Exercise 1

Dejan Porjazovski

November 7, 2018

1 Question 1

- What are the properties of MFCC features that make them well suited for automatic speech recognition?

The MFCC decorrelates the features and reduces the dimensionality, which makes them suited for automatic speech recognition.
It uses linear filter banks on lower frequencies and increases bin size on higher frequencies.

- Why spectrogram or mel-spectrum wouldn’t work so well?

The power spectrogram contains a lot of data and redundancy. It also contains a lot of noise.
The mel spectrogram is better. It contains approximately 10 times less data. It also contains less noise and less redundancy compared to the power spectrogram.
MFCC uses discrete cosine transformation in order to decorrelate the features and reduce the dimensionality and is the most used method.

SOURCE CODE (1)

```
addpath /work/courses/T/S/89/5150/general/ex1
addpath /work/courses/T/S/89/5150/general/ex1/gmmbayestb
load ex1data

plot(sampleword);


s = spectrogram(sampleword, hamming(400), 240);
imagesc(sqrt(abs(s)))
axis xy
sample_word_segmentation


s2 = spectrogram(filter([1 -0.97], 1, sampleword), hamming(400), 240);
figure
imagesc(sqrt(abs(s2)))
axis xy
sample_word_segmentation


plot(M’, ’b’)


figure
imagesc(log(M*sqrt(abs(s2))+1))
axis xy
```

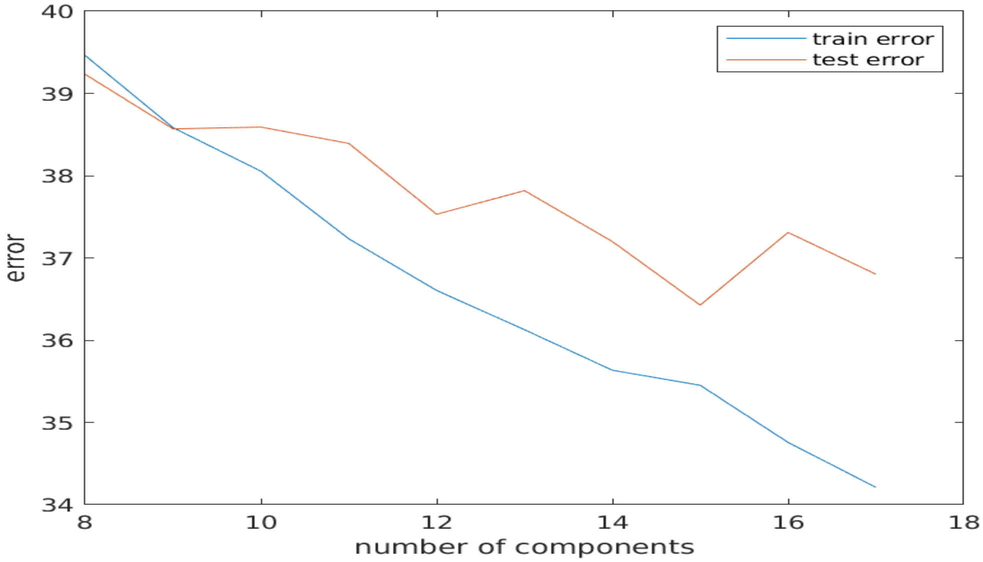
```
sample_word_segmentation


imagesc(D)
colorbar


figure
imagesc(D*log(M*sqrt(abs(s2))+1))
axis xy
sample_word_segmentation
```

2 Question 2

- Plot the error rates of both the train and the test sets with respect to the number of components in GMMs.



- Why are the recognition results with the train and the test set different?

The results are different because the classifier is trained on the training data and that is why is has lower error rate.
On the other hand, the test set data has not been seen previously by the classifier, so that is why it performs worse on it.

- What is a good number of components for recognizing an unknown set of samples?

From the plot we can see that the optimal number of components for recognizing an unknown set of samples is 15.

