theta \\
&= \int_0^{2\pi} \sigma^2 d\theta \\
&= 2\pi\sigma^2
\end{aligned}$$

Thus, $Z = \sqrt{2\pi}\sigma$ ∎

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 $||\boldsymbol{\theta}^\star||_2$ 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) "For a Gaussian distribution, $N(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \cdot e^{-\left(\frac{(x-\mu)^2}{2\sigma^2}\right)}$. Thus, if we substitute in the MAP problem, we get

$$\underset{w}{\text{argmax}} \sum_{i=1}^{N} \log \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{(y_i - w_0 - \vec{w}^T\vec{x}_i)^2}{2\sigma^2}\right)} + \sum_{j=1}^{D} \log \frac{1}{\sqrt{2\pi}\sigma} e^{\left(-\frac{w_j^2}{2\tau^2}\right)}$$

$$= \underset{w}{\text{argmax}} -\sum_{i=1}^{N} \left(\frac{(y_i - w_0 - \vec{w}^T\vec{x}_i)^2}{2\sigma^2} + \log\sqrt{2\pi}\sigma\right) - \sum_{j=1}^{D}\left(\frac{w_j^2}{2\tau^2} + \log\sqrt{2\pi}\sigma\right)$$

(constants)

$$= \underset{w}{\text{argmax}} -\left[\log\sqrt{2\pi}\sigma \cdot (N+D)) + \sum_{i=1}^{N} \frac{(y_i - w_0 - \vec{w}^T\vec{x}_i)^2}{2\sigma^2} + \sum_{j=1}^{D} \frac{w_j^2}{2\tau^2}\right]$$

Since these are constants, and argmax answers where the max occurs and not what the max is, they can be ignored. Scaling can also be ignored. Thus, it becomes

$$\underset{\text{argmax}}{\text{Negative of}} - \left(\sum_{i=1}^{N} (y_i - w_0 - \vec{w}^T\vec{x}_i)^2 + \frac{\sigma^2}{\tau^2}\sum_{i=j}^{D} w_j^2\right)$$

Negative of argmax = argmin.

$$= \text{argmin}\left(\sum_{i=1}^{N} (y_i - w_0 - \vec{w}^T\vec{x}_i)^2 + \lambda\sum_{j=1}^{D} (w_j)^2\right)$$

Note that $||x_i||_p = \left(\sum_i |x_i|^p\right)^{1/p}$

$$= \text{argmin} \sum_{i=1}^{N} (y_i - w_0 - \vec{w}^T\vec{x}_i)^2 + \lambda ||w||_2^2$$

■

Continue Problem 3

b) Suppose $f(\vec{x}) = \|A\vec{x} - \vec{b}\|_2^2 + \|\Gamma\vec{x}\|_2^2$

In order to minimize $f(\vec{x})$, we need to find its derivative:

$$\nabla f(\vec{x}) = \nabla\left[(A\vec{x}-\vec{b})^T(A\vec{x}-\vec{b}) + (\Gamma\vec{x})^T(\Gamma\vec{x})\right] \quad \text{<distribute } T\text{>}$$

$$= \nabla\left[(\vec{x}^TA^T - \vec{b}^T)(A\vec{x}-\vec{b}) + \vec{x}^T\Gamma^T\Gamma\vec{x}\right]$$

$$= \nabla\left(\vec{x}^TA^TA\vec{x} - \vec{x}^TA^T\vec{b} - \vec{b}^TA\vec{x} + \vec{b}^T\vec{b} + \vec{x}^T\Gamma^T\Gamma\vec{x}\right)$$

note that $\vec{x}^TA^T\vec{b} = \vec{b}^TA\vec{x}$

$$= \nabla\left(\vec{x}^TA^TA\vec{x} - 2\vec{x}^TA^T\vec{b} + \vec{b}^T\vec{b} + \vec{x}^T\Gamma^T\Gamma\vec{x}\right)$$

$$= 2A^TA\vec{x} - 2A^T\vec{b} + 2\Gamma^T\Gamma\vec{x}$$

Solve for $\vec{x}$ by setting $\nabla_x f = 0$:

