HW4

CISC648010 - Fall 2021

Due Date: Oct 1st at 11 PM

1 Handwritten Image Classification 20 pts

Download the file mnist_49_3000.mat from Canvas. This is a subset of the MNIST handwritten digit database, which is a well-known benchmark database for classification algorithms. This subset contains examples of the digits 4 and 9. The data file contains variables x and y, with the former containing patterns and the latter labels. The images are stored as vectors.

To load the data use the following code:

```
import scipy.io
import numpy as np
data = scipy.io.loadmat('mnist_49_3000.mat')
x = np.array(data['x'])
y = np.array(data['y'][0])
y[y==-1] = 0
```

To visualize an image, type the followings:

```
from matplotlib import pyplot as plt
index = 0 #change the index to show different images
image = x[:,index].reshape(28,28)
plt.imshow(image, interpolation='nearest')
plt.show()
```

Implement Newton's method (a.k.a. Newton-Raphson) to find a minimizer of the regularized logistic regression. Try setting $\lambda = 10$. Use the first 2000 examples as training data, and the last 1000 as test data. Please report the following:

(a) (5 points) The test error

(b) (2 points) Your termination criterion (multiple options here)

(c) (3 points) The value of the objective function at the optimum

(d) (5 points) In addition, generate plot 5 images. These 5 images should be the 5 misclassified images for which the logistic regression classifier was

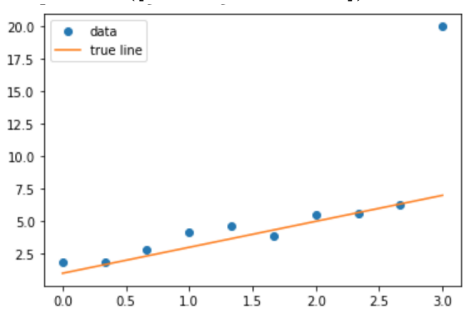
most confident about its prediction (you will have to define a notion of confidence in a reasonable way – explain what this is). In the title of each subplot, indicate the true label of the image. What you should expect to see is a bunch of 4s that look kind of like 9s and 9s that look like kind of like 4s. Hand in a printout.

(e) (5 points) To receive credit for this problem, please submit your code via Canvas, in a single file named prob3_lastname.py

2 Linear Regression (5pts each part)

Use the following line of code to generate a synthetic data set. Nine data points come from an upward sloping line plus noise. The last datapoint is an outlier.

```
import numpy as np
from matplotlib import pyplot as plt
np.random.seed(0)
n=10
x = np.linspace(0,3,n)
y = 2.0*x + 1.0+ 0.5*np.random.randn(n)
y[9] = 20
plt.plot(x,y,'o')
plt.plot(x,2*x+1)
plt.legend(['data','true line'])
```



(a) Solve ordinary least squares problem and report w, b.
(b) Solve ridge regression problem with $\lambda = 3$ and report w, b.
(c) plot data, true line, the lines obtained from part a and b in a figure
(d) to get full credit, upload your code on canvas named as LR_lastname.py