CSE 512: Homework 4 — Due Wednesday October 30

1. **Multiclass classification (2 pts)** Consider the multiclass logistic regression optimization problem

$$\underset{\Theta \in \mathbb{R}^{n \times K}}{\text{maximize}} \quad f(\Theta) = \frac{1}{m} \sum_{i=1}^{m} \left(\sum_{k=1}^{K} y_{ik} x_i^T \theta_k - \log \sum_{k=1}^{K} \exp(x_i^T \theta_k) \right).$$

where $y_{ik} = 1$ if data sample i is in class k, and 0 otherwise. As usual, $x_i \in \mathbb{R}^n$ is the ith data feature. Here, we write the entire matrix variable as

$$\Theta = \begin{bmatrix} \theta_1 & \theta_2 & \cdots & \theta_K \end{bmatrix}.$$

hw6_soln

(a) **(0.5 pt)** In terms of each θ_k, write the gradient of f with respect to θ_k.

(b) **(0.5 pt)** The function

$$f(\theta) = \log \left(\sum_{i=1}^{m} \exp(\theta_i) \right)$$

is sometimes called the log-sum-exp function. As we saw in lecture, it has the nice property of acting like a soft-max function, by "pulling" away the largest values of θ_i, to somewhat exaggerate their "lead".

A downside of using the log-sum-exp function is that it can have numerical issues. If θ_i is somewhat big, then $\exp(\theta_i)$ becomes very big, and can cause overflow. Conversely if θ_i is very negative, then all the values may be too close to 0 and cause underflow.

The "log-sum-exp-trick" is a numerical trick which deals with this issue, by adding and subtracting a constant whenever necessary. In effect, we simply do

$$f(\theta) = \log \underbrace{\left(\sum_{i=1}^{m} \exp(\theta_i - D) \right)}_{f_1(\theta)} + D.$$

Then, for the right choice of D, we can prevent overflow and underflow.

Propose a value of D such that $f_1(\theta) \leq c$ (preventing overflow), and another value such that $f_1(\theta) \geq c$ (preventing underflow), for some reasonably sized constant c.

(c) **Coding. (1 pt)** Run multiclass logistic regression on MNIST dataset, against each of the 10 classes.

While usually we pick a stepsize of $2/L$, I have tried this and found a larger stepsize of 10^{-5} will work well. Use this stepsize and run for 500 iterations, or however many you need to see reasonable "working" behavior. Show the train/test loss plot and the train/test classification plot.

2. **Entropy, conditional entropy, mutual information, information gain**

(2pts) Home Depot had a sale, so I went and bought a whole bunch of items: 5 hammers, 100 nails, 30 zip ties, 3 flathead screwdrivers, and two Phillips screwdrivers.

(a) **(0.5)** Recall the formula for entropy:

$$H(X) = -\sum_{X=x} \mathbf{Pr}(X = x) \log_2(\mathbf{Pr}(X = x)).$$

hw3_release.pdf
hw4_soln.pdf

Define X a random variable which represents an item picked from the pile, randomly (uniformly) picked. What is the entropy of this item? (Assume flathead $\neq$ Phillips screwdrivers.)

(b) **(0.5)** My mom comes to borrow a tool, and she contracts tetanus from a rusty nail. She yells at me "You must organize your things better!" So I pull out all the nails and zip ties and put them in the top drawer. The rest I put in the bottom drawer. Recall the formula for conditional entropy:

$$H(X|Y) = -\sum_{X=x,Y=y} \mathbf{Pr}(X = x, Y = y) \log_2(\mathbf{Pr}(X = x|Y = y)).$$

What is the conditional entropy, where X is the item randomly picked, and Y is the drawer of which I pick it from? Assume that I pick either drawer with equal probability, and given a drawer, my choice of item is uniformly distributed.

(c) **(0.5)** The *information gain* (also called *mutual information*) can be defined in terms of the entropy and conditional entropy

$$I(X;Y) = H(X) - H(X|Y).$$

Give the mutual information between X the item picked and Y the drawer which it comes from.

