CS57800: Statistical Machine Learning
Homework 1

Hengrui Zhang

Due: September 19, 2018 on Wednesday

1 Creativity of Categorizing Continuous Valued Features for Decision Tree

First of all, I normalized the value of Xs by using sigmoid function. Then, I reordered the whole dataset in a ascending order by the value of Y. To calculate the threshold of each attribute, I compute the threshold of each attribute which will get the best information gain of Y if the dataset is split by that threshold of that attribute. Also, the attribute is set to not just been split once.

2 Cross Validation

The 4-fold cross validation part can be accessed at the first function of decisiontree.py and the second function of knn.py programs. Function name is ”def crossValidation(dataSet,k)”. They are bascially the same. The input of the function is the whole dataset. The output is a list of length 4. Each element of the list is a dictionary, which represents each fold. There’s 3 key-value pairs in each dictionary, which are ’train set’: the content of train set,’valid set’: the content of valid set, ’test set’: the content of test set.

3 Accuracy and F1 Score Report

3.1 KNN

Table 1: Results of KNN in Each Fold

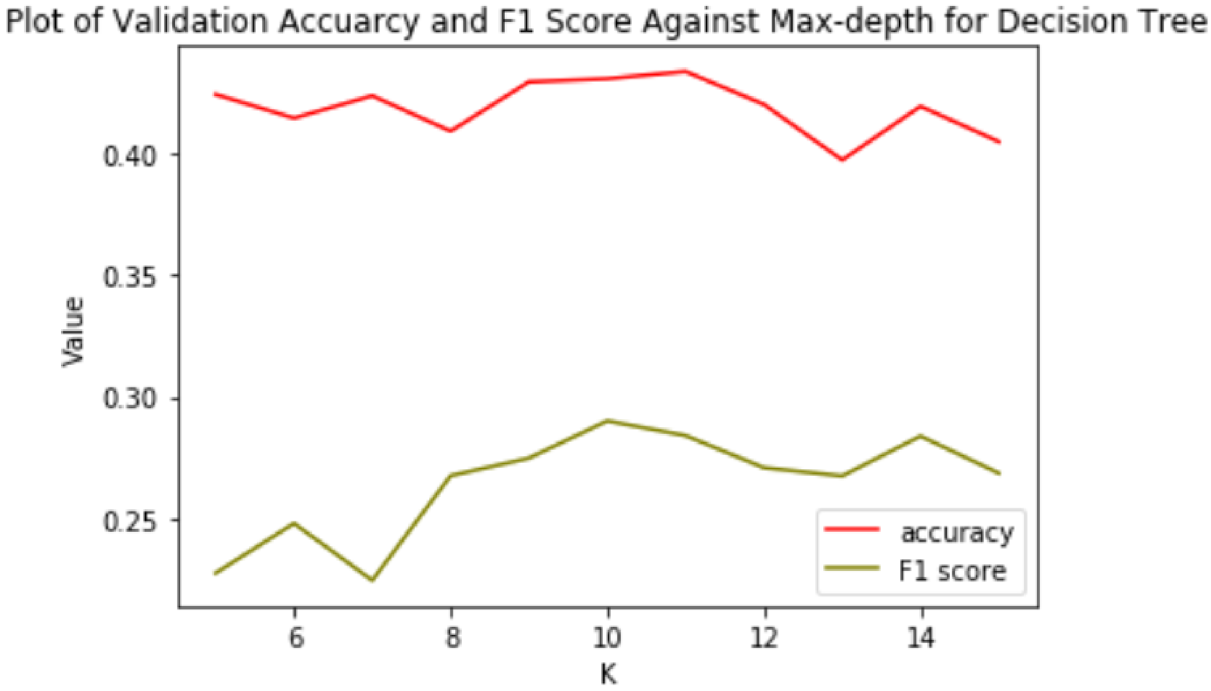
Fold	Validation Accuracy	Validation F1 Score	Test Accuracy	Test F1 Score
1	44.005	19.576	50.817	23.268
2	41.417	17.488	52.244	25.427
3	44.142	17.618	52.042	22.667
4	41.962	20.367	57.633	18.763

