**Math 7243 Machine Learning - Fall 2021**

**Instructor: He Wang**

**Test 1.**

**Student Name:** _____ _____**/50**

**Rules and Instructions for Exams:**

1. Unless otherwise specified, to receive full credits you must show **all** necessary work. The grading is based on your work shown. Only a final result from computer will receive zero point.

2. You need to finish the exam yourself. Any discussions with the other people will be considered as **academic dishonesty**. **Cheating, Unauthorized Collaboration, and Facilitating Academic Dishonesty are not allowed.** You can read a description of each here http://www.northeastern.edu/osccr/academic-integrity-policy/

3. This is an open exam. You are allowed to look at textbooks, and use a computer.

4. You are **not** allowed to discuss with any other people.

5. You are **not** allowed to ask questions on any internet platform.

6. For programming questions, if there is no specific instruction, you can only use numpy library. You should **not** use any build in function from Scikit-learn or StatsModels libraries.

**1.** (10 points) Calculate the **gradient** and **Hessian matrix** of the following functions and find the **argmin**$_\theta$ of each function. Here the norm $||\ \ ||$ is the standard $l_2$-norm. You can use any results in the lecture notes.

(1) Let $\vec{b} \in \mathbb{R}^d$ and let $J(\vec{\theta}) = ||\vec{\theta} - \vec{b}||^2$.

(2) Let $X \in \mathbb{R}^{n \times d}$ and $\vec{b} \in \mathbb{R}^d$. Suppose $\text{rank}(X) = d$. Let $F(\vec{\theta}) = ||X\vec{\theta}||^2 + \vec{\theta}^T\vec{b}$.

**2.** (10 points) In this question, you may use Python (with only numpy library) to solve the matrix equation. Consider the following data points

| $x_1$ | $x_2$ | $y$ |
|---|---|---|
| 1.1 | 2 | 2.3 |
| 2.2 | 4 | 4.3 |
| 3.1 | 6 | 6.3 |
| 4.2 | 8 | 7.8 |
| 5.3 | 10 | 9.8 |

(1). Fit a linear model $y = \theta_0 + \theta_1 x_1 + \theta_2 x_2$ to this dataset when the loss is RSS= $||X\vec{\theta} - \vec{y}||^2$. You should report the best fit function and the RSS cost value.

(2). Fit a linear function to this dataset when the loss is the Ridge Loss $J(\theta) = ||X\vec{\theta} - \vec{y}||^2 + \lambda(\theta_1^2 + \theta_2^2)$ with $\lambda = 1$ and with $\lambda = 10$. You should report the best fit function and the **RSS** cost value. (Warning: Do not put penalty on $\theta_0$)

**3.** (10 points) Consider the data

| $x^{(i)}$ | 0 | 0.2 | 0.4 | 0.6 | 0.8 | 1 | 1.2 | 1.4 |
|---|---|---|---|---|---|---|---|---|
| $y^{(i)}$ | 5.1 | 6.4 | 6.1 | 8.2 | 9.5 | 8.6 | 12 | 14.8 |

The data file $\{\vec{x}^{(i)}, y^{(i)}\}$ for $i = 1, 2, ..., n = 8$ is drawn (with noise) from

$$f(x) = \theta_0 + \theta_1 e^x$$

(1) Find a **closed formula** for parameters $\vec{\theta}$ to minimize the RSS loss

$$J(\vec{\theta}) = \sum_{i=1}^{n} \left(y^{(i)} - f(x^{(i)})\right)^2$$

(2) **Find the function** $f(x)$ fitting the data using the result in (1).

**4.** (10 points) Consider the data

| $x^{(i)}$ | 0 | 0.2 | 0.4 | 0.6 | 0.8 | 1 | 1.2 | 1.4 |
|---|---|---|---|---|---|---|---|---|
| $y^{(i)}$ | 3.2 | 4.2 | 5.6 | 5.2 | 7.7 | 8.8 | 13.9 | 18.7 |

The data file $\{\vec{x}^{(i)}, y^{(i)}\}$ for $i = 1, 2, ..., n = 8$ is drawn (with noise) from the function:

$$g(x) = \theta_0 + e^{\theta_1 x}.$$

Fit the data to the function $g(x)$ by minimizing the RSS loss

$$J(\vec{\theta}) = \sum_{i=1}^{n} \left(y^{(i)} - g(x^{(i)})\right)^2.$$

(1) Find the **gradient** of the cost function $J(\vec{\theta})$.

(2) Write down the update formula for gradient decent using $\alpha$ for the learning rate.

(3) Use gradient decent(GD) to find $\theta_*$ to minimize $J(\vec{\theta})$. You should try different learning rates and recording the cost function values to see what is the best $\alpha$. Turn in any associated computations, your learning rate, cost values, and the parameters.

**5.** Consider the categorical learning problem consisting of a data set with two labels:

**Label 1:** (contains 6 points)

| | | | | | | |
|---|---|---|---|---|---|---|
| $X_1$ | 0.2 | 0.6 | 2 | 2.6 | 3.1 | 3.8 |
| $X_2$ | 3.4 | 1.8 | 2 | 2.7 | 3.5 | 1.5 |

**Label 2:** (contains 5 points)

| | | | | | |
|---|---|---|---|---|---|
| $X_1$ | -0.7 | -2.1 | -2.5 | -3 | -3.9 |
| $X_2$ | -2.9 | -2.8 | -1.3 | -2 | -1.5 |

**(1)** (7 points) For each label above, the data follow a multivariate normal distribution Normal$(\mu_i, \Sigma)$ where the covariance $\Sigma$ is the same for both labels. Fit a pair of LDA functions to the labels by computing the covariances $\Sigma$, means $\mu_i$, and proportion $\phi$ of data. You may use Python (with only numpy library)

(a) You should report the values for $\phi$, $\mu_i$ and $\Sigma$.

(b) Give the **formula for the line** forming the decision boundary.

**(2)** (3 points) For each label above, use **logistic regression** to classify the data. You should report the **logistic function** $p(y = 1 \mid \vec{x}) = \dfrac{1}{1 + e^{-\theta^T \vec{x}}}$ and the **formula for the line** forming the decision boundary. (In this question, you can use any Python library including Scikit-learn.)

**5.** (**continue**)

**(3)**(2 **bonus** points) Find the probability $P(y = 1|\vec{x})$ for a test point $\vec{x} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ for both LDA model and the logistics model in the above two questions.

**(4)** (2 **bonus** points) Find the boundary using the QDA method. (You may use a computer, but only with numpy library)