Introduction to Machine Learning [ML01]

Assignment 1

1. Machine Learning problems:

1.1. Conceptional questions: (Make the correct answered underlined)

i- **Suppose your email program watches which emails you do or do not mark as spam, and based on that learns how to better filter spam. What is the task T in this setting?**

- o Classifying emails as spam or not spam.
- o Watching you label emails as spam or not spam.
- o The number (or fraction) of emails correctly classified as spam/not spam.
- o None of the above—this is not a machine learning problem.

ii- **Of the following examples, which would you address using an unsupervised learning algorithm? (Check all that apply.)**

- o Given email labeled as spam/not spam, learn a spam filter.
- o Given a set of news articles found on the web, group them into set of articles about the same story.
- o Given a database of customer data, automatically discover market segments and group customers into different market segments.
- o Given a dataset of patients diagnosed as either having diabetes or not, learn to classify new patients as having diabetes or not.

iii- You're running a company, and you want to develop learning algorithms to address each of two problems.

Problem 1: You have a large inventory of identical items. You want to predict how many of these items will sell over the next 3 months.
Problem 2: You'd like software to examine individual customer accounts, and for each account decide if it has been hacked/compromised.

Should you treat these as classification or as regression problems?

○ Treat both as classification problems.

○ Treat problem 1 as a classification problem, problem 2 as a regression problem.

○ Treat problem 1 as a regression problem, problem 2 as a classification problem.

○ Treat both as regression problems.

iv- Using numpy library: Assume you have three lists “List_x”, “List_y” and “List_z” as following:
List_x=[1,3,5,7]
List_y=[4,2,6,9]
List_z := $\sum_{i=0}^{size(List_{x|y})} \sqrt{x^2 + y^2}$

- Write a python code to calculate List_z, then get the index of the minimum element and the index of the maximum element in List_z and print the results.

  Your Code:

```
import numpy as np
List_x = [1,3,5,7]
List_y = [4,2,6,9]
#Write your code below.
```

1.2. Code Implementation:

You have a churn dataset:

- **About the dataset:**
  Customer churn, also known as customer attrition, occurs when customers stop doing business with a company. The companies are interested in identifying segments of these customers because the price for acquiring a new customer is usually higher than retaining the old one.
- **Hints:**
  Search online for how to spill the dataset into train and test datasets using Sklearn python library, and how to plot data using matplotlib.pylab.
- **Required:**
  1. Create python Notebook with your name and week number: "StudentName_WeekNumber.ipynb"
  2. Import the libraries that you will use.
  3. Load the dataset and show the first 5 data samples.
  4. Add the features to a pandas dataframe and numpy array, then do the same to the target "churn".
  5. Spill the features and the target to train and test datasets. [Hint: Search online.]
  6. Implement The **Nearest Neighbor Algorithm** from scratch, and calculate the error of your model.
  7. Implement The **K-Nearest Neighbor Algorithm** from scratch, then apply the function using k from 1 to 10, calculate the error of each k, then append it to a list Error[].
  8. Plot k values on the x-axis and the Error of each k on the y-axis, then from the graph conclude the k value that results the minimum error.
  9. Implement the **K-NN** algorithm using **Sklearn** library using the minimum k that you have observed from step 8, and calculate the error of the model.
  10. Compare between the errors of the 3 models (Step 6, 7, and 9)
  11. Write down your conclusion:

<u>Conclusion:</u>