Assignment 2

COGS 118A: Supervised Machine Learning Algorithms

Due: October 22, 2020, 11:59pm (Pacific Time).

Instructions: Answer the questions below, attach your code, and insert figures to create a PDF file; submit your file via Gradescope. You may look up the information on the Internet, but you must write the final homework solutions by yourself.

Late Policy: 5% of the total points will be deducted on the first day past due. Every 10% of the total points will be deducted for every extra day past due.

Grade: ____ out of 100 points

1 (10 points) Conceptual Questions

(1.1) Is the following statement true or false?

$f(x)$ is linear with respect to x, given $f(x) = w_0 + w_1 x + w_2 x^2$ where $x, w_0, w_1, w_2 \in \mathbb{R}$.

[True] [False]

(1.2) "One-hot encoding" is a standard technique that turns categorical features into general real numbers. If we have a dataset S containing m data points where each data point has 1 categorical feature. Specifically, this categorical feature has k possible categories. Thus, the shape of the one-hot encoding matrix that represents the dataset S is:

A. $k \times k$
B. $1 \times k$
C. $m \times k$
D. $m \times m$

(1.3) Assume we have a binary classification model:

$$f(\mathbf{x}) = \begin{cases} +1, & \mathbf{w} \cdot \mathbf{x} + b \geq 0, \\ -1, & \mathbf{w} \cdot \mathbf{x} + b < 0 \end{cases}$$

where the feature vector $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$, bias $b \in \mathbb{R}$, weight vector $\mathbf{w} = (w_1, w_2) \in \mathbb{R}^2$. The decision boundary of the classification model is:

$$\mathbf{w} \cdot \mathbf{x} + b = 0$$

(a) If the predictions of the classifier f and its decision boundary $\mathbf{w} \cdot \mathbf{x} + b = 0$ are shown in Figure 1, which one below can be a possible solution of weight vector $\mathbf{w}$ and bias b?

A. $\mathbf{w} = (+1, 0), b = -1$.
B. $\mathbf{w} = (-1, 0), b = +1$.
C. $\mathbf{w} = (+1, 0), b = +1$.
D. $\mathbf{w} = (0, -1), b = -1$.

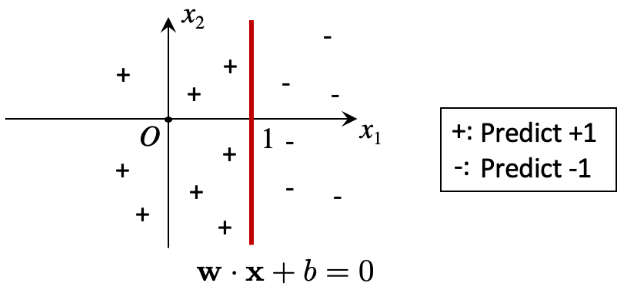

Figure 1: Decision Boundary 1

(b) If the predictions of the classifier f and its decision boundary $\mathbf{w} \cdot \mathbf{x} + b = 0$ are shown in Figure 2, which one below can be a possible solution of weight vector $\mathbf{w}$ and bias b?

A. $\mathbf{w} = (+1, 0), b = -1$.
B. $\mathbf{w} = (-1, 0), b = +1$.
C. $\mathbf{w} = (+1, 0), b = +1$.
D. $\mathbf{w} = (0, -1), b = -1$.

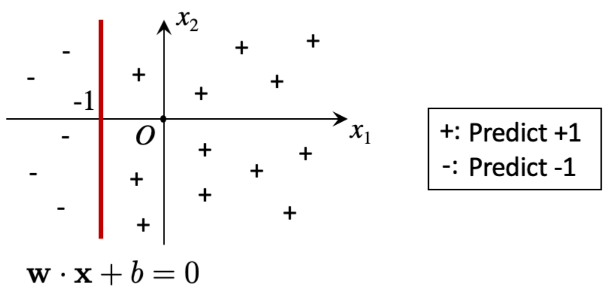

Figure 2: Decision Boundary 2

(1.4) Choose the **most** significant difference between **regression** and **classification**:

A. unsupervised learning vs. supervised learning.

B. prediction of continuous values vs. prediction of class labels.

C. features are not one-hot encoded vs features are one-hot encoded.

D. none of the above.

2 (25 points) Decision Boundary

2.1 (3 points)

We are given a classifier that performs classification in $\mathbb{R}^2$ (the space of data points with 2 features (x_1, x_2)) with the following decision rule:

$$h(x_1, x_2) = \begin{cases} 1, & \text{if} \quad 2x_1 + 4x_2 - 8 \geq 0 \\ 0, & \text{otherwise.} \end{cases}$$

Draw the decision boundary of the classifier and shade the region where the classifier predicts 1. Make sure you have marked the x_1 and x_2 axes and the intercepts on those axes.

