Intro2MachineLearningHW02 – Gökalp Ünsal 54040

The data set I used was the same as in the first homework. So, I used the same approach as last time while I was importing the data set label set. However, as we did in the lab session 04, I needed these data and label set for training and test parts as a whole. So the only difference I did was to compile those letter data together.

X: training data set (125*320 matrix)

X_test: data set of the test data (70*320 matrix)

y_truth_test: the true labels of the tet data. (70*1

y_truth: training data's labels (125*1), I changed its values as numeric so "A" will be "1" and E label will be "5".

Y_truth: Is a manipulated set of the y_trurh values. Instead of having only one column, Y_truth has 5 columns in which the cth column equals one if (y=c).

Homework and the 4th lab session were both about linear discrimination. In the 4th lab session we had to implement a linear discrimination algorithm for multiclass classification. However in this homework, we had to implement a discrimination by regression algorithm for multiclass classification. Because of the similarity, my while loop and gradient functions works as the same way in lab. As it is said in chapter 10.8 and the pdf, we use the sigmoid function rather than the softmax function. To get the best result in $r^t = y^t + (\text{Noise})$, the error must be minimized. I changed the error function in the while loop to the one in (10.44) in the book. I applied gradient descent to find the minimum point on the function. If the error function equals less then our threshold, the loop will break. After I got the training W and w0 values, I used the same values to test the test data set, which then I visualized in the confusion matrix afterwards.

The output is the following:

```
> print(confusion_matrix)
            y_truth
y_predicted  1  2  3  4  5
          1 25  0  0  0  0
          2  0 25  0  0  0
          3  0  0 25  0  0
          4  0  0  0 25  0
          5  0  0  0  0 25
> print(confusion_matrix1)
                 y_truth_test
y_predicted_test  1  2  3  4  5
               1 13  1  0  0  0
               2  1 11  0  0  1
               3  0  0 14  0  0
               4  0  1  0 14  0
               5  0  1  0  0 13
```