Machine Learning Home Work

Home Work Tittle : Persian Hand Write OCR

Saeed Soleimanifar – 3971242016

Abstract

In this home work we implement different types of classifiers like 1NN , KNN , Parzen Window and Bayesian, we can compare the result of each classifier to give comments for problems solution and approaches.

Technical Details

We know to implement any kind of classifiers we should first evaluate data and and it's natural features to find rules which can be fitted on different classes and pass our guidelines to determine each class with a high accuracy. In-order to go through this process first we had a look on net and look for different approaches that other scientist had done for feature selections here are some search results:

1. trajectory recovery
2. handwriting contour
3. chain code histograms
4. zone detection

We applied zone detection although this feature extractions listed above , we have selected this because of fast code implementations and we can provide other choices later if we could not find an appropriate solution that pass our guidelines.

At the very beginning of our process we can see that fig 1 has given as input we had separated this picture to parts including : {cropped main box and digits}

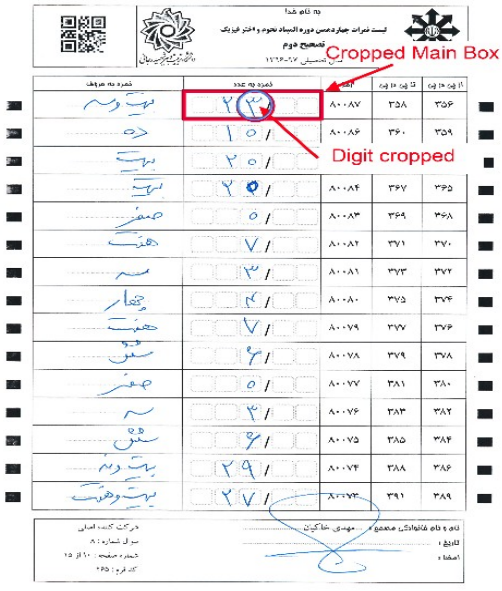


fig 1

As we see in fig1 cropped main box and digit crop area has mentioned clearly we have developed some image processing codes to obtain following results you can find all those image processing codes in getCroppedMainBox file developed in MATLAB.

finally we cropped around 10298 digits out of 651 forms in 9 sub-folders and now we are ready to create training sets and implementing code for classifiers, the more you should know is our feature extraction and selection phase.

First let's have look on digit pre-processing steps

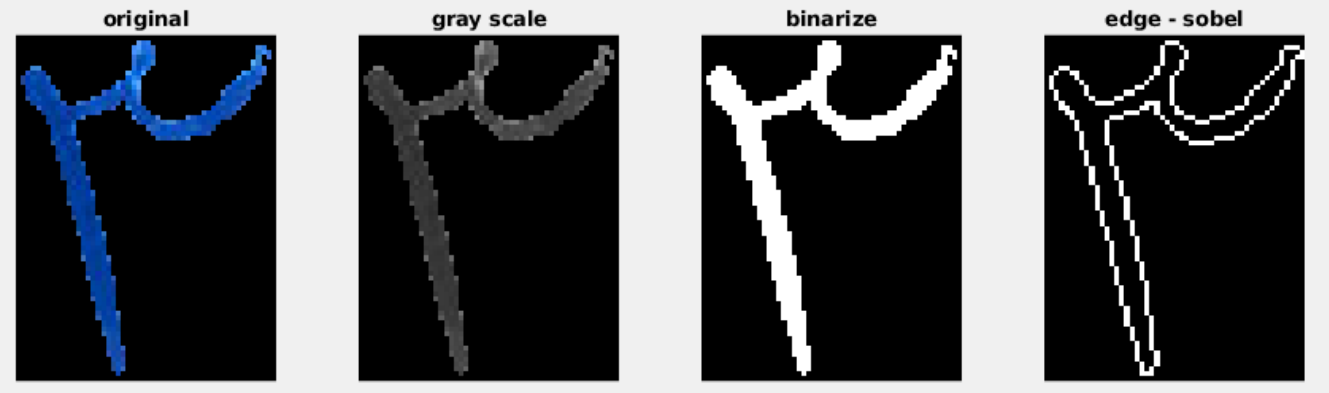


fig 2

You see in fig 2 that we had selected digit "3" after gray scale map and barbarize it with an automatic threshold and find shape edges the final image to process is created and shown in figure 2 under edge-sobel title.

