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 $\mathbf{S} = \text{diag}(\mu_1(1 - \mu_1), \dots, \mu_n(1 - \mu_n))$. 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.

---

1. Taking the derivative,

$$\sigma'(x) = \frac{e^{-x}}{(1 + e^{-x})^2}. \tag{1}$$

   Then, a simple calculation shows that

$$\frac{1}{1 + e^{-x}} - \frac{1}{(1 + e^{-x})^2} = \sigma'(x). \tag{2}$$

2. From the hint, the negative log-likelihood for logistic regression is

$$\text{NLL}(\mathbf{w}) = -\sum_{i=1}^{N} \left[y_i \log \mu_i + (1 - y_i)\log(1 - \mu_i)\right]. \tag{3}$$

   By definition,

$$\mu(\mathbf{x}) = \sigma(\mathbf{w}^T \mathbf{x}). \tag{4}$$

Hence,

$$\text{NLL}(\mathbf{w}) = -\sum_{i=1}^{N}\left[y_i \log \sigma(\mathbf{w}^T\mathbf{x}_i) + (1-y_i)\log\left(1 - \sigma(\mathbf{w}^T\mathbf{x}_i)\right)\right]. \tag{5}$$

Which means that

$$\nabla_{\mathbf{w}}\text{NLL}(\mathbf{w}) = -\sum_{i=1}^{N}\left[\frac{y_i}{\sigma(\mathbf{w}^T\mathbf{x})}\cdot\sigma'(\mathbf{w}^T\mathbf{x}_i)\cdot\mathbf{x}_i - \frac{1-y_i}{1-\sigma(\mathbf{w}^T\mathbf{x}_i)}\cdot\sigma'(\mathbf{w}^T\mathbf{x}_i)\cdot\mathbf{x}_i\right] \tag{6}$$

$$= -\sum_{i=1}^{N}\left[\frac{y_i(1-\sigma(\mathbf{w}^T\mathbf{x}_i))}{\sigma'(\mathbf{w}^T\mathbf{x}_i)}\sigma'(\mathbf{w}^T\mathbf{x}_i)\mathbf{x}_i - \frac{(1-y_i)\sigma(\mathbf{w}^T\mathbf{x}_i)}{\sigma'(\mathbf{w}^T\mathbf{x}_i)}\sigma'(\mathbf{w}^T\mathbf{x}_i)\mathbf{x}_i\right] \tag{7}$$

$$= \sum_{i=1}^{N}(\sigma(\mathbf{w}^T\mathbf{x}_i) - y_i)\mathbf{x}_i \tag{8}$$

using the results of part (a). Substituting, we have

$$= \sum_{i=1}^{N}(\mu_i - y_i)\mathbf{x}_i \tag{9}$$

$$= \mathbf{X}^T(\boldsymbol{\mu} - \mathbf{y}), \tag{10}$$

as desired.

3. The Hessian for logistic regression is

$$\mathbf{H} = \nabla_{\mathbf{w}}\left(\nabla_{\mathbf{w}}\text{NLL}(\mathbf{w})\right)^T, \tag{11}$$

which means that (using part (b) – and then part (a))

$$\mathbf{H} = \nabla_{\mathbf{w}}\left(\mathbf{X}^T(\boldsymbol{\mu} - \mathbf{y})\right)^T \tag{12}$$

$$= \nabla_{\mathbf{w}}\left((\boldsymbol{\mu}^T - \mathbf{y}^T)\mathbf{X}\right) \tag{13}$$

$$= \nabla_{\mathbf{w}}\left(\boldsymbol{\mu}^T\mathbf{X}\right) \tag{14}$$

$$= \nabla_{\mathbf{w}}\left[\sigma(\mathbf{w}^T\mathbf{X})\right]^T\mathbf{X} \tag{15}$$

$$= \mathbf{X}^T\sigma'(\mathbf{w}^T\mathbf{X})^T\mathbf{X} \tag{16}$$

$$= \mathbf{X}^T\left(\boldsymbol{\mu}(1-\boldsymbol{\mu})\right)^T\mathbf{X} \tag{17}$$

$$= \mathbf{X}^T\mathbf{S}\mathbf{X} \tag{18}$$

To show $\mathbf{H}$ is positive semidefinite, it suffices to show that $\mathbf{S}$ has nonnegative eigenvalues. But this is true by definition, since $\mathbf{S}$ is a diagonal matrix and $0 \leq \mu \leq 1$ ($0 \leq \sigma \leq 1$), so each diagonal entry $\mu_i(1-\mu_i) \geq 0$. This concludes the proof.

∎

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.

