DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING, IIT MADRAS

CS5691: Assignment 3
Team 1

Sheth Dev Yashpal
CS17B106

Harshit Kedia
CS17B103

1 PART A

1.1 SPOKEN DIGITS DATA

Output Digit \ Target Digit	1	3	4	7	9	Total
1	12	0	0	0	4	16
3	0	12	0	0	0	12
4	0	0	12	0	0	12
7	0	0	0	12	0	12
9	0	0	0	0	8	8
Total	12	12	12	12	12	60

Table 1.1: Confusion Matrix for DTW applied on the Spoken Digits Data set. Accuracy for the model is **93.33%**.

Output Digit \ Target Digit	1	3	4	7	9	Total
1	12	0	0	0	4	12
3	0	12	0	0	0	12
4	0	0	12	0	0	12
7	0	0	0	12	0	12
9	0	0	0	0	12	12
Total	12	12	12	12	12	60

Table 1.2: Confusion Matrix for HMM applied on the Spoken Digits Data set. Accuracy for the model is **100%**.

- Looking at the confusion matrix we see that the HMM is the better performing model out of the two. But there is a catch here. For DTW case, we have chosen the majority voting among the $Top-39$ similarity cost and reported the Digit. This can be further optimized by reducing the the K in $Top-K$ or even by using a threshold to get a **100%** accuracy model.
- In the HMM case, we just need to look at the likelihood of a sequence corresponding to the model and make decision. For generating the codebook for HMM, we used K-Means Clustering

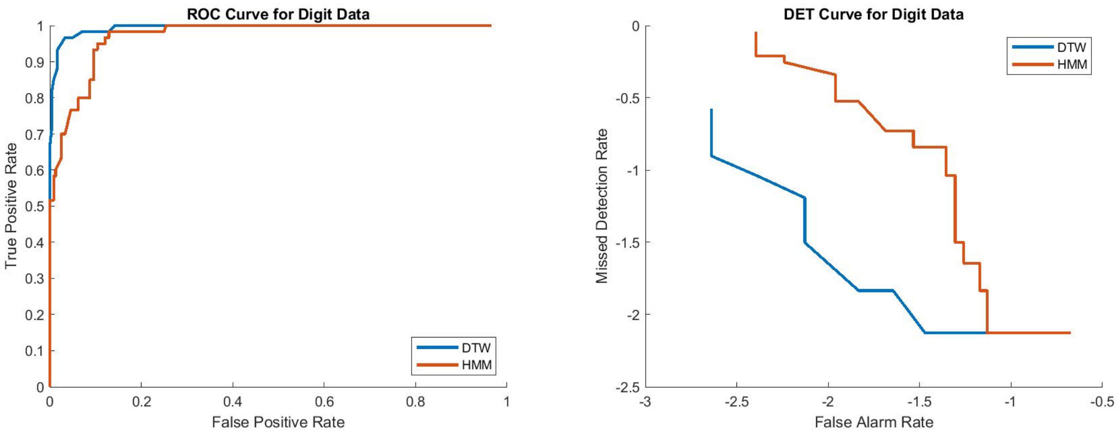

Figure 1.1: Here, we plot the ROC and DET curves for both our DTW and HMM models. We see that our DTW performs better than the HMM countering the confusion matrices presented above.

and generated one codebook each for one Digit data instead of generating an overall codebook over all digits. We have used $K = 5$.

- We have used 18 states for one, 10 states for three, 4 states for four, 13 states for seven and 10 states for nine in our individual HMMs.

- We also observe steps in our ROC and DET curves (Figure 1.1) because the likelihood of a sequence belonging to a model varies. For example, Probability that Seq. of 1 belongs to Model of 1 is e^{-37} while the same for digit 4 is e^{-130} and hence when we vary the threshold all these points will localize themselves on the ROC and DET plots and we will observe step sizes like these.

1.2 Handwritten Characters Data

Target Character / Output Character	ai	bA	chA	dA	tA	Total
ai	20	0	0	0	0	20
bA	0	18	0	0	2	20
chA	0	2	20	0	1	23
dA	0	0	0	20	0	20
tA	0	0	0	0	17	17
Total	20	20	20	20	20	100

Table 1.3: Confusion Matrix for DTW applied on the Handwritten Characters Data set. Accuracy for the model is **95%**.

- For Handwritten Data both confusion matrix and ROC, DET curves (Figure 1.2) point out that DTW performs better than HMM. We again observe the steps in the curves the reason for them is similar as before. For DTW, we have considered the $Top - 66$ similarity costs and reported the majority.

- For generating the codebook of HMM, we have calculated slope at each point and classified them among 8 directions (boundaries at intervals of 45 degrees).

