ALKAN ULUTAŞ

CMP5130 Homework #2

PCA & K-NN

1737285

Contents

[1. Homework Details 3](#_Toc502244133)

[2. Dataset 4](#_Toc502244134)

[2.1. Input Information 4](#_Toc502244135)

[2.2. Output Information 5](#_Toc502244136)

[3. Result 6](#_Toc502244137)

[4. Source Code 6](#_Toc502244138)

# Homework Details

The aim is to reduce dimensionality of a spam email dataset using principal component analysis (PCA) and forward selection approach. The dataset can be found at 'dataset.txt' file. Detailed information about the spam base dataset can be found at <http://archive.ics.uci.edu/ml/datasets/Spambase>

Use random half of the dataset for training and other half for validation by preserving the distribution of the classes in the original dataset. The number of features in the reduced subset will be optimized on validation set. Do not optimize nearest neighbor parameter or distance metric of k-NN.

For all the following cases, always use 5 nearest neighbor (k = 5) and Euclidean distance to implement k-NN classifier.

1) Feed the original dataset without any dimensionality reduction as input to k-NN.

2) Feature extraction: Use PCA to reduce dimensionality to m, followed by k-NN. Try for different values of m corresponding to proportion of variance of 0.80, 0.81, 0.82, ...., 0.99. Plot the data for m=2.

3) Feature Selection: Use forward selection to reduce dimensionality to m using k-NN as predictor. Train the model for each m between 1 and 57. Also plot the data for m=2.

**For Results**

1) For each case, choose the model that gives the highest classification accuracy on the training set, and report the classification accuracy, precision, and recall for this model obtained both on training and validation sets.

2) Plot the accuracy, precision, and recall of 'PCA+k-NN' and 'forward selection with k-NN' versus m (i.e. number of features in the reduced subset).

3) Briefly write your comparative comments about the obtained results.

# Dataset

# Input Information

I used original dataset as you give me, Sir. Dataset has 4601 instances and 57 attributes.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| word\_freq\_make | continuous | word\_freq\_credit | continuous | word\_freq\_pm | continuous |
| word\_freq\_address | continuous | word\_freq\_your | continuous | word\_freq\_direct | continuous |
| word\_freq\_all | continuous | word\_freq\_font | continuous | word\_freq\_cs | continuous |
| word\_freq\_3d | continuous | word\_freq\_000 | continuous | word\_freq\_meeting | continuous |
| word\_freq\_our | continuous | word\_freq\_money | continuous | word\_freq\_original | continuous |
| word\_freq\_over | continuous | word\_freq\_hp | continuous | word\_freq\_project | continuous |
| word\_freq\_remove | continuous | word\_freq\_hpl | continuous | word\_freq\_re | continuous |
| word\_freq\_internet | continuous | word\_freq\_george | continuous | word\_freq\_edu | continuous |
| word\_freq\_order | continuous | word\_freq\_650 | continuous | word\_freq\_table | continuous |
| word\_freq\_mail | continuous | word\_freq\_lab | continuous | word\_freq\_conference | continuous |
| word\_freq\_receive | continuous | word\_freq\_labs | continuous | char\_freq\_; | continuous |
| word\_freq\_will | continuous | word\_freq\_telnet | continuous | char\_freq\_( | continuous |
| word\_freq\_people | continuous | word\_freq\_857 | continuous | char\_freq\_[ | continuous |
| word\_freq\_report | continuous | word\_freq\_data | continuous | char\_freq\_! | continuous |
| word\_freq\_addresses | continuous | word\_freq\_415 | continuous | char\_freq\_$ | continuous |
| word\_freq\_free | continuous | word\_freq\_85 | continuous | char\_freq\_# | continuous |
| word\_freq\_business | continuous | word\_freq\_technology | continuous | capital\_run\_length\_average | continuous |
| word\_freq\_email | continuous | word\_freq\_1999 | continuous | capital\_run\_length\_longest | continuous |
| word\_freq\_you | continuous | word\_freq\_parts | continuous | capital\_run\_length\_total | continuous |

Table 1: Attribute Details

# Output Information

I developed my application on Microsoft .Net.

Firstly, I used the original dataset without any dimensionality reduction as input to k-NN. (k=5). (Experiment 1)

Second step was feature extraction. I used PCA to reduce dimensionality to m, followed by k-NN which I tried for different values of m corresponding to proportion of variance of 0.80, 0.81, 0.82, ...., 0.99. I tried m values between 1-57 with variance values which is between 0.80 and 0.99. (Experiment 2)

Thirdly; I used forward selection to reduce dimensionality to m using k-NN as predictor and trained the model for each m between 1 and 57. (Experiment 3). I got the best answer for m=30

**I got the best result by using PCA whose parameter m=32 and variance=0.80 which I mentioned before on second step.**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Experiment 1** | **Experiment 2** | **Experiment 3** |
| **Accuracy** | 87.7826 % | 90.3043 % | 90.2609 % |
| **Precision** | 0.837 | 0.860 | 0,860 |
| **Recall** | 0.849 | 0.894 | 0,893 |

Table 2: Results

# Result

As you see on Table 2, Experiment 2 has the best result for our study. Since Accuracy of KNN (PCA) has the highest value, but there is not too much difference between all experiments as you see, but I prefer to apply Experiment 2 with m=32 and variance=0.80.

# Source Code