2.2 (9 points)

We are given a classifier that performs classification on $\mathbb{R}^2$ (the space of data points with 2 features (x_1, x_2)) with the following decision rule:

$$h(x_1, x_2) = \begin{cases} 1, & \text{if} \quad w_1x_1 + w_2x_2 + b \geq 0 \\ 0, & \text{otherwise.} \end{cases}$$

Here, the normal vector $\mathbf{w}$ of the decision boundary is normalized, i.e.:

$$||\mathbf{w}||_2 = \sqrt{w_1^2 + w_2^2} = 1.$$

1. Compute the parameters w_1, w_2 and b for the decision boundary in Figure 3. Please make sure the predictions from the obtained classifier are consistent with Figure 3.

   Hint: Please use the intercepts in the Figure 3 to find the relation between w_1, w_2 and b. Then, substitute it into the normalization constraint to solve for parameters.

2. Use parameters from the above question to compute predictions for the following two data points: $A = (3, 6)$, $B = (1, -4)$.

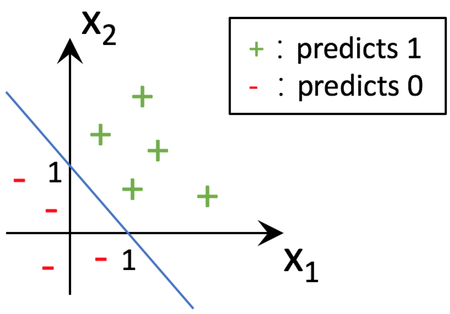

Figure 3: Decision boundary to solve for parameters.

2.3 (10 points)

We are given a classifier that performs classification on $\mathbb{R}^3$ (the space of data points with 3 features (x_1, x_2, x_3)) with the following decision rule:

$$h(x_1, x_2, x_3) = \begin{cases} 1, & \text{if} \quad w_1x_1 + w_2x_2 + w_3x_3 + b \geq 0 \\ 0, & \text{otherwise.} \end{cases}$$

Here, the normal vector $\mathbf{w}$ of the decision boundary is normalized, i.e.:

$$||\mathbf{w}||_2 = \sqrt{w_1^2 + w_2^2 + w_3^2} = 1.$$

In addition, we set $b \leq 0$ to have an unique equation for the decision boundary.

1. Compute the parameters w_1, w_2, w_3 and b for the decision boundary that passes through three points $A = (3, 2, 4)$, $B = (-1, 0, 2)$, $C = (4, 1, 5)$ in Figure 4.

   Hint: Please use the intercepts in the Figure 4 to find the relation between w_1, w_2, w_3 and b. Then, substitute it into the normalization constraint to solve for parameters.

2. Use parameters from the above question to compute predictions for the following two data points: $D = (0, 0, 0)$, $E = (1, 0, 5)$.

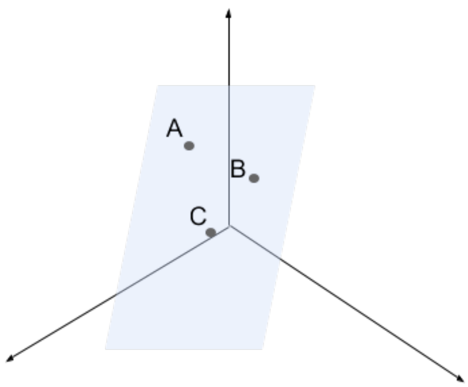

Figure 4: Decision boundary to solve the parameters.

2.4 (3 points)

We are given a classifier that performs classification in $\mathbb{R}^2$ (the space of data points with 2 features (x_1, x_2)) with the following decision rule:

$$h(x_1, x_2) = \begin{cases} 1, & \text{if} \quad x_1^2 + x_2^2 - 10 \geq 0 \\ 0, & \text{otherwise.} \end{cases}$$

Draw the decision boundary of the classifier and shade the region where the classifier predicts 1. Make sure you have marked the x_1 and x_2 axes and the intercepts on those axes.

3 (10 points) Derivatives

3.1 Function Defined by Scalars

1. (3 points) Given a function $f(w) = (y_1 + wx_1)^2$ where $(x_1, y_1) = (3, 4)$ represents a data point, derive $\dfrac{\partial f(w)}{\partial w}$.

2. (3 points) Given a function $f(w) = \sum_{i \in \{1,2\}} (y_i - wx_i)^2$ where $(x_1, y_1) = (1, 1), (x_2, y_2) = (2, 3)$ are two data points, derive $\dfrac{\partial f(w)}{\partial w}$.

3.2 Function Defined by Vectors

1. (4 points) Given a function $f(w) = (\mathbf{y} - w\mathbf{x})^T(\mathbf{y} - w\mathbf{x})$ where $\mathbf{x} = [1, 2]^T$ and $\mathbf{y} = [1, 3]^T$, derive $\dfrac{\partial f(w)}{\partial w}$.

