Homework Assignment 1

CSE 251A: ML - Learning Algorithms

Due: April 11th, 2023, 9:30am (Pacific Time)

Instructions: Please answer the questions below, attach your code in the document, and insert figures to create **a single PDF file**. You may search information online but you will need to write code/find solutions to answer the questions yourself.

Grade: ____ out of 100 points

1 (10 points) Classification vs. Clustering

In this question, you are provided with several scenarios. You need to identify if the given scenario is better formulated as a *classification* task or a *clustering* task. You should also provide the reason that supports your choice.

1. Scenario 1: Assume there are 100 graded answer sheets for a homework assignment (scores range from 0 to 100). We would like to split them into several groups where each group has similar scores.

   Choice: ______________ task

   Reason:

2. Scenario 2: Assume there are 100 graded answer sheets for a homework assignment (scores range from 0 to 100). We would like to split them into several groups where each group represents a letter grade (A, B, C, D) following the criteria: A (90-100), B (75-90), C (60-75), D (0-60).

   Choice: ______________ task

   Reason:

2 (40 points) Basic Calculus

2.1 (20 points) Derivatives with Scalars

1. $f(x) = \frac{1}{2}(ax - b)^2$ where $a, b \in \mathbb{R}$ are constant scalars, derive $\dfrac{\partial f(x)}{\partial x}$.

2. $f(x) = \ln(1 + e^x)$, derive $\dfrac{\partial f(x)}{\partial x}$.

2.2 (20 points) Derivatives with Vectors

Several particular vector derivatives are useful for this course. For matrix $\mathbf{A} \in \mathbb{R}^{M\times M}$, column vector $\mathbf{x} \in \mathbb{R}^M$ and $\mathbf{a} \in \mathbb{R}^M$, we have

- $\dfrac{\partial \mathbf{a}^\top \mathbf{x}}{\partial \mathbf{x}} = \dfrac{\partial \mathbf{x}^\top \mathbf{a}}{\partial \mathbf{x}} = \mathbf{a}$,
- $\dfrac{\partial \mathbf{x}^\top \mathbf{A}\mathbf{x}}{\partial \mathbf{x}} = (\mathbf{A} + \mathbf{A}^\top)\mathbf{x}$. If $\mathbf{A}$ is symmetric, $\dfrac{\partial \mathbf{x}^\top \mathbf{A}\mathbf{x}}{\partial \mathbf{x}} = 2\mathbf{A}\mathbf{x}$.
  A special case is, if $\mathbf{A} = \mathbf{I}$ (identity matrix), $\dfrac{\partial \mathbf{x}^\top \mathbf{x}}{\partial \mathbf{x}} = \dfrac{\partial \mathbf{x}^\top \mathbf{I}\mathbf{x}}{\partial \mathbf{x}} = 2\mathbf{I}\mathbf{x} = 2\mathbf{x}$.

The above rules adopt a *denominator-layout* notation. For more rules, you can refer to this Wikipedia page. Please apply the above rules and calculate following derivatives:

1. $f(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{a})^\top(\mathbf{x} - \mathbf{a})$ where $\mathbf{a} \in \mathbb{R}^M$ is a constant vector, derive $\dfrac{\partial f(\mathbf{x})}{\partial \mathbf{x}}$.

2. $f(\mathbf{x}) = \frac{1}{2}(\mathbf{A}\mathbf{x} - \mathbf{b})^\top(\mathbf{A}\mathbf{x} - \mathbf{b})$ where $\mathbf{A} \in \mathbb{R}^{M\times M}$ is a constant matrix and $\mathbf{b} \in \mathbb{R}^M$ is a constant vector, derive $\dfrac{\partial f(\mathbf{x})}{\partial \mathbf{x}}$.

   Hint: Note that $(\mathbf{A}^\top\mathbf{A})^\top = \mathbf{A}^\top\mathbf{A}$, thus $\mathbf{A}^\top\mathbf{A}$ is a symmetric matrix.