3.2 ID3 Decision Tree

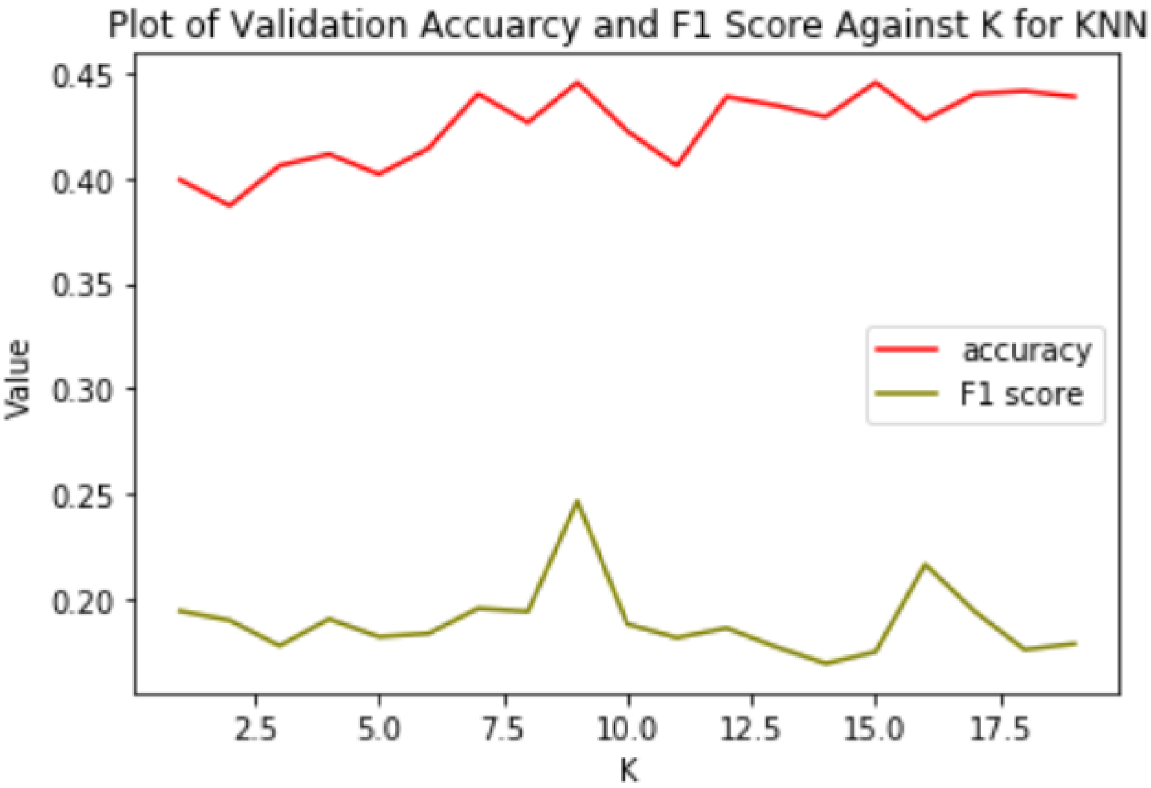
Table 2: Results of Decision Tree in Each Fold

Fold	Training Accuracy	Training F1	Validation Accuracy	Validation F1	Test Accuracy	Test F1
1	32.51	28.029	36.667	29.982	36.012	20.784
2	28.333	14.556	26.667	24.22	22.01	18.739
3	44.166	20.773	38.774	27.987	40.012	22.3557
4	22.885	35.625	29.1667	25.276	31.012	20.374

