Untitled

Naveen Gabriel ,Sridhar Adhirkala

2019-02-21

1. Give an overview of the data from a machine learning perspective. Consider if you need linear or non-linear classifiers etc.

With respect to machine learning, data requires two primary things which is features and observations. Features are the characteristic of a observation. There are usually more than 1 feature and observations.Datas are given so that we can predict make model out of it and predict the future trends. With the given data there are can be primarly two task associated with it which is classification and regression.

Classification is required when we need to divide the data into different classes and predict them as accurately as one can. Regression is to calculate the discrete value given set of features and observations.

With respect to regression, we might get data which might be linear or non linearly related and accordingly we need to make model which may be polynomial or not. With respect to classification, if the data is linearly seperable then linear classifier is required which can be solved in 2D space but usually most of the data are non linearly seperable and due to lot features they are represented in multidimensional space. When the data is classfified using curve then it is called as non linear classifier.

2. Explain why the down sampling of the OCR data (done as pre-processing) result in a more robust feature representation.

Usually colored data is down scaled or normalized so that it becomes computationally less intensive and gives invariance to small distortions. Moreover by downscaling, the trace of digits and the background are much more distinguishable on the normalized scaled than colored map.

3. Give a short summary of how you implemented the kNN algorithm.

Knn.m is the function where Knn is actually implemented. The function accepts X(Features to be classified), k (Number of neighbours), Xt(Training features), LT (Correct labels of each feature vector).

Based on k we create matrix to save k nearest points. For each points, we find the distance of that point from all our training data and sort then sort the distance in increasing order. Using k, we pick up only the k nearest neighbour. Which ever k nearest neighbour is picked up, the label is assigned to the X based on majority vote of k nearest neighbour.

4. Explain how you handle draws in kNN, e.g. with two classes (k = 2)? For 4 data sets we didnt require k more than 1 and got accuracy of more than 99% in test set except for the 4th data set. In case of tie, we can label the class based on the point which is nearest. Moreover to handle this ambiguity we can choose k as always odd.

5. Explain how you selected the best k for each dataset using cross validation. Include the accuracy and images of your results for each dataset. Using cross validation, we are checking which vlue of k would give us accuracy more than 99% so which ever k gives accuracy more than 99%, we are breaking the loop and returning that k.

6. Give a short summary of your backprop network implementations (single + multi). You do not need to derive the update rules.

7. Present the results from the backprop training and how you reached the accuracy criteria for each dataset. Motivate your choice of network for each dataset. Explain how you selected good values for the learning rate, iterations and number of hidden neurons. Include images of your best result for each dataset, including parameters etc

Figure 1: Dataset-1, Accuracy=99%

Figure 2: Dataset-2, Accuracy=99%

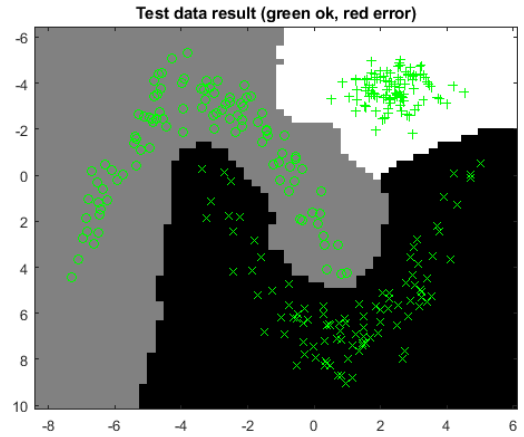

Figure 3: Dataset-3, Accuracy=100%

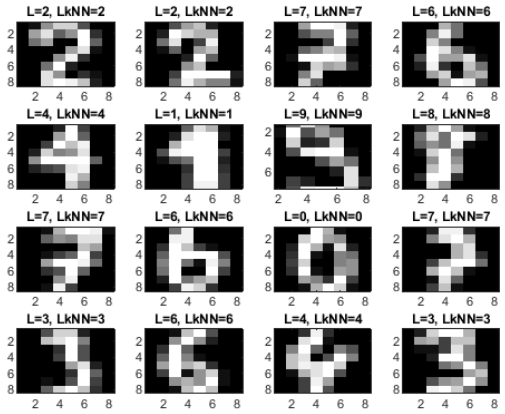

Figure 4: OCR, Accuracy=96.69%

8. Present the results, including images, of your example of a non-generalizable backprop solution. Explain why this example is non-generalizable

9. Give a final discussion and conclusion where you explain the differences between the performances of the different classifiers. Pros and cons etc.

10. Do you think there is something that can improve the results? Pre-processing, algorithm-wise etc.