CS669 Pattern Recognition

Odd Sem 2018

November 6^{th}, 2018

KNN & HMM

Assignment - 3

Group 04

Roll No	Name
B16066	Nikhil T R
B16019	Hemant Kumar

Contents

Chapter 1

Objective

1.1 KNN Algorithm

K Nearest Neighbours algorithm is a non parametric technique for classification. Here we want to explore the performance of the classifier for $K = 4, 8, 16, 32$. We will be using Dynamic Time Warping as the distance measure between 2 sequences.

1.2 HMM

The Hidden Markov Model assumes a first order Markov property for the dataset. We want to explore the performance of the classifier for $N = 2, 3, 5, 8$ and $M = 8, 16, 32$, where N is the number of states and M is the number of observation symbols.

1.3 Tasks

We have been provided with speech dataset for various utterances of "TA". Namely, "ta" (Class 1), "tA" (Class 2), and "TA" (Class 3).

- To test the KNN Algorithm using DTW as distance measure.
- To build and test a HMM for different values of N and M.

Chapter 2

Procedure

2.1 KNN

1. Implement the DTW algorithm assuming the feature vectors exist in a euclidean space.
2. Load the Training Data with class labels loaded appropriately.
3. Load the Testing Data with the actual class labels loaded appropriately.
4. Iterate through the list of the Testing sequences and assign a class label. Store it appropriately.
5. Using the actual and the assigned class labels, calculate the confusion matrix.
6. Using the confusion matrix, calculate the performance metrics: Recall, Mean Recall, Precision, Mean Precision, F-Measure, Mean F-Measure, and finally the accuracy.

2.2 HMM

2.2.1 Vector Quantization

1. Since the values for M are 8, 16, and 32, we need to perform K-Means clustering on the entire training dataset for $K = 8, 16, 32$.
2. Having obtained the means in each case, assign a class for each feature vector in each sequence in both the training and test data.
3. Make a separate folder of the data for each case.

2.2.2 Training

1. Estimate the initial parameters of HMM i.e. state transition matrix and state observation matrix from the vector quantized speech feature vectors for all the 3 training classes.
2. Reestimate the parameters i.e. state transition matrix and state observation matrix iteratively using Baum Welch method until the parameters converge to the accuracy of 0.001.

2.2.3 Testing

1. Classify the test data using the built HMM classifier.
2. Calculate confusion matrix and the performance metrics: Recall, Mean Recall, Precision, Mean Precision, F-Measure, Mean F-Measure, and finally the accuracy.

Chapter 3

Observations

The data set given:

1. Class 1 “ta”: 148 files.
2. Class 2 “tA”: 408 files.
3. Class 3 “TA”: 76 files.

3.1 KNN

General Observations:

1. The provided data set has a huge disparity in the number of data points for the classes.
2. The recall for the class 3 is very poor.
3. The accuracy peaks for $K = 8$ and then falls off.

3.1.1 K=4

Accuracy = 0.77

	class 1	class 2	class 3
Class 1	26	11	0
Class 2	6	94	2
Class 3	2	15	2

	class 1	class 2	class 3	Average
Precision	0.76	0.78	0.50	0.68
Recall	0.70	0.92	0.10	0.57
F Measure	0.73	0.73	0.73	0.73

3.1.2 K=8

Accuracy = 0.78

	class 1	class 2	class 3
Class 1	24	12	1
Class 2	2	99	1
Class 3	2	16	1

	class 1	class 2	class 3	Average
Precision	0.85	0.77	0.33	0.65
Recall	0.64	0.97	0.05	0.55
F Measure	0.73	0.73	0.73	0.73

3.1.3 K=16

Accuracy = 0.77

	class 1	class 2	class 3
Class 1	21	16	0
Class 2	1	101	0
Class 3	1	17	1

	class 1	class 2	class 3	Average
Precision	0.91	0.75	1.00	0.88
Recall	0.56	0.99	0.05	0.53
F Measure	0.69	0.69	0.69	0.69

3.1.4 K=32

Accuracy = 0.75

	class 1	class 2	class 3
Class 1	18	19	0
Class 2	1	101	0
Class 3	1	18	0

	class 1	class 2	class 3	Average
Precision	0.90	0.73	ND	ND
Recall	0.48	0.99	0.00	0.49
F Measure	0.63	0.63	ND	ND

3.2 HMM

Here N is the number of hidden states, and M is the number of observation symbols for the HMM. **General Observations**:

1. HMM has lesser accuracy as compared to KNN.

3.2.1 N=2

M=8

Accuracy = 0.48

	class 1	class 2	class 3
Class 1	22	11	4
Class 2	23	43	36
Class 3	4	4	11

	class 1	class 2	class 3	Average
Precision	0.45	0.74	0.22	0.47
Recall	0.60	0.42	0.58	0.53
F Measure	0.51	0.54	0.31	0.50

M=16

Accuracy = 0.55

	class 1	class 2	class 3
Class 1	28	4	5
Class 2	21	49	32
Class 3	3	6	10

	class 1	class 2	class 3	Average
Precision	0.53	0.83	0.21	0.53
Recall	0.75	0.48	0.52	0.59
F Measure	0.63	0.61	0.31	0.55