There we can extract features from this data you can find the code of whole process in `getFeature10X6.m` file in MATLAB.

After this operations we calculate zones of data the size of our data window is 100 pixel in height and 60 pixel in width so we should resize digit image if it's original size is more than our fixed window after that we divide a 100x60 pixel to 60 zone of 10x10 pixel regions and finally we count each zones data by available one's in each zones, for example for zone (1,1) at fig 3 this value is zero and for zone(3,1) the value is 16. by above information we conclude that each zone can have a value from zero to 100.

So we can create a 10x6 feature vector for each digit we obtain this for all of our data set for creating a training set and we saved it in trainigSet.mat file for further use in classifires.

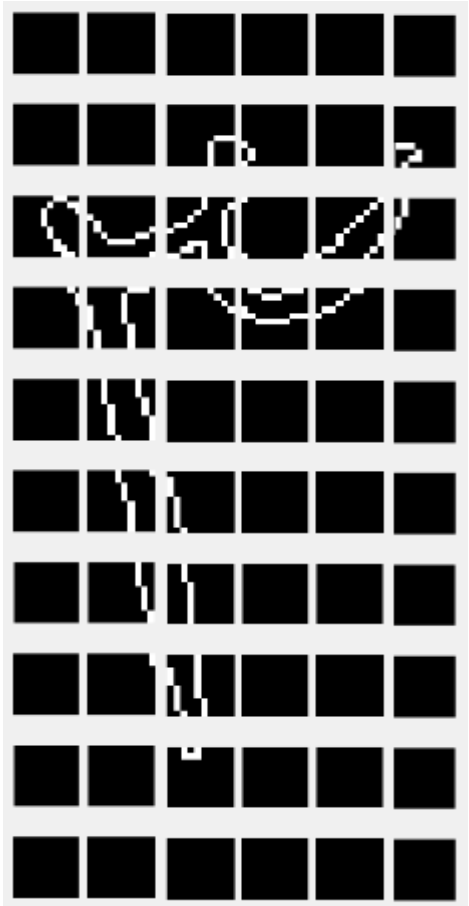


fig 3

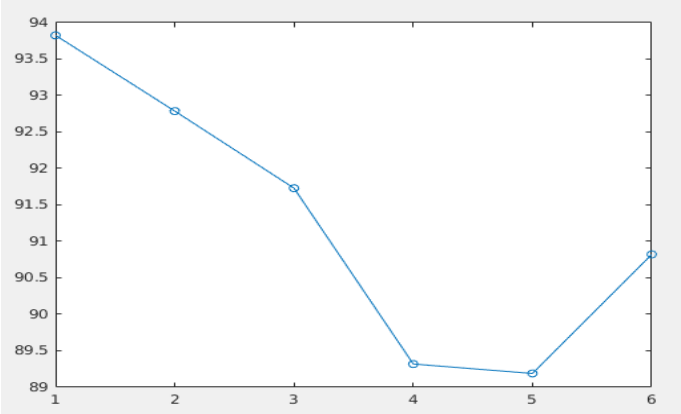
OCR Classifications :
1. 1-NN

We have developed one nearest neighbor classifier by calculate distance of test sample to training samples and find nearest neighbor of training sample and choose that class as result.
To obtain this we first calculate absolute subtraction of each 10x6 feature vector of test sample and training one and summation of the whole result in each row and column to calculate a single value distance of final result the winner class is the one which has minimum distance to test sample here is the result of this classifier over 40% test data up on all classes including error class is 93.825%.

2. K-NN

All we done in this classifier is similar to 1-NN but, at the end of function we choose the winner class by k nearest neighbor and maximum occurrence of a class. In table 1 we can see the results of accuracy after evaluating 40% of data as validation set.

Model Validation Results	
Method	Accuracy Result
1-NN	93.82
KNN K=3	92.79
KNN K=5	91.73
KNN K=7	89.32
KNN K=9	89.19
KNN K=11	90.82



3. Bayzian

As a specific research to implementing bayze classification we first should calculate PDF of each class individually from all training set data. To achieve this we first should evaluate our 10x6 feature vector because if we don’t reduce the diminutions before we calculate PDF we have to evaluate 60 PDF for each class and this make calculation time slower. This process is done with PCA analysis of data , first we create a matrix of 60 feature variable for all training samples and run PCA algorithm over it to reduce dimensions.

4. Parzen Window

Conclusion