SOURCE CODE (2)

```
addpath /work/courses/T/S/89/5150/general/ex1
addpath /work/courses/T/S/89/5150/general/ex1/gmmbayestb
load ex1data

error = [];
error_test = [];
num_components = [];

for i = drange(8:17)
    S = train_gmm(train_data, train_class, i);
    result = gmmb_decide(gmmb_normalize(gmmb_pdf(train_data, S)));
    %phonemes(result(2991:3010))
    error = [error, length(find(result~=train_class))/length(train_class)*100];

    result_test = gmmb_decide(gmmb_normalize(gmmb_pdf(test_data, S)));
    error_test = [error_test, length(find(result_test~=test_class))/length(test_class)*100];

    num_components = [num_components, i];
end

figure1 = figure;
plot(num_components, error)
hold on;
plot(num_components, error_test)
hold off;
legend({’train error’, ’test error’});
xlabel(’number of components’);
ylabel(’error’);
```

3 Question 3

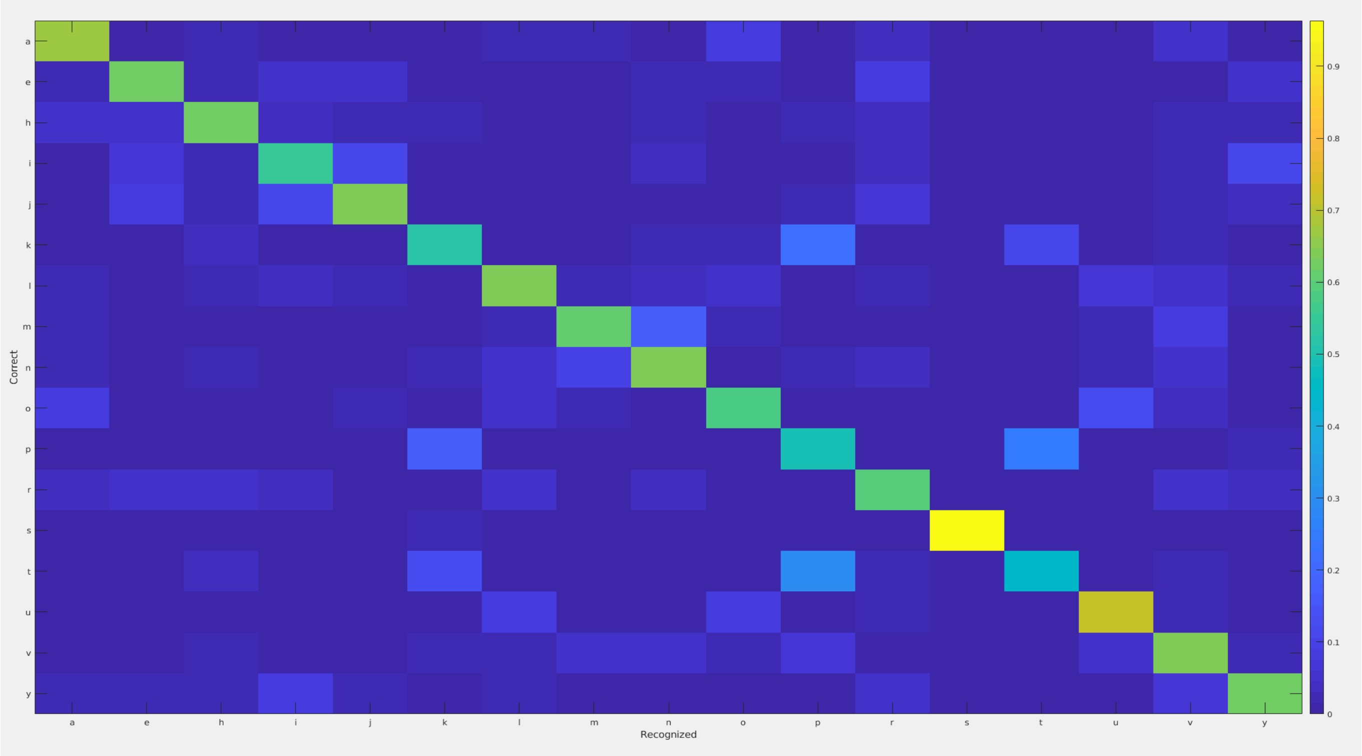
- Based on the confusion matrix, what can you conclude about phoneme recognition as a task and recognition performance of different phoneme classifiers?

