KTH - Royal Institute of Technology

Pattern Recognition (EQ2340) - Exercise Project A.3 - Backward Algorithm

Alessio Russo - alessior@kth.se (911103-T192)
Lars Lindemann - llindem@kth.se (891113-4131)

October 2015

Contents

1 Test: Finite duration HMM

To test the finite duration HMM the example from the assignment was used in order to compare the results.

In that example we have the following parameters for the HMM:

$$A = \begin{pmatrix} 0.9 & 0.1 & 0 \\ 0 & 0.9 & 0.1 \end{pmatrix} \quad \mathbf{q} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \mathbf{b} \sim \begin{pmatrix} N(0,1) \\ N(3,4) \end{pmatrix} = \begin{pmatrix} f_1(x) \\ f_2(x) \end{pmatrix}$$

From the example we know that given $\underline{x} = (-0.2, 2.6, 1.3)$ we obtain from the Forward Algorithm the scale factor:

$$\underline{c} = (1, 0.1625, 0.8266, 0.0581)$$

Using this data we ran the following test:

```
%% finite test case
A       = [0.9 0.1 0;0 0.9 0.1];
q       = [1;0];
mc      = MarkovChain(q,A);

B(1)    = GaussD('Mean',0,'StDev',1);
B(2)    = GaussD('Mean',3,'StDev',2);

x   = [-0.2 2.6 1.3];
c   = [1; 0.1625; 0.8266; 0.0581];

pX = prob(B,x);
betaHat=backward(mc,pX,c)
```

Where $prob(\mathbf{b}, \underline{x})$ computes $f_i(x_j)$.
The output of the test is the following one:

```
betaHat =

    1.0003    1.0393         0
    8.4182    9.3536    2.0822
```

Which is equal to the result obtained in the assignment example.

2 Test: Infinite duration HMM

For the test of an infinite HMM an example of the course compendium on page 121 has been taken. This example uses a discrete output distribution.

$$A = \begin{pmatrix} 0.3 & 0.7 & 0 \\ 0 & 0.5 & 0.5 \\ 0 & 0 & 1 \end{pmatrix} \quad \mathbf{q} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.5 & 0.4 & 0.1 \\ 0.1 & 0.1 & 0.2 & 0.6 \end{pmatrix}$$

The observed sequence is $\underline{x} = (1, 2, 4, 4, 1)$ and the forward algorithm gives:

$$\underline{c} = (1, 0.35, 0.35, \frac{79}{140}, 0.0994)$$

In order to compare the computational results, the backward variables have been calculated by hand as described in the course compendium, i.e. for $t = 4$ and $t = 5$:

$$\begin{aligned}
\hat{\beta}_{1,5} &= \frac{1}{0.0994} = 10.0604 \\
\hat{\beta}_{2,5} &= \frac{1}{0.0994} = 10.0604 \\
\hat{\beta}_{3,5} &= \frac{1}{0.0994} = 10.0604 \\
\hat{\beta}_{1,4} &= \frac{140}{79}[\frac{0.3}{0.0994} + 0 + 0] = 5.3485 \\
\hat{\beta}_{2,4} &= \frac{140}{79}[0 + 0 + \frac{0.5 \cdot 0.1}{0.0994}] = 0.8914 \\
\hat{\beta}_{3,4} &= \frac{140}{79}[0 + 0\frac{0.1}{0.0994}] = 1.7828
\end{aligned}$$

This recursion goes on until the final result is obtained

$$\hat{\beta} = \begin{pmatrix} 0.9997 & 0.3311 & 0.1783 & 5.3485 & 10.0604 \\ 0.9760 & 2.8562 & 1.6555 & 0.8914 & 10.0604 \\ 0.5239 & 5.2394 & 3.0563 & 1.7828 & 10.0604 \end{pmatrix}$$

Afterwards, the following algorithm has been run

```
clc; clear all; close all;

A       = [0.3 0.7 0;0 0.5 0.5; 0 0 0.1];
q       = [1;0;0];
mc      = MarkovChain(q,A);

B(1)    = DiscreteD([1 0 0 0]);
B(2)    = DiscreteD([0 0.5 0.4 0.1]);
B(3)    = DiscreteD([0.1 0.1 0.2 0.6]);

x   = [1 2 4 4 1];
c   = [1; 0.35 ; 0.35; 79/140; 0.0994];

pX  = prob(B,x);
%pX = [1 0 0 0 1;0 0.5 0.1 0.1 0;0.1 0.1 0.6 0.6 0.1];

betaHat=backward(mc,pX,c)
```

The output of this lines is

```
betaHat =

    0.9997    0.3311    0.1783    5.3485   10.0604
    0.9760    2.8562    1.6555    0.8914   10.0604
    0.5239    5.2394    3.0563    1.7828   10.0604
```

which proves that the algorithm also works for infinite HMM. Furthermore a second test has been run which is omitted here.