We must find Z such that

$$\frac{1}{Z} \int_{\mathbb{R}} \exp\left(-\frac{x^2}{2\sigma^2}\right) dx = 1. \tag{19}$$

(took this hint from solution) this is equivalent to finding Z such that

$$\frac{1}{Z^2} \int_{\mathbb{R}} \exp\left(-\frac{x^2}{2\sigma^2}\right) dx \cdot \int_{\mathbb{R}} \exp\left(-\frac{y^2}{2\sigma^2}\right) dy = 1 \cdot \int_{\mathbb{R}} \exp\left(-\frac{y^2}{2\sigma^2}\right) dy = 1. \tag{20}$$

In other words, finding Z such that

$$\int_{\mathbb{R}^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) dxdy = Z^2. \tag{21}$$

Changing from Cartesian to polar coordinates, we have:

$$\int_0^{2\pi} \int_0^{\infty} r \exp\left(-\frac{r^2}{2\sigma^2}\right) dr\, d\theta = Z^2 \tag{22}$$

after doing the substitution $u = r^2$ and $du = 2r$,

$$Z^2 = \frac{1}{2} \int_0^{2\pi} \int_0^{\infty} \exp\left(-\frac{u}{2\sigma^2}\right) dr\, d\theta = \int_0^{2\pi} \sigma^2\, d\theta = 2\pi\sigma^2. \tag{23}$$

Thus,

$$Z = \sqrt{2\pi}\sigma. \tag{24}$$

∎

3

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

(continued on the following pages)

4

3 (continued)

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

1. Since

$$\mathcal{N}(\mathbf{x} \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(\mathbf{x} - \mu)^2}{2\sigma^2}\right), \tag{25}$$

we begin by performing the appropriate substitution to the given maximum a posteriori problem. We have,

$$\arg\max_{\mathbf{w}} \sum_{i=1}^{N} \log \mathcal{N}(y_i | w_0 + \mathbf{w}^\top \mathbf{x}_i, \sigma^2) + \sum_{j=1}^{D} \log \mathcal{N}(w_j | 0, \tau^2) \tag{26}$$

$$= \arg\max_{\mathbf{w}} \sum_{i=1}^{N} \log \left[\frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(y_i - w_0 - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2}\right) \right]$$

$$+ \sum_{j=1}^{D} \log \left[\frac{1}{\sqrt{2\pi}\tau} \exp\left(-\frac{w_j^2}{2\tau^2}\right) \right] \tag{27}$$

$$= -\arg\max_{\mathbf{w}} \sum_{i=1}^{N} \left[\log\left(\sqrt{2\pi}\sigma\right) + \left(\frac{(y_i - w_0 - \mathbf{w}^T\mathbf{x}_i)^2}{2\sigma^2}\right) \right]$$

$$- \sum_{j=1}^{D} \left[\log\left(\sqrt{2\pi}\tau\right) + \left(\frac{w_j^2}{2\tau^2}\right) \right] \tag{28}$$

$$= -\arg\max_{\mathbf{w}} \left[N \log\left(\sqrt{2\pi}\sigma\right) + D \log\left(\sqrt{2\pi}\tau\right) + \sum_{i=1}^{N} \left(\frac{(y_i - w_0 - \mathbf{w}^T \mathbf{x}_i)^2}{2\sigma^2} \right) \right.$$
$$\left. + \sum_{j=1}^{D} \left(\frac{w_j^2}{2\tau^2} \right) \right] \quad (29)$$

via logarithm rules. Recognizing that the $-\arg\max_{\mathbf{w}} = \arg\min_{\mathbf{w}}$ we have

$$= \arg\min_{\mathbf{w}} \left[N \log\left(\sqrt{2\pi}\sigma\right) + D \log\left(\sqrt{2\pi}\tau\right) + \sum_{i=1}^{N} \left(\frac{(y_i - w_0 - \mathbf{w}^T \mathbf{x}_i)^2}{2\sigma^2} \right) \right.$$
$$\left. + \sum_{j=1}^{D} \left(\frac{w_j^2}{2\tau^2} \right) \right] \quad (30)$$

$$= \arg\min_{\mathbf{w}} \left[\sum_{i=1}^{N} \left(\frac{(y_i - w_0 - \mathbf{w}^T \mathbf{x}_i)^2}{2\sigma^2} \right) + \sum_{j=1}^{D} \left(\frac{w_j^2}{2\tau^2} \right) \right] \quad (31)$$

but multiplying by a factor of $2\sigma^2$ to the argument of the $\arg\min_{\mathbf{w}}$ function does not change the output of the $\arg\min_{\mathbf{w}}$ function, so performing the multiplication:

$$= \arg\min_{\mathbf{w}} \left[\sum_{i=1}^{N} \left(y_i - w_0 - \mathbf{w}^T \mathbf{x}_i \right)^2 + \sum_{j=1}^{D} \left(\frac{\sigma^2}{\tau^2} \cdot w_j^2 \right) \right] \quad (32)$$

$$= \arg\min_{\mathbf{w}} \left[\sum_{i=1}^{N} \left(y_i - (w_0 + \mathbf{w}^T \mathbf{x}_i) \right)^2 + \lambda ||\mathbf{w}||_2^2 \right] \quad (33)$$

we get the desired result.