3 (20 points) Metrics

In machine learning, we have many metrics to evaluate the performance of our model. For example, in a binary classification task, there is a dataset $S = \{(\mathbf{x}_i, y_i), i = 1, .., N\}$ where each data point $(\mathbf{x}, y)$ contains a feature vector $\mathbf{x} \in \mathbb{R}^M$ and a ground-truth label $y \in \{0, 1\}$. We have obtained a classifier $f : \mathbb{R}^M \to \{0, 1\}$ to predict the label $\hat{y}$ of feature vector $\mathbf{x}$:

$$\hat{y} = f(\mathbf{x})$$

Assume $N = 200$ and we have the following *confusion matrix* to represent the result of classifier f on dataset S:

	Actual Positives ($y = 1$)	Actual Negatives ($y = 0$)
Predicted Positives ($\hat{y} = 1$)	5	5
Predicted Negatives ($\hat{y} = 0$)	10	180

Please follow the lecture notes to compute the metrics below:

1. Please compute the *accuracy* of the classifier f on dataset S.

2. Please compute the *precision* of the classifier f on dataset S.

3. Please compute the *F1 score* of the classifier f on dataset S.

4. You may find the accuracy of current model very high. Does it mean the performance of this model is always very good? Why?

   Hint: You may refer to other metrics you have computed.

4 (10 points) Loss functions

1. Prove the Rooted Mean Square Error (RMSE) is always greater than or equal to the Mean Absolute Error (MAE).

   Hint: You may use Cauchy-Schwarz inequality.

2. Another commonly used loss function is called Max Error (ME), which is the maximum absolute difference between two vectors. Suppose we have two vectors $y \in \mathbb{R}^n$ and $\hat{y} \in \mathbb{R}^n$, then $ME = \max_{i=1,\cdots,n} |\hat{y}_i - y_i|$. Prove ME is always greater than or equal to RMSE.

5 (10 points) Data Visualization

We will be using the UCI Wine dataset for this problem and Question 6. The description of the dataset can be found at `https://archive.ics.uci.edu/ml/datasets/wine`. You can load the dataset using the code below (recommended), or you can download the dataset ***here*** and load it yourself. You may refer the the Jupyter notebook `HW1-Q4-Q5.ipynb` for some skeleton code.

1. Show a scatter plot for the first 2 feature dimensions in 2-D space.

Some useful instructions are shown below:

- Import several useful packages into Python:

```
import matplotlib.pyplot as plt
from sklearn import datasets
```

- Load Wine dataset into Python:

```
wine = datasets.load_wine()
X = wine.data
Y = wine.target
```

Report ***your code*** and the ***scatter plot*** in Gradescope submission.

6 (10 points) Data Manipulation

We have already had a glimpse of the Wine dataset in Question 5. In this question, we will still use the Wine dataset. In fact, you can see the shape of array X is (178, 13) by running `X.shape`, which means it contains 178 data points and 13 features per data point. You may refer the the Jupyter notebook `HW1-Q4-Q5.ipynb` for some skeleton code. Here, we will calculate some measures of the array X and perform some basic data manipulation:

1. Show the first 2 features of the first 3 data points (i.e. first 2 columns and first 3 rows) of array X. (You can print the 3×2 array).

2. Calculate the mean and the variance of the 1st feature (the 1st column) of array X.

3. Randomly sample 3 data points (rows) of array X by randomly choosing the row indices. Show the indices and the sampled data points.

   Hint: You may use `np.random.randint()`.

4. Add one more feature (one more column) to the array X after the last feature. The values of the added feature for all data points are constant 1. Show the first data point (first row) of the new array.

   Hint: You may use `np.ones()` and `np.hstack()`.

Some useful instructions are shown below:

- Get a row or a column of the array X:

```
print X[0]          # Print the first row of array X.
print X[:, 0]       # Print the first column of array X.
                    # ':' here means all rows and '0' means column 0.
```

- Get part of the array:

```
print X[3:5, 1:3] # Print 4th and 5th rows, 2nd and 3rd columns.
print X[:3, :2]   # Print first 3 rows, first 2 columns.
```

- You may refer to a quick tutorial using NumPy here:
  http://cs231n.github.io/python-numpy-tutorial/

Report ***your code*** and the ***results of data manipulation*** in Gradescope submission.