Output Character \ Target Character	ai	bA	chA	dA	tA	Total
ai	20	0	0	0	0	20
bA	0	14	5	1	1	21
chA	0	4	15	4	5	28
dA	0	1	0	15	8	24
tA	0	1	0	0	6	7
Total	20	20	20	20	20	100

Table 1.4: Confusion Matrix for DTW applied on the **UNSCALED** Handwritten Characters Data set. Accuracy for the model is **70%**.

Output Character \ Target Character	ai	bA	chA	dA	tA	Total
ai	20	0	0	0	0	20
bA	0	20	0	0	0	20
chA	0	0	13	0	0	13
dA	0	0	7	20	0	27
tA	0	0	0	0	20	20
Total	20	20	20	20	20	100

Table 1.5: Confusion Matrix for HMM applied on the Handwritten Characters Data set. Accuracy for the model is **93%**.

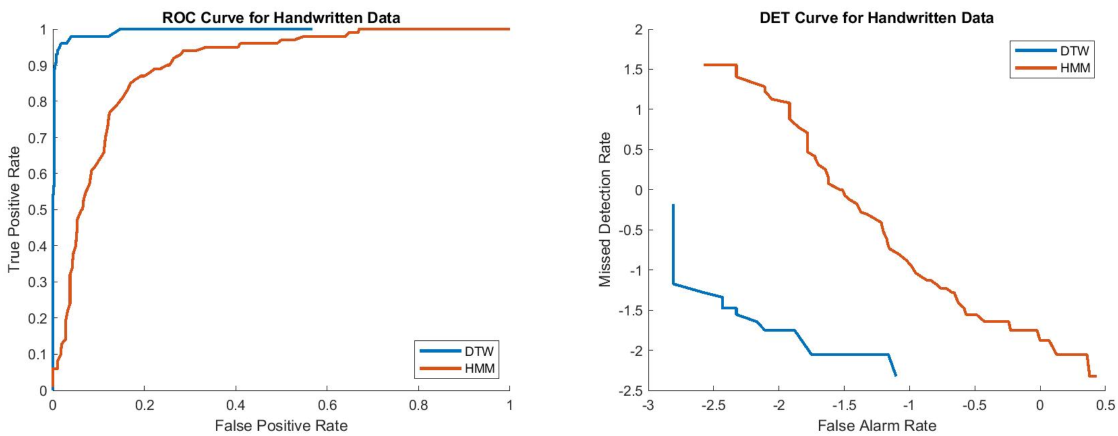

Figure 1.2: Here, we plot the ROC and DET curves for both our DTW and HMM models. We see that our DTW performs better than the HMM which is as expected from the confusion matrices presented above.

- We can see that the HMM is getting confused between characters **chA** and **dA**. Plotting the characters (Figure 1.3) reveals that they have been written in a way which makes them almost indistinguishable.

2 Part B

2.1 Handwritten Characters Data

- For the Handwritten character sequences we have just calculated the slope at each point and fed them into the HMMs to get the Answer. Below in Table 2.1 and Table 2.2 we show how our model performed for Development data and the predicted sequences for test data.

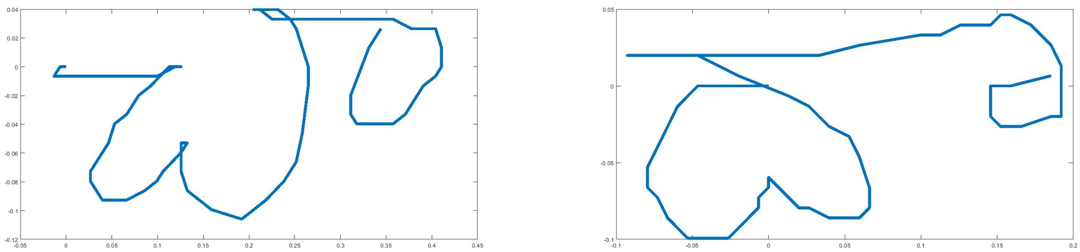

Figure 1.3: These are the characters **chA** and **dA** getting confused in the above HMM model and we can obviously see why it is happening from the above character plots.

- Note that we testes each sequence with a total of 155 HMMs. First 5 are single character. Next 25 are for two character sets so a total of 5x5 possibilities and last 125 representing all possibilities for three character sequences.

