Data Mining Assignment 1

Ahmet Oral

November 2020

.

DECLARATION OF HONOR CODE:
Student ID: 180709008
Name: Ahmet
Surname: Oral
In the course of Data Mining (CENG 3521), I take academic integrity very seriously and ask you to do as well. That's why, this page is dedicated to some clear statements that defines the policies of this assignment, and hence, will be in force. Before reading this assignment booklet, please first read the following rules to avoid any possible violation on academic integrity.

• This assignment must be done individually unless stated otherwise.

• You are encouraged to discuss with your classmates about the given assignments, but these discussions should be carried out in an abstract way. That is, you cannot copy code (in whole or in part) of someone else, cannot share your code (in whole or in part) with someone else either.

• The previous rule also holds for the material found on the web as everything on the web has been written by someone else.

• You must not look at solution sets or program code from other years.

• You cannot share or leave your code (in whole or in part) in publicly accessible areas.

• You have to be prepared to explain the idea behind the solution of this assignment you submit.

• Finally, you must make a copy of your solution of this assignment and keep it until the end of this semester.

I have carefully read every of the statements regarding this assignment and also the related part of the official disciplinary regulations of Mugla Sıtkı Kocman University and the Councilof Higher Education. By signing this document, I hereby declare that I shall abide by the rulesof this assignment to prevent any violation on academic integrity.

Signature

2 Regression Task

2.1 Curse of dimensionality

In this part I created a for loop with all dimension and tuple sizes.Loop runs 6 times, calculates and prints training time and error.I used "make_classification" to create a toy dataset and I split it with randomly with train_test_split.To apply linear regression with Stochastic Gradient Descent (SGD) solverI used SGDClassifier.At first I was going to use SGDRegressor but than I decided SGDClassifier would fit better.I calculated the error with mean_squared_error.
Table 1: The effectiveness and efficacy of linear regression algorithm with respect to the varying tuple and dimension size.

Tuple Size (m)	**Dimension Size (n)**	**Training Time (in ms)**	**Error**
a. 10,000	100	0.107985 s	0.131
b. 10,000	1,000	1.891934 s	0.16133
c. 10,000	2,000	1.919974 s	0.20666
d. 100,000	100	0.820766 s	0.0836
e. 250,000	100	1.293179 s	0.08969
f. 500,000	100	2.05887 s	0.11162

Here in a,b,c I kept the tuple size same and increased the dimension.As I increase the dimension we can see that training time is also increasing.So we can say more dimension equals more work on cpu.As the dimension size increases we can also see that Error is increasing.When dimension first increases 10 times error is increasing by 0.3.After that when dimension only increases 2 times and error is increasing by 0.39 times.We can see when dimension keeps growing, error is increasing exponentially.We can see the Curse of Dimensionality clearly.

In d,e,f I keep the dimension size same and increase tuple.Like dimensions as tuple size grows training time is also increasing.As tuple size grows training time increases with it.Also error is increasing as tuple size grow.We can also see that error is increasing exponentially.So we can say tuple size have positive relation with training time and error.

2.2 Sampling and dimensionality reduction

Table 2: The effect of dimensionality reduction and sampling strategies on learning time and performance

I used the same algorithms in first 3 steps.I created 2 for loops, one for first table, the other one is for second table.First loop keeps the tuple size same and decreases dimension size.Inside the loop I decreased dimension size using PCA.Loop runs 5 times to calculate training time and error of 5 different dimension sizes.Second loop does the same expect it decreases tuple size in each turn.To decrease tuple size I used resample algroithm from sklearn.

Dimension Size (n)	Training Time (in ms)	Error (cost)
500	0.635974 s	0.1573
100	0.136032 s	0.1476
10	0.015999 s	0.1386
4	0.011999 s	0.1423
1	0.012000 s	0.248

From the table we can see when dimension size gets smaller training time drops as expected.In the error part,first it seems to decrease but after dimension size is getting close to 1, error is increasing very fast.I think this is because of the lack of columns.Because there are so little columns predictions can't find the correct values so error increases.

Tuple Size (m)	Training Time (in ms)	Error (cost)
300,000	0.42087 s	0.1148
150,000	0.23828 s	0.1169
100,000	0.13563 s	0.1138
1000	0.001994 s	0.15
100	0.001994 s	0.2333

As tuple size decreases, training time is getting faster faster.Less tuple to work on = less work on cpu.On the other hand for first 3 values error stays around same values I think after we give around 100,000 tuples to train and test, predictions aren't that much affected.But after that, as there are less tuples to train and test, error keeps getting bigger.I think this is because of the lack of training values.I can't feed the computer with enough training values so how can I expect it to test it without error.