M=32

Accuracy = 0.56

	class 1	class 2	class 3
Class 1	29	4	4
Class 2	25	52	25
Class 3	4	7	8

	class 1	class 2	class 3	Average
Precision	0.50	0.82	0.21	0.51
Recall	0.78	0.50	0.42	0.57
F Measure	0.61	0.63	0.29	0.54

3.2.2 N=3

M=8

Accuracy = 0.46

	class 1	class 2	class 3
Class 1	19	10	8
Class 2	23	42	37
Class 3	4	3	12

	class 1	class 2	class 3	Average
Precision	0.41	0.76	0.21	0.46
Recall	0.51	0.41	0.63	0.51
F Measure	0.45	0.54	0.32	0.49

M=16

Accuracy = 0.52

	class 1	class 2	class 3
Class 1	24	8	5
Class 2	18	46	38
Class 3	1	6	12

	class 1	class 2	class 3	Average
Precision	0.56	0.76	0.22	0.51
Recall	0.65	0.45	0.63	0.58
F Measure	0.60	0.56	0.32	0.54

M=32

Accuracy = 0.51

	class 1	class 2	class 3
Class 1	21	11	5
Class 2	19	53	30
Class 3	3	9	7

	class 1	class 2	class 3	Average
Precision	0.48	0.72	0.16	0.46
Recall	0.56	0.51	0.36	0.48
F Measure	0.52	0.61	0.23	0.47

3.2.3 N=5

M=8

Accuracy = 0.45

	class 1	class 2	class 3
Class 1	19	10	8
Class 2	18	40	44
Class 3	3	3	13

	class 1	class 2	class 3	Average
Precision	0.47	0.75	0.20	0.47
Recall	0.51	0.39	0.68	0.52
F Measure	0.50	0.52	0.31	0.50

M=16

Accuracy = 0.51

	class 1	class 2	class 3
Class 1	24	5	8
Class 2	13	43	46
Class 3	1	4	14

	class 1	class 2	class 3	Average
Precision	0.63	0.82	20	0.55
Recall	0.65	0.42	0.73	0.60
F Measure	0.64	0.56	32	0.57

M=32

Accuracy = 0.54

	class 1	class 2	class 3
Class 1	23	10	4
Class 2	14	52	36
Class 3	3	8	8

	class 1	class 2	class 3	Average
Precision	0.57	0.80	0.16	0.51
Recall	0.62	0.50	0.57	0.59
F Measure	0.59	0.61	0.23	0.51

3.2.4 N=8

M=8

Accuracy = 0.49

	class 1	class 2	class 3
Class 1	17	12	8
Class 2	16	49	37
Class 3	3	4	12

	class 1	class 2	class 3	Average
Precision	0.47	0.75	0.21	0.48
Recall	0.46	0.48	0.63	0.53
F Measure	0.46	0.59	0.32	0.50

M=16

Accuracy = 0.53

	class 1	class 2	class 3
Class 1	26	6	5
Class 2	14	45	43
Class 3	1	5	13

	class 1	class 2	class 3	Average
Precision	0.63	0.80	0.21	0.55
Recall	0.70	0.44	0.68	0.60
F Measure	0.67	0.56	0.32	0.58

M=32

Accuracy = 0.50

	class 1	class 2	class 3
Class 1	24	10	3
Class 2	15	48	39
Class 3	3	9	7

	class 1	class 2	class 3	Average
Precision	0.57	0.71	14	0.48
Recall	0.65	0.47	0.37	0.49
F Measure	0.61	0.57	0.20	0.49

Chapter 4

Inference

4.1 KNN

1. Because of the disparity in the number of data points for training, we can see that as the value of K increases, the number of data points labelled as class 3 falls off to the point that it reaches 0 for $K = 32$.

2. $K = 8$ seems to be the best in terms of the accuracy.

4.2 HMM

1. Quantization of given data reduced the accuracy of HMM.

2. Observing the trend in the accuracies, we can say for sure that $M = 16$ & 32 outperform $M = 8$ in every case.

3. We cannot infer whether $M = 16$ or $M = 32$ is better because either one outperforms the other equal number of times.

4. The accuracies tend to decrease as we increase the number of states. They reach a minima at $N = 5$ and increase for $N = 8$.

Chapter 5

Conclusion

5.1 KNN

1. We should keep the number of data points in each file similar so that it doesn't cause trouble.

2. KNN Algorithm requires no training procedures, but the computation gets larger for larger data sets. The trade off is that, larger the number of data points, better the classification.

5.2 HMM

1. The accuracy of HMM classifier improves as we increase the number of clusters in vector quantization step provided reasonable number of states are taken and there is sufficient training data.

2. The initial MFCC feature vector was 39-dimensional. Applying dimensional reduction in order to remove redundant features and train our classifier on the basis of most relevant features would have resulted in a better HMM classifier.

3. We got some unexpected results wherein the classifier showed decrease in accuracy as we increased the number of states. But it increased for the last case of $N = 8$.

4. HMM was supposed to perform better than KNN because of the fact that it was taking into account the Markov Property of the sequence. A possible explanation is the very crude vector quantization technique which was used.