2. Note that we can decompose the norm in the following way:

$$||A\mathbf{x} - \mathbf{b}||_2^2 + ||\Gamma\mathbf{x}||_2^2 = (A\mathbf{x} - \mathbf{b})^T(A\mathbf{x} - \mathbf{b}) + (\Gamma\mathbf{x})^T(\Gamma\mathbf{x}). \quad (34)$$

We are asked to minimize the problem with respect to $\mathbf{x}$. Hence, taking the gradient, we have

$$\nabla_{\mathbf{x}} \left[||A\mathbf{x} - \mathbf{b}||_2^2 + ||\Gamma\mathbf{x}||_2^2 \right] = \nabla_{\mathbf{x}} \left[(A\mathbf{x} - \mathbf{b})^T(A\mathbf{x} - \mathbf{b}) + (\Gamma\mathbf{x})^T(\Gamma\mathbf{x}) \right] \quad (35)$$

$$= \nabla_{\mathbf{x}} \left[(\mathbf{x}^T A^T - \mathbf{b}^T)(A\mathbf{x} - \mathbf{b}) + \mathbf{x}^T \Gamma^T \Gamma \mathbf{x} \right] \quad (36)$$

$$= \nabla_{\mathbf{x}} \left[\mathbf{x}^T A^T A\mathbf{x} - \mathbf{x}^T A^T \mathbf{b} - \mathbf{b}^T A\mathbf{x} + \mathbf{b}^T \mathbf{b} + \mathbf{x}^T \Gamma^T \Gamma \mathbf{x} \right]$$
$$(37)$$

$$= 2\mathbf{x} A^T A - 2A^T \mathbf{b} + 2\mathbf{x} \Gamma^T \Gamma, \quad (38)$$

since

$$\frac{\partial x A^T}{\partial x} = \frac{\partial A x^T}{\partial x}. \quad (39)$$

setting this to zero, we get

$$\mathbf{x}\left(A^T A + \Gamma^T \Gamma\right) = A^T \mathbf{b} \tag{40}$$

$$\implies \mathbf{x} = \left(A^T A + \Gamma^T \Gamma\right)^{-1} A^T \mathbf{b}, \tag{41}$$

which is what we want.

3. see figures on gitHub. I was given an optimal regularization parameter of 8.7293 with a validation set RMSE of .8340 and a test set RMSE of .8628.

4. Likewise, note that

$$||A\mathbf{x} + b\mathbf{1} - \mathbf{y}||_2^2 + ||\Gamma\mathbf{x}||_2^2 = (A\mathbf{x} + b\mathbf{1} - \mathbf{y})^T(A\mathbf{x} + b\mathbf{1} - \mathbf{y}) + (\Gamma\mathbf{x})^T(\Gamma\mathbf{x}) \tag{42}$$

$$= \mathbf{x}^T A^T A\mathbf{x} + \mathbf{x}^T A^T b\mathbf{1} - \mathbf{x}^T A^T \mathbf{y} + \mathbf{1}^T b^T A\mathbf{x} + b^2 n - \mathbf{1}^T b^T \mathbf{y} - \mathbf{y}^T A\mathbf{x} - \mathbf{y}^T b\mathbf{1} + \mathbf{y}^T \mathbf{y} \\ + \mathbf{x}^T \Gamma^T \Gamma\mathbf{x}. \tag{43}$$

Taking the gradients, we get:

$$\nabla_{\mathbf{x}}\left[||A\mathbf{x} + b\mathbf{1} - \mathbf{y}||_2^2 + ||\Gamma\mathbf{x}||_2^2\right] = 2A^T A\mathbf{x} + 2A^T b\mathbf{1} - 2A^T \mathbf{y} + 2\Gamma^T \Gamma\mathbf{x}, \tag{44}$$

and

$$\nabla_b\left[||A\mathbf{x} + b\mathbf{1} - \mathbf{y}||_2^2 + ||\Gamma\mathbf{x}||_2^2\right] = 2\mathbf{1}^T A\mathbf{x} - 2\mathbf{1}^T \mathbf{y} + 2bn. \tag{45}$$

solving for b from equation (45) (after setting it to 0), we get

$$b = \frac{\mathbf{1}^T (\mathbf{y} - A\mathbf{x})}{n} \tag{46}$$

which we then plug into equation (44) to get (after setting it equal to 0)

$$\nabla_{\mathbf{x}}\left[||A\mathbf{x} + b\mathbf{1} - \mathbf{y}||_2^2 + ||\Gamma\mathbf{x}||_2^2\right] = 2A^T A\mathbf{x} + 2A^T \left[\frac{\mathbf{1}^T (\mathbf{y} - A\mathbf{x})}{n}\right]\mathbf{1} \\ - 2A^T \mathbf{y} + 2\Gamma^T \Gamma\mathbf{x}, \tag{47}$$

which we can then solve for $\mathbf{x}$ to get:

$$\mathbf{x} = \left(A^T\left(\mathbb{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T\right)A + \Gamma^T \Gamma\right)^{-1} A^T\left(\mathbb{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T\right)\mathbf{y} \tag{48}$$

where we use $\mathbb{I}$ to denote the identity matrix.

From the code we get that the time elapsed for the training was 39.82 seconds, the difference in bias is 3.2961E-10, and the difference in weights is 5.6536E-10. Quite small.

5. see GitHub figure. I got a difference in bias that was 1.5387E-01 and a difference in weights that was 7.9992E-01. Beautiful.

$$\blacksquare$$