3 Classification Task

3.1 Visualization and binary-class classification

In this part I used the make_moons dataset from sklearn as asked in assignment.I divided in 0.3 ratio using trai_test_split.To run logistic regression algorithm with SGD solver I used SGDClassifier with loss = 'log' parameter to run with logistic regression.It runs with 10,000 iterations.I created the figure as similar to the figure in the assignment example.Colours and row/column values are same as in the assignment.I implemented the train and test values in the figure.You see all parameters in my code.

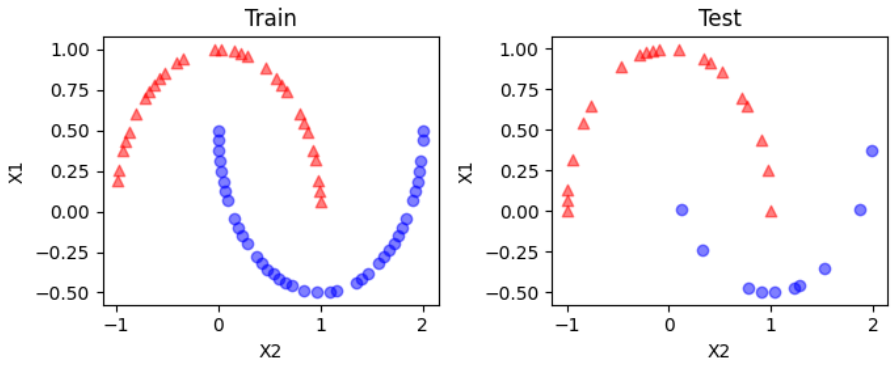

Figure 1: Visualization of makedataset.

3.2 Noising and multi-class classification

I tried to do 3.2 so much but whatever I do I failed.I couldn't understand the concept behind it.I did some online research and sometimes I thought I am getting closer but all my attempts has been for nothing :/ .I asked my friends for advice but it didn't help either.So sadly this part is empty.

3.3Evaluation/performance metric demonstration

I used the digits as asked and applied logistic regression algorithm with SGD solver with 10,000 iterations.I Implemented test and prediction values to confusion matrix then displayed it as heatmap.

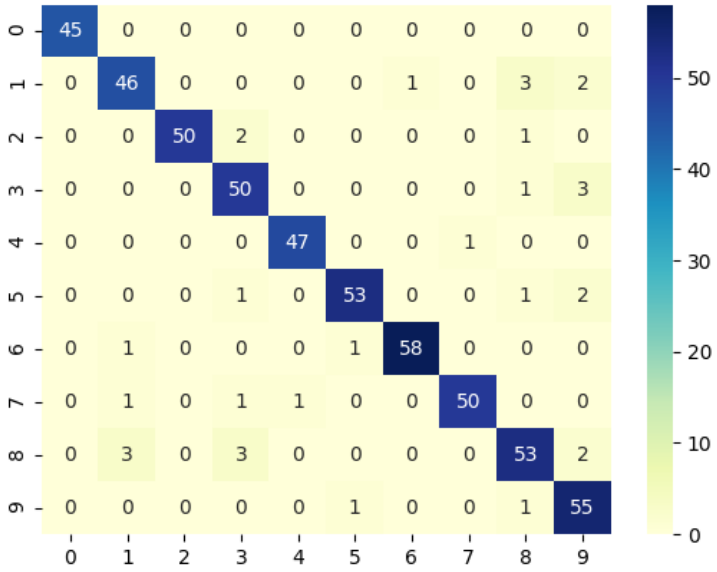

Figure 2: Heatmap chart revealing confusion matrix for digit dataset.

My thoughts:

(I tried to add detailed comments and explain my code in my .py file so more detailed explanations can be found there.)

I started by learning sklearn libraries and it was very hard for me.This assignment really pushed me because I didn't had any experience with this subject before.I don't remember working this hard on an assignment before.I think I spend over 20 hours but I also think I learned a lot.Anyways, I think while working on this assignment I learned most of the skills you said we would get as listed in the assignemnt.

Hardest part for me was to learn how to use all of the algorithms in sklearn library.There are more than 10 algorithms I needed to learn.I didn't know which ones to use at first so I tried lots of different approaches and failed many times :D Getting everything to run smoothly was really hard.At this point I am hoping that I didn't learn the subject in a wrong way.Because I am so new, I might misunderstood some of the main logic behind all of this.I will understand everything clearly after you show us the solutions of this assignment.
Thanks for reading.