$$0 = 2\left(A^TA\vec{x} - A^T\vec{b} + \Gamma^T\Gamma\vec{x}\right)$$

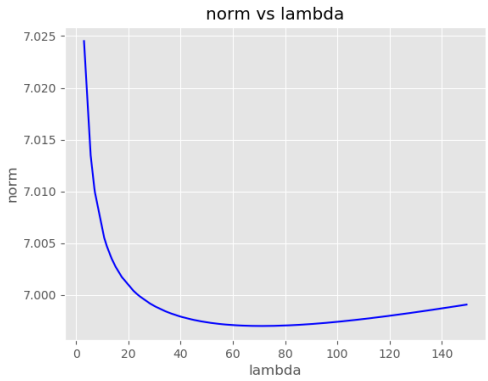
$$A^T\vec{b} = A^TA\vec{x} + \Gamma^T\Gamma\vec{x}$$

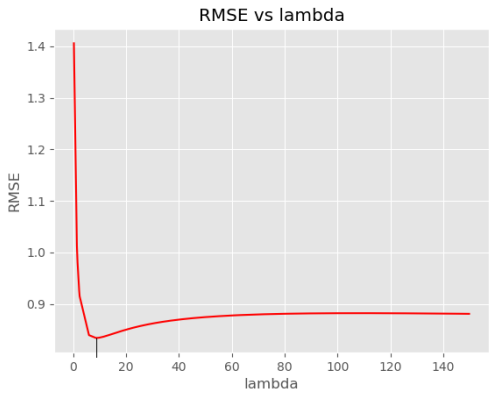
$$A^T\vec{b} = (A^TA + \Gamma^T\Gamma)\vec{x}$$

$$\boxed{\vec{x} = (A^TA + \Gamma^T\Gamma)^{-1}A^T\vec{b}}$$

(continued part 3 on next page)

c)





The optimal RSME on validation set is 0.8340.

The optimal RSME on test set is 0.8628.

$\lambda^* = 9.0262$

d) $f(\vec{x}) = \|A\vec{x} + b\vec{1} - \vec{y}\|_2^2 + \|\Gamma\vec{x}\|^2$

Norm 2 can be re-written into the following:

$f(\vec{x}) = (A\vec{x} + b\vec{1} - \vec{y})^T(A\vec{x} + b\vec{1} - \vec{y}) + (\Gamma\vec{x})^T(\Gamma\vec{x})$

$= (\vec{x}^TA^T + \vec{1}^T \cdot b^T - \vec{y}^T)(A\vec{x} + b\vec{1} - \vec{y}) + (\vec{x}^T\Gamma^T)(\Gamma\vec{x})$

$= \vec{x}^TA^TA\vec{x} + \vec{x}^TA^Tb\vec{1} - \vec{x}^TA^T\vec{y} + \vec{1}^Tb^TA\vec{x} + \vec{1}^Tb^Tb\vec{1} - \vec{1}^Tb^T\vec{y}$
$- \vec{y}^TA\vec{x} - \vec{y}b\vec{1} + \vec{y}^T\vec{y} + \vec{x}^T\Gamma^T\Gamma\vec{x}$

$= \vec{x}^TA^TA\vec{x} + 2\vec{x}^TA^Tb\vec{1} - 2\vec{x}^TA^T\vec{y} - 2\vec{1}^Tb\vec{y} + \vec{1}^Tb^2\vec{1}$
$+ \vec{y}^T\vec{y} + \vec{x}^T\Gamma^T\Gamma\vec{x}$

To find the optimal b, we need to set $0 = \nabla_b f(\vec{x})$

$0 = \nabla_b(\vec{x}^TA^TA\vec{x} + 2\vec{x}^TA^Tb\vec{1} - 2\vec{x}^TA^T\vec{y} - 2\vec{1}^Tb\vec{y} + \vec{1}^Tb^2\vec{1}$
$+ \vec{y}^T\vec{y} + \vec{x}^T\Gamma^T\Gamma\vec{x})$
$= 2\vec{x}^TA^T\vec{1} - 2\vec{1}^T\vec{y} + 2bn$

