Learn2Launch: Artificial Intelligence Homework
Spring 2019

March 14, 2019

Names: _____

email: _____

Question 1. Given $1 \leq d \leq n$, a matrix $P \in \mathbb{R}^{n \times n}$ is said to be a rank-d orthogonal projection matrix if $\text{rank}(d) = P$, $P = P^\top$ and $P^2 = P$.

(a) Prove that P is a rank-d projection matrix if and only if there exists a $U \in \mathbb{R}^{n \times d}$ such that $P = UU^\top$ and $U^\top U = I$.

(b) Prove that if P is a rank d projection matrix, then $\text{tr}(P) = d$.

(c) Prove that, for all $v \in \mathbb{R}^n$,

$$Pv = \arg \min_{w \in \text{range}(P)} \|v - w\|_2^2.$$

(d) Prove that if $X \in \mathbb{R}^{d \times d}$ and $\text{rank}(X) = d$, then $X(X^\top X)^{-1} X^\top$ is a rank-d orthogonal projection matrix. What is the corresponding matrix U?

(e) Let $y = X\theta_* + z$, where $X \in \mathbb{R}^{n \times d}$, $\theta_* \in \mathbb{R}^d$, $y \in \mathbb{R}^n$, and $z = \mathcal{N}(0, I) \in \mathbb{R}^n$, and suppose $\text{rank}(X) = d$. Prove that if $\widehat{\theta} = \arg\min_\theta \|X\theta - y\|_2^2$, then

$$\mathbb{E}[\|\theta_* - \widehat{\theta}\|_2^2] = \text{tr}((X^\top X)^{-1})$$

(f) In the setting of the Part (e), show that

$$\frac{1}{n} \mathbb{E}[\|X(\theta_* - \widehat{\theta})\|_2^2] = \frac{d}{n}.$$

How does the answer change if $\text{rank}(X) < d$?

(g) Let

$$\widehat{\theta}_\lambda = \arg\min_\theta \|X\theta - y\|_2^2 + \lambda\|\theta\|_2^2,$$

for $\lambda > 0$, be the solution to the ridge regression problem. Suppose X has the singular value decomposition $U\Sigma V^\top$, where $\Sigma = \text{diag}(s_1, \cdots, s_d)$, $s_i \geq 0$. Using part b), show that $\widehat{\theta}_\lambda = VDU^\top y$, where D is a diagonal matrix to be determined.

(h) Let A be a $d \times n$ matrix and B be a $n \times d$ matrix. For any $\mu > 0$, show that $(AB + \mu I)^{-1}A = A(BA + \mu I)^{-1}$, if $AB + \mu I$ and $BA + \mu I$ are invertible. Why is this fact useful when fitting linear regression models? How is it related to the kernel trick?

Question 2. We consider a classification problem where we want to classify data points from $\mathbb{R}^d$ into *two* classes. In this problem we explore whether linear regression can solve a classification tasks of this form based on the MNIST data set. The MNIST dataset contains labelled images of handwritten digits. Each image has a label in $\{0, \ldots, 9\}$. We only keep the digits labelled 0 or 1.

The provided data has already been flattened. Each row in train_features.npy and test_features.npy contains the pixel values for one image. Each column represents a pixel position in the images.

(a) We now want to use linear regression for the problem, treating class labels as real values $y = -1$ for class "zero" and $y = 1$ for class "one". Solve the corresponding optimization problem $\min_\theta \|X\theta - y\|_2^2$ where the entry y_i is the value of the class (-1 or 1) corresponding to the image in row i of the feature matrix X. Report the values of $\|X\theta - y\|_2^2$ and θ. Use numpy to solve the optimization problem, don't use machine learning libraries.

(b) Given a new flattened image x, one natural rule to classify it is this: output 0 if $x^\top\theta \leq 0$ and output 1 if $x^\top\theta > 0$. Report what percentage of the digits in the training set are correctly classified by this rule. Report what percentage of the digits in the test set are correctly classified by this rule.

(c) Raw pixel values are generally not a good idea (even if here they yield a good performance). Let's try random features as well. For each row x in X we construct the random features

$$\phi(x) = \cos(Gx + b)\,,$$

where each entry of $G \in \mathbb{R}^{p\times d}$ is drawn i.i.d. as $\mathcal{N}(0, 0.01)$ and each entry in the vector $b \in \mathbb{R}^p$ is drawn i.i.d from the uniform distribution on $[0, 2\pi]$. Here, p denotes the number of random features and d denotes the number of pixels in each image. Note that $\phi(x)$ is a p dimensional vector because the cosine function is taken. What performance do you get if $p = 5000$. Note that G and b are sampled in the beginning and they are fixed throughout; so the same G and b are used for at test time.

(d) Another good exercise is to implement an SVM for this problem. Namely, find θ and a bias term b that minimize the loss function

$$\sum_{i=1}^{n} \max\{1 - y_i(\theta^\top x_i + b), 0\} + \frac{\lambda}{2}\|\theta\|_2^2.$$

You should use SGD to optimize θ and b. Once the optimization is complete what prediction would you make for a new image x? Evaluate your model on the test set.