Target Sequence	Output Sequence	match ?
bA bA dA	bA bA dA	yes
bA chA chA	bA chA chA	yes
bA chA dA	bA chA dA	yes
bA tA chA	bA tA chA	yes
chA chA bA	chA chA bA	yes
chA chA chA	chA **dA** chA	**no**
chA chA dA	**dA** chA dA	**no**
chA chA tA	**bA** chA tA	**no**
chA tA chA	**bA** tA chA	**no**
dA bA bA	dA bA dA	yes
dA bA dA	dA bA dA	yes
dA chA dA	**chA** chA **chA**	**no**
dA dA dA	**chA** dA **chA**	**no***
dA dA tA	**tA** dA tA	**no**
dA tA dA	dA tA dA	yes
tA bA tA	tA bA tA	yes
tA bA tA	tA bA tA	yes
tA chA dA	tA chA **bA**	**no**
tA chA dA	tA chA dA	yes
tA tA bA	tA tA bA	yes
tA tA bA	tA tA bA	yes
tA tA chA	tA tA chA	yes

Table 2.1: Results for Development Data of Handwritten Telugu Characters for our set of 155 Sequential HMM models. The star(*) marked entry represents that we have attached a character plot (Figure 2.1) for the sequence.

- The same confusion between characters **chA** and **dA** has a cascading effect in our model as we used the same single character models to form our multi character models. In Figure 2.1 we have plotted the character sequence which is giving high likelihood but is unfortunately incorrect !

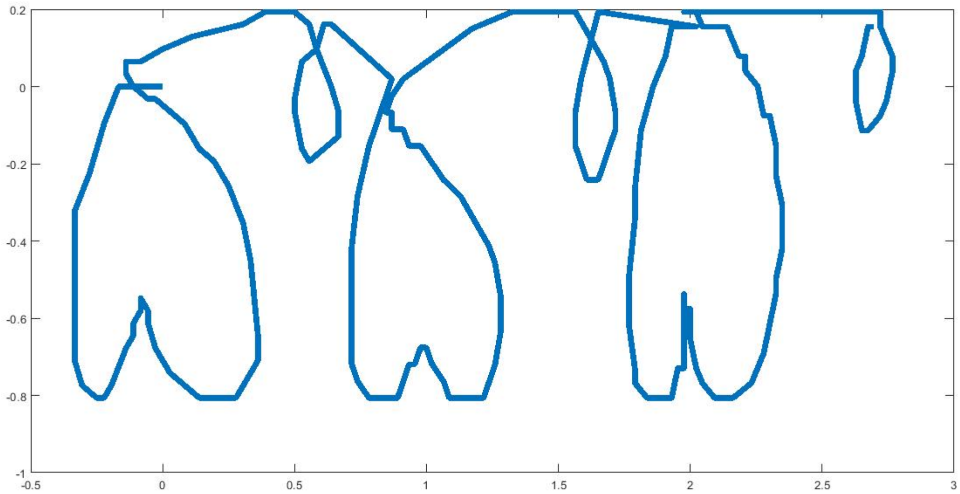

Figure 2.1: This is the plot for character sequence **dA dA dA**. We can see from our previous plots for character **chA** and **dA** that these two confuse a lot as they look the same. Hence we get the result **chA dA chA**.

Test Number	Output Character(s)
1	chA chA dA
2	tA bA bA
3	tA tA bA
4	tA bA bA
5	bA bA dA

Table 2.2: Results for T**TEST** Data of Handwritten Telugu Characters for our set of 155 Sequential HMM models.

2.2 Spoken Digits Data

- For the Spoken digit sequences we have performed K-Means clustering over all the pooled data for all digits keeping $K = 15$. Then using these means we generated symbol sequences for individual models and then for the sequence data models.
- We have used 18 states for one, 10 states for three, 4 states for four, 13 states for seven and 10 states for nine in our individual HMMs.
- We have tested 2-digit sequences with 25 HMMs and 3-digit sequences with 125 HMMs representing all possibilities. For the test data we test against all possible lengths and hence all 155 HMMs.
- We see that our model confuses a lot between one and four which is as expected. You can listen to the audio files for one and four and realize that it will be difficult to separate them.

Target Sequence	Output Sequence	match ?
11	91	no
13	13	yes
14	71	no
19	43	no
31	91	no
33	43	no
34	39	no
39	31	no
41	13	no
43	43	yes
44	19	no
47	47	no
71	41	no
74	71	no
77	77	yes
79	73	no
91	93	no
93	93	yes
97	94	no
99	93	no
111	494	no
141	491	no
171	341	no
174	973	no
331	431	no
344	434	no
393	333	no
413	139	no
434	431	no
449	413	no
477	477	yes
711	449	no
747	733	no
779	479	no
914	911	no
933	433	no
949	333	no
991	911	no

Table 2.3: Results for Development Data of Spoken Digits for our set of 25 and 125 Sequential HMM models.

Test Number	Output Digit(s)
1	343
2	49
3	141
4	313
5	419

Table 2.4: Results for **TTEST** Data of Spoken Digits for our set of 155 Sequential HMM models.