Homework Assignment 2

CSE 251A: ML – Learning algorithms

Due: April 29th, 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) Nearest Neighbor

1.1 Nearest neighbor classification

You are given the points belonging to class-1 and class-2 as follows:
Class 1 points: $(11, 11), (13, 11), (8, 10), (9, 9), (7, 7), (7, 5), (16, 3)$
Class 2 points: $(7, 11), (15, 9), (15, 7), (13, 5), (14, 4), (9, 3), (11, 3)$
What is the label of the sample $(14, 3)$ using the nearest neighbor classifier using L2 distance?

1.2 Error rate of nearest neighbor

Give an example of a binary classification dataset with 3 points (x, y) for which the 1-NN classifier does not have zero training error (that is, it makes mistakes on the training set). You should plot the three points and show where the error is.

2 (10 points) Gradient Descent - Linear Regression

Consider house rent prediction problem where you are supposed to predict price of a house based on just its area. Suppose you have n samples with their respective areas, $x^{(1)}, x^{(2)}, \ldots, x^{(n)}$, their true house rents $y^{(1)}, y^{(2)}, \ldots, y^{(n)}$. Let's say, you train a linear regressor that predicts $f\left(x^{(i)}\right) = \theta_0 + \theta_1 x^{(i)}$. The parameters θ_0 and θ_1 are scalars and are learned by minimizing mean-squared-error loss through gradient descent with a learning rate α. Answer the following questions.

1. Express the loss function(L) in terms of $x^{(i)}, y^{(i)}, n, \theta_0, \theta_1$.
2. Compute $\frac{\partial L}{\partial \theta_0}$
3. Compute $\frac{\partial L}{\partial \theta_1}$
4. Write update rules for θ_0 and θ_1

3 (10 points) Gradient Descent - Linear Regression with L1 Regularization

Consider the same house rent prediction problem where you are supposed to predict price of a house based on just its area. Suppose you have n samples with their respective areas, $x^{(1)}, x^{(2)}, \ldots, x^{(n)}$, their true house rents $y^{(1)}, y^{(2)}, \ldots, y^{(n)}$. Let's say, you train a linear regressor that predicts $f(x^{(i)}) = \theta_0 + \theta_1 x^{(i)}$. The parameters θ_0 and θ_1 are scalars and are learned by minimizing mean-squared-error loss with L1-regularization through gradient descent with a learning rate α and the regularization strength constant λ. Answer the following questions.

1. Express the loss function(L) in terms of $x^{(i)}, y^{(i)}, n, \theta_0, \theta_1, \lambda$.
2. Compute $\frac{\partial L}{\partial \theta_0}$
3. Compute $\frac{\partial L}{\partial \theta_1}$
4. Write update rules for θ_0 and θ_1

Hint:

$$\frac{d|w|}{dw} = \begin{cases} 1 & w > 0 \\ \text{undefined} & w = 0 \\ -1 & w < 0 \end{cases}$$

4 (10 points) Gradient Descent - Linear Regression with L2 Regularization

Consider the same house rent prediction problem where you are supposed to predict price of a house based on just its area. Suppose you have n samples with their respective areas, $x^{(1)}, x^{(2)}, \ldots, x^{(n)}$, their true house rents $y^{(1)}, y^{(2)}, \ldots, y^{(n)}$. Let's say, you train a linear regressor that predicts $f(x^{(i)}) = \theta_0 + \theta_1 x^{(i)}$. The parameters θ_0 and θ_1 are scalars and are learned by minimizing mean-squared-error loss with L2-regularization through gradient descent with a learning rate α and the regularization strength constant λ. Answer the following questions.

1. Express the loss function(L) in terms of $x^{(i)}, y^{(i)}, n, \theta_0, \theta_1, \lambda$.
2. Compute $\frac{\partial L}{\partial \theta_0}$
3. Compute $\frac{\partial L}{\partial \theta_1}$
4. Write update rules for θ_0 and θ_1

5 (50 points) Implementing a Linear Regression Model from Scratch

Now, you will implement a linear regression model from scratch. We have provided a skeleton code file (i.e. LinearRegression.py) for you to implement the algorithm as well as a notebook file (i.e. Linear_Regression.ipynb) for you to conduct experiment and answer relevant questions. Libraries such as numpy and pandas may be used for auxiliary tasks (such as matrix multiplication, matrix inversion, and so on), but not for the algorithms. That is, you can use numpy to implement your model, but cannot directly call libraries such as scikit-learn to get a linear regression model for your skeleton code. We will grade this question based on the three following criteria:

1. Your implementation in code. Please do not change the structure of our skeleton code.
2. Your model's performance (we check if your model behaves correctly based on the results from multiple experiments in the notebook file).
3. Your written answers for questions in the notebook file.

6 (10 points) Ridge regression

Consider the loss function for ridge regression (ignoring the intercept term):

$$L(\mathbf{w}) = \sum_{i=1}^{n} (y^{(i)} - w \times \mathbf{x^{(i)}})^2 + \lambda ||\mathbf{w}||^2$$

where $(\mathbf{x^{(1)}}, y(1)), \ldots, (\mathbf{x^{(n)}}, y^{(n)}) \in R^d \times R$ are n data points and labels; and $\mathbf{w} \in R^d$. There is a closed-form equation for the optimal $\mathbf{w}$, but suppose that we decide instead to minimize the function using local search.

1. What is $\nabla L(\mathbf{w})$?
2. Write down the update step for gradient descent.

7 What to submit?

A single **PDF** file that includes:

1. Answers for Q1-Q4 and Q6
2. LinearRegression.py file for your linear regression model implementation. If you are unable to directly convert the .py file into a PDF, you can copy and paste all the code from that .py file into the Linear_Regression.ipynb within a single cell.
3. The Linear_Regression.ipynb file for your experiments and relevant answers. **Please do not clear the outputs of each code cell for submission as we need to check your model's outputs as well.**