1111 Deep Learning – Homework 1

Due: 10/22, 2022, 11:59pm

For the following questions, please upload the source code to moodle and show the results in your report.

1. Please load ‘data.mat’ into your Python code, where you will find . Now do the following procedures.
   1. (**5%**) Plot the data using plot function.
   2. (**10%**) Compute the least square line using the given data and overlay the line over the given data.
2. (**15%**) Using the same data from Question 2, compute the least square parabola (i.e. second order polynomial ) to fit the data. (**5%**) Explain which formulation (line or parabola) is more suitable for this dataset and why (please calculate the mean square error for these two fitting equations)?
3. (**15%**) Following the previous two questions, please randomly select 100 data samples for 50 times and plot these 50 lines (and parabolas (**)** in two separate figures, one for lines and the other for parabolas. (**5%**) Explain these visualizations based on the bias and variance.
4. (**10%**) In ‘train.mat,’ you can find 2-D points X=[x1, x2] and their corresponding labels Y=y. Please use logistic regression to find the decision boundary (optimal ) based on ‘train.mat.” Please use a gradient descent method to solve it and report the test error on the test dataset ‘test.mat.’ (percentage of misclassified test samples)
5. Download the MNIST dataset using the following example code:

##############################################

from \_\_future\_\_ import print\_function

import keras

from keras.datasets import mnist

# input image dimensions 28x28

img\_rows, img\_cols = 28, 28

# the data, split between train and test sets

(x\_train, y\_train), (x\_test, y\_test) = mnist.load\_data()

x\_train = x\_train.astype('float32')

x\_test = x\_test.astype('float32')

x\_train /= 255

x\_test /= 255

##############################################

Please randomly choose 1,000 different handwritten images from either the training or the testing dataset to construct your own dataset, where each digit has 100 data samples.

* 1. (**5%**) Use the following code to show 50 images in your own dataset.

##############################################

import numpy as np

import matplotlib.pyplot as plt

amount= 50

lines = 5

columns = 10

number = np.zeros(amount)

for i in range(amount):

number[i] = y\_test[i]

# print(number[0])

fig = plt.figure()

for i in range(amount):

ax = fig.add\_subplot(lines, columns, 1 + i)

plt.imshow(x\_test[i,:,:], cmap='binary')

plt.sca(ax)

ax.set\_xticks([], [])

ax.set\_yticks([], [])

plt.show()

##############################################

* 1. (**15%**) Normalize the data (subtracting the mean from it and then dividing it by the standard deviation) and compute the eigenpairs for the covariance of the data (sorted in a descending order based on eigenvalues).
  2. (**15%**) Please use PCA to reduce the 784-dimensional data to that with 500, 300, 100, and 50 dimensions, and then show the decoding results, respectively. How do you interpret the results?