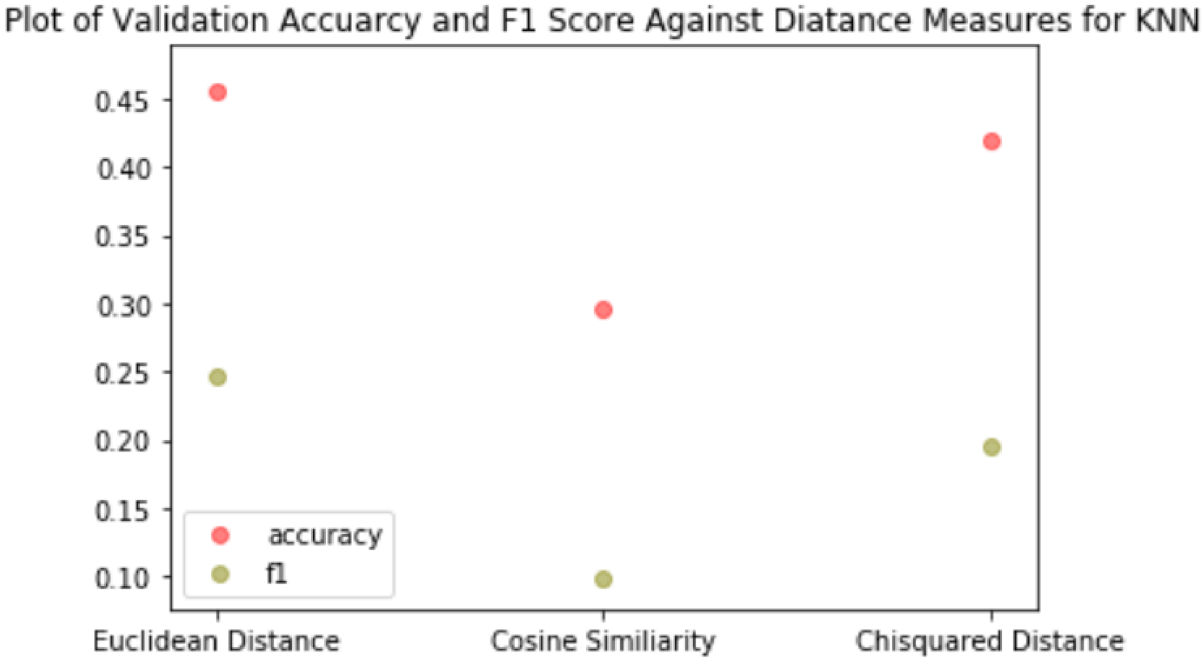
4 Graph of Tuning Hyper-parameters of Decision Tree



5 Graph of Tuning Hyper-parameters of KNN



As we can see from the graph above, the best K is 9.



The best performance is when using Euclidean Distance. As we can see from the above graph, the performance pf Chisquared distance is also not bad. It is slightly less than Euclidean distance. However, the problem is raised when computing the Chisquared distance. As we know the formula of Chisquare Distance is:

$$\sum_{i=1}^{n} \frac{(x_i - y_i)^2}{x_i + y_i} \tag{1}$$

There's some instances that the denominator is 0, which is uncomputatable. Hence, I choose to use Euclidean Distance as distance measures.

6 Answers

6.1

Max-depth and numbers of features are independent variables. However, it we allow the max-depth up to the number of features, it will most likely leading to over fitting.

6.2

6.2.1

KNN is unsupervised, Decision Tree is supervised. KNN determines neighborhoods by calculating distance. Distance metrics may be effected by varying scales between attributes and also high-dimensional space. DT predicts a class for a given input vector. The attributes may be numeric or nominal. Mostly if we want to find similar examples we use KNN. If we want to classify, we use Decision Tree.

6.2.2

KNN can be computationally expensive since it requires frequently check the training set.Decision Tree Classifier does not require such checking since it generate a in-memory model.

6.2.3

Decision Tree work in batches, modeling one group of training observations at a time. So it is not fit for incremental learning. But KNN naturally supports incremental learning since it is an instance-based model.

6.3

There's several ways to do that. 1. Mean rank of each item is calculated and the predicted ranking is obtained by ordering the mean ranks; 2. The top-choice frequency of each labels is calculated and is ordered to give the predicted ranking; 3. the most frequently observed ranking represents the predicted ranking; 4. Instead of estimating the probability of class membership using simple voting at the leaf where the test instance falls into, we can use similarity-weighted voting.

7 Additional Works

In my decision tree, minimum number of leaf is also involved as a hyper-parameter. After tuning the tree, I set this parameter equals to 5. The best performing fold has accauracy of 46.012 and f1 score of 30.303