Based on the confusion matrix, we can say that the classifier did fairly well recognizing the phonemes. The model did best recognizing the phonemes ’s’ and ’u’.

- Give examples of difficulties this classifier has.

The classifier had difficulties recognizing ’p’, ’t’, ’k’.
It mostly struggles recognizing the unvoiced phones.

- Include the visualized confusion matrix with the answer.



SOURCE CODE (3)

```
addpath /work/courses/T/S/89/5150/general/ex1
addpath /work/courses/T/S/89/5150/general/ex1/gmmbayestb
load ex1data


S = train_gmm(train_data, train_class, 15);

result_test = gmmb_decide(gmmb_normalize(gmmb_pdf(test_data, S)));
error_test = length(find(result_test~=test_class))/length(test_class)*100;

C = confusion_matrix(result_test, test_class);

plot_confusion(C, phonemes);
```

4 Question 4

- What problems do you see in the frame based classification if one wants to recognize whole words?

It doesn't work well for recognizing whole words because it repeats the same phoneme many times and also if it misclassifies a phoneme, the word might change it's meaning. For example for the first word (tw1) I got:

'ttptkkpkttpptkkkkkkkooooooooollllllloolmmmmmmmmmiiieeeeeyt'

I don't know Finnish that much but this might be the word 'kolme'.

The second word looks like this:

'sssssssssssssssseeeeeeeeeeeeeyiiyyiiiiiyiiiisssssssssssssssssssstt'

The third word looks like this:

'hrhhhyereeeeeeeellllirnnmmmmmvvviiiykktttkkkukouuuuuuuuuuusssssssssssssssstavaaaaoaavaon '

- Describe ideas to improve the results.

One idea would be to increase the number of GMM components because I am using 15, which gave me the lowest error but that might be local minimum.
We can also gather more data and maybe use different speakers.
Another idea for word recognition would be to use some kind of windowing to separate te word and find the most common phoneme in each window.

SOURCE CODE (4)

```
addpath /work/courses/T/S/89/5150/general/ex1
addpath /work/courses/T/S/89/5150/general/ex1/gmmbayestb
load ex1data

S = train_gmm(train_data, train_class, 15);


tw1_predict = gmmb_decide(gmmb_normalize(gmmb_pdf(tw1, S)));
first_word = [];

for i = drange(1, length(tw1_predict))
   first_word = [first_word, phonemes(tw1_predict(i))];
end
first_word


tw2_predict = gmmb_decide(gmmb_normalize(gmmb_pdf(tw2, S)));
second_word = [];

for i = drange(1, length(tw2_predict))
   second_word = [second_word, phonemes(tw2_predict(i))];
end
second_word



tw3_predict = gmmb_decide(gmmb_normalize(gmmb_pdf(tw3, S)));
third_word = [];

for i = drange(1, length(tw3_predict))
   third_word = [third_word, phonemes(tw3_predict(i))];
end
third_word
```

5 Bonus Question 1

- Which model performs classification better, the DNN or your best GMM?

The results are similar but DNN model performs slightly better than GMM. With more data, the DNN model should outperform the GMM with a lot more difference.

- The DNN training script tells you the number of parameters. Look inside your best model (struct S in Matlab). Which has more parameters?

The GNN model has 17 parameters and the DNN model has 145707.

6 Bonus Question 2

- Which model has lower classification error on the sampled MFCCs? Why might that be?

The DNN model has lower classification error which is 31.3% compared to 37% in the GNN model.

That might be because DNN model has a lot more parameters than the GNN model so it learns more about the data and thus has lower error.