Report of the 3rd exercise sheet

Marco Adamczyk (MA) Till Brinkmann (TB)

Submission by 06th December 2019

Tutorial: Tuesday, Tutor: Riza Velioglu

1 Introduction

This is an template for a report. You may use the outline or change it freely. There are no directives.

1.1 Datasets

"dataset1"

2 Methods/Models

Three different approaches were chosen to give a more broad impression of ways a classifier can be realized.

2.1 Bayes classifier

<Erkäre hier deinen classifer> TM

2.2 k-nearest-Neighbor classifier

The knn classifier represent the data on a n-dimational space and in this space the distance of dataset element to the new element is calculated by a specific metric. In the given case, the p-Norm with p = 1 is used, because the dataset vector entries are zero or one, so the relative distance will stay the same. After the distance calculation, the k closest dataset points to the new element are chosen. Now the new data has to be classified by those selected dataset points. There are multiple approaches. The naive apporach would be to select the class that appears on the selected data most often. More ideas are disscussed below. MA

2.3 Decision tree classifier

The classfier structures the trainingsdata as a tree. A node of a tree represents a decision.The decision made at a specific node is based on a features a data of the dataset has. For example if the norm of the set is greater then 0,25. After the step it will be compared how many element that have the feature are in a certain class. If all elements are from one class, the class can be assumed for the new data. If this is not the case the tree will be splitted into more features. When no obvious result can be selected the most probable class is chosen. MA

3 Bayes experiment

<Hier das experiment>

4 KNN experiment

The knn-algorithm was implemented as a class in python. A method was saving the training set and its outputs in two variables. With the second method "predict(element,k)" the estimated class for a new element was returned. In short, it calculated the distance in a help method for every data in the trainings set. Then it iterates through that list and selects the k-nearest elements. From the selected elements the class is chosen that appears the most in these elements. MA

4.1 Data

The two given datasets were used to test the classifier. The datasets were split in two parts. The testset contains 20 random elements from the data set. (When the last 20 element were taken for the testset without shuffeling the result for each k was similar. This indicates that the elements in the dataset are sorted in clusters of the same class.) The other data is used as the trainingset. Every data from the testset will be classified for a k = 1..10. As seen in the graphics bellow, there is not significant difference between the balanced and unbalanced dataset. The kNN only classifies with selected element not the whole dataset, so it is only the dependent how the classes in the dataset are distributed. MA

4.2 Results

Now the results are presented after running the kNN-classifier for the two given datasets. The entry is the percentage how often

Table 1: An Example of a Table

k	1	2	3	4	5	6	7	8	9	10
dataset 1	0.54	0.6	0.98							
dataset 2	0.74	0.54	0.48							

MA

5 Desision tree experiment

The Desision tree was implemented as a class in python. It uses a tree structure to save the subsets and its probabilities. After initializing the tree, it can be traversed for every new data without computing it again.

5.1 Data

Here we used both give datasets. Is difficult to extract relevant features, because there are no obvious categories of the data. There is also the risk of overfitting the data, so that to many features are used that are to detailed to descibe the datasset. The selected features: -The vector has more ones in the upper half on the array -The vector norm is above average.

5.2 Results

<Wenn du Zeit hast, kannst du das auch schnell noch implementieren. Sollte nicht schwer sein>

6 Discussion

As seen in the results some classifiers are more suited for specific tasks.

TB/MA

7 Further improvement/thoughs

7.1 Data improvements

Treegramms TB

7.2 kNN improvments

The kNNs selection of the class only the depends on the number of occurences of a certain class. It could be reasonable to include the distance, because the following problem could occur: Assume k = 4. 3 elements are from the class 1 and 1 element is from the class 0. Normally, the data would be classified as 1, but if the distances of the class 1 elements are far away and the element with class 0 is a lot closer it could be more reasonable to classify the new data as class 0. To include the distance as a new parameter to the calculation, the average distance from the new data and a specific class could be calculated and the compared. With this change, we get the following result for the same setup presented in the results above:

Figure 1: Results

You can refer to Figure 1. MA

Table 2: An Example of a Table

Data	Method 1	Method 2	Method 3
data 1	0.54	0.6	0.98
data 2	0.74	0.54	0.48
data 3	0.82	0.71	0.67