   Note: In $f(w)$, $w \in \mathbb{R}$ is still a scalar.

4 (9 points) Concepts

Select the correct option(s). Note that there might be multiple correct options.

1. For two monotonically increasing functions $f(x)$ and $g(x)$:

   A. $f(x) + g(x)$ is always monotonically increasing.
   B. $f(x) - g(x)$ is always monotonically increasing.
   C. $f(x^2)$ is always monotonically increasing.
   D. $f(x^3)$ is always monotonically increasing.

2. For a function $f(x) = x(10 - x), x \in \mathbb{R}$, please choose the correct statement(s) below:

   A. $\arg\max_x f(x) = 5$.
   B. $\arg\min_x f(x) = 25$.
   C. $\min_x f(x) = 5$.
   D. $\max_x f(x) = 25$.

3. Assume we have a function $f(x)$ which is differentiable at every $x \in \mathbb{R}$. There are three properties that describe the function $f(x)$:
   (1) $f(x)$ is a convex function.
   (2) When $x = x_0$, $f'(x_0) = 0$.
   (3) $f(x_0)$ is a global minimum of $f(x)$.

   Which one of the following statements is **wrong**?
   Hint: You can use a failure case to disprove a statement.

   A. Given (1) and (2), we can prove that (3) holds.
   B. Given (2) and (3), we can prove that (1) holds.
   C. Given (1) and (3), we can prove that (2) holds.

5 (4 points) Argmin and Argmax

An unknown estimator is given an estimation problem to find the minimizer and maximizer of the objective function $G(w) \in (0, 2]$:

$$(w_a, w_b) = (\arg\min_w G(w), \arg\max_w G(w)). \tag{1}$$

The solution to Eq. 1 by the estimator is $(w_a, w_b) = (10, 20)$.

Given this information, please obtain the value of w^* such that:

$$w^* = \arg\min_w [10 - 4 \times \ln(G(w))]. \tag{2}$$

6 (12 points) Data Manipulation

In this question, we still use the Iris dataset from Homework 1. In fact, you can see the shape of array X is $(150, 4)$ by running `X.shape`, which means it contains 150 data points where each has 4 features. Here, we will perform some basic data manipulation and calculate some statistics:

1. Divide array X evenly to five subsets of data points:

   Group 1: 1st to 30th data point,

   Group 2: 31st to 60th data point,

   Group 3: 61st to 90th data point,

   Group 4: 91st to 120th data point,

   Group 5: 121st to 150th data point.

   Then calculate the mean of feature vectors in each group. Your results should be five 4-dimensional vectors (i.e. shape of `NumPy` array can be `(4,1)`, `(1,4)` or `(4,)`).

2. Remove 2nd and 3rd features from array X, resulting a 150×2 matrix. Then calculate the mean of all feature vectors. Your result should be a 2-dimensional vector.

3. Remove last 10 data points from array X, resulting a 140×4 matrix. Then calculate the mean of feature vectors. Your result should be a 4-dimensional vector.

7 (15 points) Training vs. Testing Errors

In this problem, we are given two trained predictive models on a modified Iris dataset. Each data point $(\mathbf{x}, y)$ has a feature vector $\mathbf{x}_i \in \mathbb{R}^4$ and its corresponding label $y_i \in \{0, 1\}$, where $i \in \{1, 2, \ldots, 150\}$. To predict on the new data, here we consider two types of model: a regression model and a classification model. The regression model is trained to predict a real number, while the classification model applies a threshold to the output of the regression model, converting the real number into a binary value.

The regression model is as followed:

$$\hat{y}_i(\mathbf{x}_i) = \mathbf{w}^T \mathbf{x}_i + b$$

The classifier is as followed:

$$h(\mathbf{x}_i) = \begin{cases} 1, & \text{if} \quad \hat{y}_i(\mathbf{x}_i) \geq 1/2 \\ 0, & \text{otherwise.} \end{cases}$$

where $\mathbf{w} = [0.1297, 0.1225, -0.1171, 0.6710]^T, b = -1.1699$.

The regression error is defined as:

$$\sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2}$$

and the classification error is defined as:

$$\frac{1}{n} \sum_{i=1}^{n} \mathbf{1}(h(\mathbf{x}_i) \neq y_i)$$

where n is the number of data points.

The data as well as the split of training and testing set are given in the Jupyter notebook we provided. **You should not use the `scikit-learn` library**.

Please download the notebook `training_test_errors.ipynb` from the course website and fill in the missing blanks. Follow the instructions in the skeleton code and report:

- Training error of the regression model.
- Testing error of the regression model.
- Training error of the classification model.
- Testing error of the classification model.

