CSC 791 - IOT Analytics

HMM Task1

This task is attempted for extra credit as I am in the section CSC 791

I have used matlab's hmmtrain package for training the model. The code is also attached below

>> HMM

Started with equi-probable values in the A, B and Pi matrices and used the hmmtrain to get the estimates for tranisition matrix A and emission matrix B. (This is explained in the code section below).

The values

Transition matrix (A) is:

0.333333333333333	0.333333333333333	0.333333333333333
0	1.000000000000000	0
0	0	1.000000000000000

Emission matrix (B) is:

0.159375	0.2225	0.2625	0.20125	0.154375
0.159375	0.2225	0.2625	0.20125	0.154375
0.159375	0.2225	0.2625	0.20125	0.154375

Initial States:

Pi = [1/3,1/3,1/3]

I found the state at the 1600th observation using the initial state Pi and transition matrix.

Then I used hmmgenerate package in matlab to predict the future values. The obtaied values for SSE, RMSE and R_square are:

SSE

1314

RMSE

1.812456896039186

R_square

0.952153872260259

As the R_square value is close to 1, the prediction done by the HMM model is quite good.

Code:

```
%read the csv file "PATHAK NEETISH"

data = csvread('PATHAK NEETISH.csv');

%extract training and testing data
data_train =  data(1:1600);
data_test = data(1601:2000);

%there are two ways to estimate the transmission and emission matrices :
%hmmestimate and hmmtrain
%hmmestimate requires the sequence of states to be present which is not the
case here
%so we use hmmtrain with an initial guesses for tranisition and emission
%matrices
%since there  are three states, let us consider an equi-probable
%tranisition matrix
trans_guess = [1./3,1./3,1./3;1./3,1./3,1./3;1./3,1./3,1./3];

%since there are five observations , we guess an equi-probable
%emission matrix
emission_guess =
[1./5,1./5,1./5,1./5,1./5;1./5,1./5,1./5,1./5,1./5;1./5,1./5,1./5,1./5,1./5
];

%also define  sarting probability for initial state
init_prob = [1./3 1./3 1./3];
init_prob2 = [1 0 0];
%estimate trans and emission matrix will be
[TRANS_EST, EMIS_EST]=
hmmtrain(data_train,trans_guess,emission_guess,'Maxiterations',1000,'Tolera
nce',1e-5);

disp('Transition matrix is: ')
disp(TRANS_EST);

disp('Emission matrix is: ')
disp(EMIS_EST);

%state of the system after 1600 observations
disp('State after 1600 observations');
%state after 1600 observation
p = init_prob * ((TRANS_EST)^1600);
disp(p);

%create an augmented matrix based on the state of the system afetr 1600
%traininig examples
TRANS_HAT = [0 p; zeros(size(TRANS_EST,1),1) TRANS_EST];
```

```
EMIS_HAT = [zeros(1,size(EMIS_EST,2)); EMIS_EST];

%generated Data
[seqData,statesData] = hmmgenerate(400,TRANS_HAT,EMIS_HAT);

%generated observations
disp('Generated sequence of Observations');
disp(seqData)

%calculate errors
%calculate SSEdiff will be the errors
diff = (transpose(seqData)-(data_test));
SSE = sum(diff.^2);
disp('SSE');
disp(SSE);

RMSE = rms(diff);
disp('RMSE');
disp(RMSE);

disp('R_square');
meanDataTest = mean(data_test);
meanDataTestVec(1:400) = meanDataTest;
%disp(meanDataTestVec);
SST = sum((transpose(data_test)-meanDataTestVec).^2);
SSR = sum((seqData-meanDataTestVec).^2);
R_sq = SSR/SST;
disp(R_sq);
```