Machine Learning

CAP 5610

Instructor: Dr. Mengxin Zheng

TA: Mayank Kumar

Department of Computer Science

University of Central Florida (UCF)

Spring, 2025

January 31, 2025

Assignment #2

Submission Instructions

Please submit your solutions via Canvas. You should submit your assignment as a PDF file.
Please do not include blurry scanned/photographed equations as they are difficult for us to grade.

Late Submission Policy

The late submission policy for assignments will be as follows unless otherwise specified:

1. 75% credit within 0-48 hours after the submission deadline.

2. 50% credit within 48-96 hours after the submission deadline.

3. 0% credit after 96 hours after the submission deadline.

Problem 1: Types of Attributes (10 points)

Classify the following attributes as nominal, ordinal, interval, ratio. **Explain why.**

(a) Rating of an Amazon product by a person on a scale of 1 to 5

(b) The Internet Speed

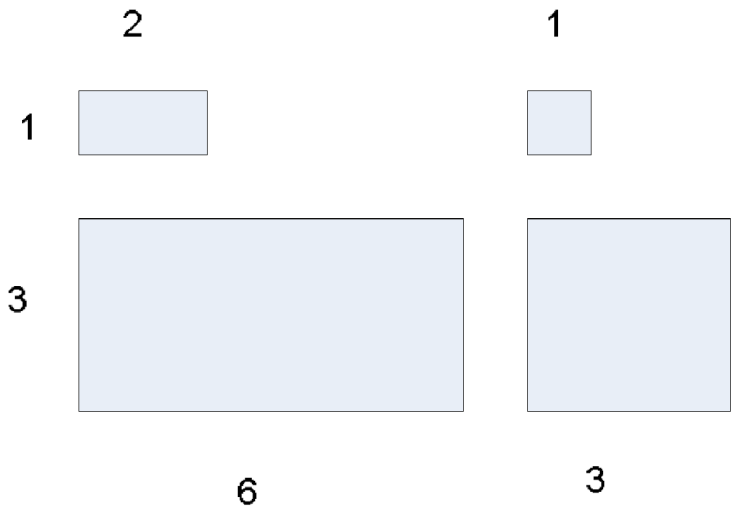
(c) Number of customers in a store.

(d) UCF Student ID

(e) Letter grade (A, B, C, D)

Problem 2: Distance/Similarity Measures (20 points)

Given the four boxes shown in the following figure, answer the following questions. In the diagram, numbers indicate the lengths and widths and you can consider each box to be a vector of two real numbers, length and width. For example, the top left box would be (2,1), while the bottom right box would be (3,3). Restrict your choices of similarity/distance measure to Euclidean distance and correlation. **Please explain your choice.**



(a) [10 points] Which proximity measure would you use to group the boxes based on their shapes (length-width ratio)?

(b) [10 points] Which proximity measure would you use to group the boxes based on their size?

Problem 3 (Coding Question 20 points)

Please write a Python code to calculate Cosine similarity, and Euclidean distance using NumPy. The input can be two randomly generated vectors or fixed vectors written by yourself.

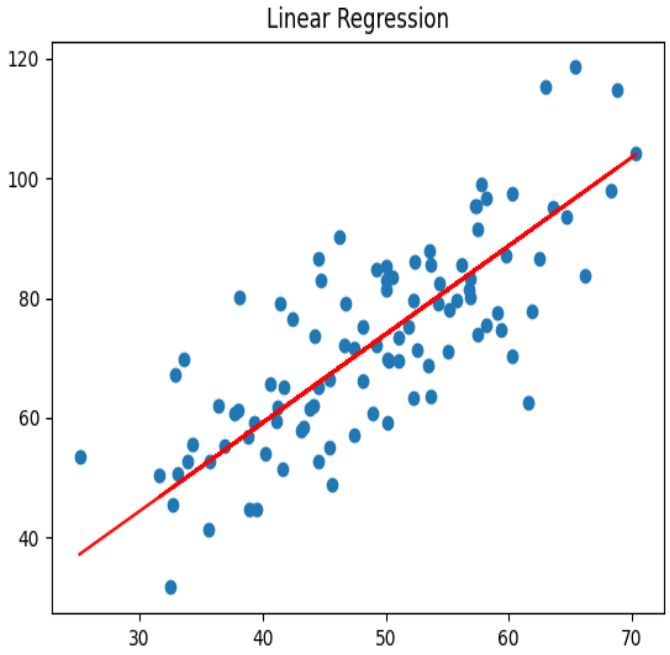
Note that: For Coding Questions, please **do not** directly call linear regression and non-linear regression built-in functions in existing library packages such as scikit-learn. You may call basic computation functions built in Numpy.

Problem 4 (Coding Question 25 points)

Please implement a Linear Regression to find the best linear model for the provided HW2_linear_data.csv. Please plot the result using "matplotlib.pyplot".

Note that

(1) The linear model is in the following format Y=mX+c
(2) Use MSE as the loss function
(3) You may use "pandas" to read the csv file and load the values into two vectors X and Y.
(4) Use Gradient Descent for the training. You may choose fixed learning rate (such as 0.0001) and epochs (such as 1000) without considering mini-batch.
(5) The result will look like the following image.



Problem 5 (Coding Question 25 points):

Please implement a non-linear regression to find the best cubic function model for the provided HW2_nonlinear_data.csv. Please plot the result, too.

(1) The cubic function is in the following format: Y=aX^3+bX^2+cX+d

(2) Use MSE as the loss function.

(3) Use Gradient Descent for the training. You may choose fixed learning rate (such as 0.000001 (1e-6)) and epochs (such as 10000) without considering mini-batch. It may take 10-15 seconds to finish the running for 10000 steps. Please be patient.

(4) The result will look like the following