$\vec{1}$ has dimension $1 \times n$ so $\vec{1}^Tb^2\vec{1} = b^2n$

Solving for b, we get

$b = \frac{\vec{1}^T\vec{y} - \vec{x}^TA^T\vec{1}}{n} = \frac{\vec{1}^T(\vec{y} - A\vec{x})}{n}$ (1)

To find the optimal $\vec{x}$, we need to set $0 = \nabla_x f(\vec{x})$

$0 = \nabla_x f(\vec{x})$
$= \nabla_x(\vec{x}^TA^TA\vec{x} + 2\vec{x}^TA^Tb\vec{1} - 2\vec{x}^TA^T\vec{y} - 2\vec{1}^Tb\vec{y} + \vec{1}^Tb^2\vec{1}$
$+ \vec{y}^T\vec{y} + \vec{x}^T\Gamma^T\Gamma\vec{x})$ — not dependent on x
$= 2A^TA\vec{x} + 2A^Tb\vec{1} - 2A^T\vec{y} + 2\Gamma^T\Gamma\vec{x}$

(continue)

If we substitute in the optimal b,

$$0 = A^T A\vec{x} + A^T\left[\frac{\vec{1}^T(\vec{y} - A\vec{x})}{n}\right]\vec{1} - A^T\vec{y} + \vec{1}^T\vec{1}\vec{x}$$

$$0 = \left(A^T A - \frac{A^T\vec{1}\vec{1}^T A\vec{1}}{n} + \vec{1}^T\vec{1}\right)\vec{x} + \frac{A^T\vec{1}\vec{1}^T\vec{y}\,\vec{1}}{n} - A^T\vec{y}$$

$$\vec{x} = \left[A^T\left(I - \frac{\vec{1}\vec{1}^T}{n}\right)A + \vec{1}^T\vec{1}\right]^{-1} A^T\left(I - \frac{\vec{1}\vec{1}^T}{n}\right)\vec{y} \qquad (2)$$

Substituting (2) in code, we get:

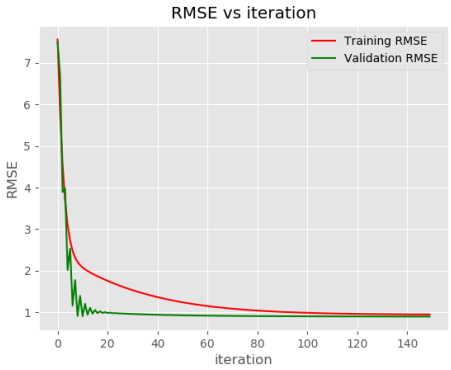
difference in bias = $4.6495 \times 10^{-10} \approx 0$

difference in weights = $6.2312 \times 10^{-10} \approx 0$

$\Rightarrow$ same as previous

e) difference in bias = 1.5386×10^{-1}

difference in weights = 7.9639×10^{-1}



FYI:

since $\vec{x}^T A^T b \vec{1} = \begin{bmatrix} (x_1a_{11} + x_2a_{21} + \cdots) & (x_1a_{12} + x_2a_{22} + \cdots) & \cdots \end{bmatrix} \begin{bmatrix} b \\ b \\ \vdots \end{bmatrix}$

$= (x_1a_{11} + x_2a_{21} + \cdots)b + (x_1a_{12} + x_2a_{22} + \cdots)b + \cdots (x_na_{1n} + x_na_{2n} + \cdots)b$

$= \begin{bmatrix} b & b & \cdots \end{bmatrix} \begin{bmatrix} (x_1a_{11} + x_2a_{21} + \cdots) \\ (x_1a_{12} + x_2a_{22} + \cdots) \\ \vdots \end{bmatrix}$

$= \vec{1}b \begin{bmatrix} a_{11} & a_{21} & \cdots \\ a_{12} & a_{22} & \\ \vdots & & \ddots \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \end{bmatrix}$

$= \vec{1}bA^T\vec{x}$