Assignment4 - The Forward Algorithm

Carl Holmqvist and Philip Stiff

October 16, 2017

1 Correcting the forward pass algorithm

1.1 The bugs and their implications

All line numbers below refer to the original line numbers in the downloaded *forward_fixme.m*.

1. Line 40 was changed from:

   ```
   T=size(pX,1);
   ```

   to:

   ```
   T=size(pX,2);
   ```

   This error caused the number of timesteps to be set to the number of possible states. This bug could be invisible if there was as many states as timesteps.

2. Line 43 was changed from:

   ```
   numberOfStates = length(mc.InitialProb)+10;
   ```

   to:

   ```
   numberOfStates = length(mc.InitialProb);
   ```

   This error caused the number of states to be 10 too many.

3. Line 48 was changed from:

   ```
   c = zeros(numberOfStates);
   ```

   to:

   ```
   c = zeros(1, numberOfStates);
   ```

   This error caused the c array to be quadratic in size when running the code on infinite HMMs, which would of course lead to wrong outputs.

4. Line 54 was removed:

   ```
   q = [q;q];
   ```

   This error did not produce any wrong results, but it is still a logical fault since q:s dimensions were correct to start with.

5. Line 61 was changed from:

   ```
   initAlfaTemp = zeros(2,1);
   ```

   to:

   ```
   initAlfaTemp = zeros(numberOfStates,1);
   ```

   This error caused the length of alphaTemp to always be wrongly set to 2, when it in reality should be *numberOfStates* long. This would cause the program to crash in the following loop.

6. Line 78 was changed from:

   ```
   alfaTemp(j) = sum(B(j,t))*(alfaHat(:,t-1)'*A(:,j));
   ```

   to:

   ```
   alfaTemp(j) = B(j,t)*(alfaHat(:,t-1)'*A(:,j));
   ```

   Since B(j, t) is simply a number the sum command will not make any difference, but it is unnecessary.

7. Line 82 was changed from:

   ```
   alfaTemp(j) = alfaTemp(j)/c(1);
   ```

   to:

   ```
   alfaTemp(j) = alfaTemp(j)/c(t);
   ```

   The value was always divided by the first c value, when it should be divided by the current one. This leads to wrong results.

8. Line 87 was removed:

   ```
   [rows,columns] = size(A);
   ```

   These values had already been collected, so this line was simply redundant.

9. Line 90 was changed from:

   ```
   c(T) = alfaHat(:,T)'*A(:,numberOfStates+1);
   ```

   to:

   ```
   c(T+1) = alfaHat(:,T)'*A(:,numberOfStates+1);
   ```

   This bug only effected the results when running on a finite HMM. It wrongly set c(T+1):s value to c(T), causing both the value in c(T) to be wrong and the c array to be too short.

All code using the variables *Alpha_old* or Z were removed, or commented, as these variables had no function.

1.2 Verification

We verified that the corrected implementation of the algorithm worked also for an infinite HMM by manually computing the forward pass for the following HMM setup:

$$q = \begin{pmatrix} 1 \\ 0 \end{pmatrix} A = \begin{pmatrix} 0.9 & 0.1 \\ 0 & 1 \end{pmatrix} B = \begin{pmatrix} \mathcal{N}(0,1) \\ \mathcal{N}(3,2) \end{pmatrix}$$

We first generated the random observation sequence $\underline{x} = (0.7172, 1.6302, 0.4889)$ of length 3 using the stated HMM setup. By manual calculations of the forward algorithm this resulted in the following scale factors $\underline{c} = (0.7479, 0.4208, 0.6610)$, using a normalized B.

By manually calculating $\hat{\alpha}$ (and rounding some numbers along the way) we got the following

$$\hat{\alpha} = \begin{pmatrix} 1 & 0.8577 & 0.9297 \\ 0 & 0.1423 & 0.0703 \end{pmatrix}$$

By running the corrected implemented algorithm we got the following

$$\hat{\alpha} = \begin{pmatrix} 1 & 0.8577 & 0.9296 \\ 0 & 0.1423 & 0.0704 \end{pmatrix}$$

The results were similar enough for us to be convinced that the corrected implementation is correct.

2 Creating a song database

We have created a song database consisting of 8 songs, with 16 samples of each song being hummed. The songs in the database are all intros from swedish childrens songs and were chosen because they were easy to hum.

The following songs were chosen:

1. Bä bä vita lamm
2. Blinka lilla stjärna
3. En sockerbagare
4. Björnen sover
5. Vem kan segla för utan vind
6. Imse vimse spindel
7. Du käre lille snickerbo
8. Prästens lilla kråka

The songs were recorded by two people, 8 samples per person, in a home environment using a computers microphone. Each person recorded 2 low pitched, 2 medium pitched, 2 high pitched and 2 freely pitched samples of each song.

The recordings were saved in 22 050 Hz 16 bit mono wav files, one for each example.