(d) **(0.5)** As an alternative organizational scheme, I could have put all the screwdrivers (Phillips and flathead) in top drawer, half my nails in the top drawer, and the rest in the bottom drawer. Which organizational scheme gives the best (largest) information gain?

3. **Bias-variance tradeoff. (2pts)** Suppose I want to identify the location of a star, which lives at coordinates defined by $x \in \mathbb{R}$. Every day I go to the telescope, and I receive a new measurement $y_i = x + z_i$, where $z_i \sim \mathcal{N}(0, 1)$ is the noise in each measurement.

   After m days, I receive m measurements, $y_1, ..., y_m \in \mathbb{R}$.

   (a) **(0.75 pt)** Denote x_{MLE} as the maximum likelihood estimate of x.

   hw5_solns.pdf

      i. Compute x_{MLE} in terms of y_i and m.
      ii. Compute the bias and variance of x_{MLE}.
      iii. **(0.3)** Describe the behavior of the bias and variance of x_{MLE} as $m \to +\infty$.

   (b) **(0.75 pt)** A colleague walks in the room and scoffs at my experiment. "I already know where this star is!" the colleague exclaims, and gives me a new set of measurements $\bar{x} \in \mathbb{R}^n$. "You can just cancel your experiment now!" Trouble is, I know the colleague is full of hot air, so while this is valuable information, I'm not willing to take it without any verification. Instead, I estimate x by solving a linear regression problem

$$\underset{x}{\text{minimize}} \quad \frac{1}{m}\sum_{i=1}^{m}(y_i - x)^2 + \rho(x - \bar{x})^2$$

   for some $\rho > 0$. Denote x_{MAP} as the solution to this linear regression problem. You should treat $\bar{x}$ as an external constant, which is not random, but may not be equal to x.

      i. Compute x_{MAP} in terms of y_i, m, $\bar{x}$, and ρ.
      ii. Compute the bias and variance of x_{MAP}.
      iii. Describe the behavior of the bias and variance of x_{MAP} as $m \to +\infty$.

   (c) **(0.5 pt)** A natural question to ask is, how to choose ρ? In general, we want ρ to be big when our MLE estimate is not very powerful, either because m is very small or the variance of y_i is very big. On the other hand, if our prior $\bar{x}$ is very close to x, large ρ can also help guide our guess. But while we can't in general know $\|x - \bar{x}\|$, we can try to make some statistical arguments over how ρ should depend on m.

   Recall from lecture that

$$\text{MSE} = \mathbb{E}[(x - \hat{x})^2] = B^2 + V.$$

   Show that by introducing a dummy variable $\beta = \frac{1}{1+\rho}$, that the MSE can be written as a convex function of β. Use this to find the ρ that minimizes the MSE, as a function of m and an error term $\Delta = \bar{x} - x$.

4. **Linear regression (2 pts)** Suppose that I observe some data $x_i \in \mathbb{R}^n$, $i = 1, ..., m$ and some target values $y_i \in \mathbb{R}$. I pack these into the matrix and vector

$$X = \begin{bmatrix} x_1^T \\ x_2^T \\ \vdots \\ x_m^T \end{bmatrix}, \qquad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix}.$$

I try to find a linear model fit by solving the optimization problem

$$\underset{\theta}{\text{minimize}} \quad \|X\theta - y\|_2^2 + \rho\|\theta\|_2^2.$$

Assume that X^TX is invertible, and additionally, define $\lambda > 0$ its smallest eigenvalue. For this question, we will only accept answers with justification. No guessing.

(a) **(0.25 pts)** In terms of X, y, and ρ, write a closed-form expression for the minimizer θ.

(b) **(0.25 pts)** Now suppose that I found out that each label y_i was actually *randomly generated*, e.g.

$$y_i = x_i^T\bar{\theta} + z_i, \quad z_i \sim \mathcal{N}(0, 1)$$

for some ground truth $\bar{\theta} \in \mathbb{R}^n$. Now, the solution θ as written in part (a) is a random variable. In terms of $\bar{\theta}$, X and ρ, what is the expectation (mean) of θ?