8 (15 points) Linear Regression

Assume we are given a dataset $S = \{(x_i, y_i), i = 1, \dots, n\}$. Here, $x_i \in \mathbb{R}$ is a feature scalar (a.k.a. value of input variable) and $y_i \in \mathbb{R}$ is its corresponding value (a.k.a. value of dependent variable). In this section, we aim to fit data points with a line:

$$y = w_0 + w_1 x \tag{3}$$

where $w_0, w_1 \in \mathbb{R}$ are two parameters to determine the line. Next, we measure the quality of fitting by evaluating a sum-of-squares error function $g(w_0, w_1)$:

$$g(w_0, w_1) = \sum_{i=1}^{n} (w_0 + w_1 x_i - y_i)^2 \tag{4}$$

When $g(w_0, w_1)$ is near zero, it means the proposed line can fit the dataset and model an accurate relation between x_i and y_i. The best line with parameters (w_0^*, w_1^*) can reach the minimum value of the error function $g(w_0, w_1)$:

$$(w_0^*, w_1^*) = \arg\min_{w_0, w_1} g(w_0, w_1) \tag{5}$$

To obtain the parameters of the best line, we will take the gradient of function $g(w_0, w_1)$ and set it to zero. That is:

$$\nabla g(w_0, w_1) = \mathbf{0} \tag{6}$$

The solution (w_0^*, w_1^*) of the above equation will determine the best line $y = w_0^* + w_1^* x$ that fits the dataset S.

In reality, we typically tackle this task in a matrix form: First, we represent data points as matrices $X = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]^T$ and $Y = [y_1, y_2, \dots, y_n]^T$, where $\mathbf{x}_i = [1, x_i]^T$ is a feature vector corresponding to x_i. The parameters of the line are also represented as a matrix $W = [w_0, w_1]^T$. Thus, the sum-of-squares error function $g(W)$ can be defined as (a.k.a. squared L_2 norm):

$$\begin{align} g(W) &= \sum_{i=1}^{n} (\mathbf{x}_i^T W - y_i)^2 \tag{7} \\ &= \|XW - Y\|_2^2 \tag{8} \\ &= (XW - Y)^T (XW - Y) \tag{9} \end{align}$$

Similarly, the parameters $W^* = [w_0^*, w_1^*]^T$ of the best line can be obtained by solving the equation below:

$$\nabla g(W) = \frac{\partial g(W)}{\partial W} = \mathbf{0} \tag{10}$$

(a) According to Eq. 8 and 9, compute the gradient of $g(W)$ with respect to W. Your result should be in the form of X, Y and W.

(b) By setting the answer of part (a) to $\mathbf{0}$, prove the following:

$$W^* = \arg\min_W g(W) = (X^T X)^{-1} X^T Y \tag{11}$$

Note: The above formula demonstrates a closed form solution of Eq. 10.

Previously, we define a sum-of-squares error function $g(w_0, w_1) = \sum_{i=1}^{n}(y_i - w_0 - w_1 x_1)^2$ and represent it in a matrix form $g(W) = \|XW - Y\|_2^2$. Actually, we can have multiple choices of the error function: For example, we can define a sum-of-absolute error function $h(w_0, w_1)$:

$$h(w_0, w_1) = \sum_{i=1}^{n} |w_0 + w_1 x_i - y_i| \tag{12}$$

and represent it in a matrix form $h(W)$ (a.k.a. L_1 norm):

$$\begin{aligned} h(W) &= \sum_{i=1}^{n} |\mathbf{x}_i^T W - y_i| && (13) \\ &= \|XW - Y\|_1 && (14) \end{aligned}$$

(c) According to the Eq. 13, compute the gradient of the error function $h(W)$ with respect to W. Your result should be in the form of $\mathbf{x}_i$, y_i and W.

Hint: Given a function $f(\mathbf{x}) \in \mathbb{R}$, we have:

$$\frac{\partial |f(\mathbf{x})|}{\partial \mathbf{x}} = \text{sign}(f(\mathbf{x})) \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}}$$

where

$$\text{sign}(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0. \end{cases}$$

While we can represent the problem as in (c), where the gradient is calculated for each $\mathbf{x}_i$ input, it can be very useful to calculate the gradient over an entire dataset. While there's not a problem here for you to solve for points, we wanted you to know that problem (c) can be re-written in terms of the entire training set X as:

$$\nabla h(W) = \frac{\partial h(W)}{\partial W} = \Big(\big(\text{sign}(XW - Y)\big)^\top X \Big)^\top \tag{15}$$

where $\text{sign}(A)$ means performing element-wise $\text{sign}(a_{ij})$ over all elements a_{ij} in a matrix A. This matrix form of the gradient will be useful in next week's homework.