(c) **(0.5 pts)** Is the θ computed in part (a) a biased estimate of the ground truth $\bar{\theta}$, in terms of this probabilistic model? If so, what is the bias? (This answer should be a vector.)

(d) **(0.5 pts)** In terms of $\bar{\theta}$, X and ρ, what is Σ the covariance matrix of θ?

(e) **(0.5 pts)** What is the trend of the bias and variance in terms of m and ρ? That is, if I write "$\|\text{bias}\|_2 = O(g(m, \rho))$" and $\|\Sigma\|_2 = O(h(m, \rho))$", what are the functions g and h?

5. **Overfitting SVMs. (2 pts)** Before the world of deep neural nets, the Arnold Schwarzenegger of the machine learning world was actually kernel SVMs. These things are big, bulky, and will fit anything you want–for a price. In this problem you will play around with your own encoded kernel SVM, around a seemingly easy dataset, and see what troubles may arise.

hw2_soln.pdf
4.py

(a) Recall that the soft primal kernel SVM problem proposes to solve

$$\begin{array}{ll} \underset{\theta\in\mathbb{R}^n, s\in\mathbb{R}^m}{\text{minimize}} & \frac{1}{2}\|\theta\|_2^2 + \rho\sum_{i=1}^m s_i \\ \text{subject to} & y_i\phi(x_i)^T\theta \geq 1 - s_i, \qquad i = 1, ..., m \\ & s_i \geq 0, \qquad i = 1, ..., m \end{array} . \tag{p-KSVM}$$

We will first derive the dual, using a few steps.

i. **(0.125 pts)** Write down the unconstrained Lagrangian penalty formulation of (p-KSVM).

ii. **(0.125 pt)** The Lagrange dual of (p-KSVM) is the unconstrained minimization of $\mathcal{L}$ over the primal variables, where the dual variables are properly constrained. Derive the dual of (p-KSVM). (Note that at this point, your answer should still contain ϕ somewhere.) Make sure to eliminate at least one of the dual vector variables, so that there is only one dual vector variable left.

iii. **(0.25 pts)** Now we use the simplification $K(X, Y) = \phi(X)\phi(Y)^T$, and in particular, assume that we have access to a matrix $K \in \mathbb{R}^{m\times m}$ where $K_{i,j} = \phi(x_i)^T\phi(x_j)$. Rewrite your dual formulation such that it is in terms of the matrix K, and remove all instances of ϕ.

iv. **(0.25 pts)** Write down the gradient of the objective function in your final proposed formulation.

(b) **(0.25 pts)** Before experimenting with fun kernels, open the accompanying `kernel_svms.ipynb` file. The notoriously challenging half-moon dataset should be available. Load it.

First, we will solve this using the linear kernel. Use the provided kernel function for linear kernels and code up a simple projected gradient descent method, e.g.

$$u^{(k+1)} = \max\{0, \min\{\rho, u^{(k)} - \alpha\nabla f(u^{(k)})\}\}$$

Pick an appropriate step size α and number of iterations through experimentation. Give the resulting train and test errors for the following choices of ρ: $\rho = 0.01, 0.1, 1, 10$.

(c) **(0.25 pts)** Pick the best choice of hyperparameters (α, ρ, and maximum iterations) and plot a contour plot showing the learned landscape of the model. (Use the provided code.)

(d) **(0.25 pts)** Next, code up an RBF Kernel, e.g. a function

$$K(x_1, x_2) = \exp\left(-\frac{\|x_1 - x_2\|_2^2}{2\sigma^2}\right)$$

Sweep σ across values $\sigma = 1, 0.1, 0.001, 0.0001$, and for each value, pick the best hyperparameters (α, ρ, and maximum iterations), and return the train and test errors of the learned RBF model for each value of σ.

(e) **(0.25 pts)** Plot the contour plots of the best-learned RBF model under the choices of σ: $\sigma = 1, 0.1, 0.001, 0.0001$.

(f) **(0.25 pts)** In your own words, describe what is happening to the model as $\sigma \to 0$.