Project 4

Richard Affolter

21 3 2021

Problem 8: Estimating match emission probabilities

Table 1 shows the number of observations of the symbol a at position i across all sequences in the multiple alignment$Ei(a)$ of our data.

Table 1: $E_i(a)$

	1	2	3
A	4	0	0
C	0	0	5
G	0	4	0
T	0	0	0

With this we can calculate the match emission probabilities $e_i(a)$ by adding a pseudo-count and dividing each column by its column-sum. The resulting table is shown below.

Table 2: $e_i(a)$

	1	2	3
A	5/8	1/8	1/9
C	1/8	1/8	6/9
G	1/8	5/8	1/9
T	1/8	1/8	1/9

Problem 9: Estimating insert emission probabilities

We can repeat the same procedure as above with the insert states. Again we get the observed inserts $E_i(a)$ and can use them to calculate the estimated insert emission probabilities $e_i(a)$

Table 3: $E_i(a)$

	1	2	3
A	0	5	0
C	0	0	0
G	0	1	0
T	0	0	0

Table 4: $e_i(a)$

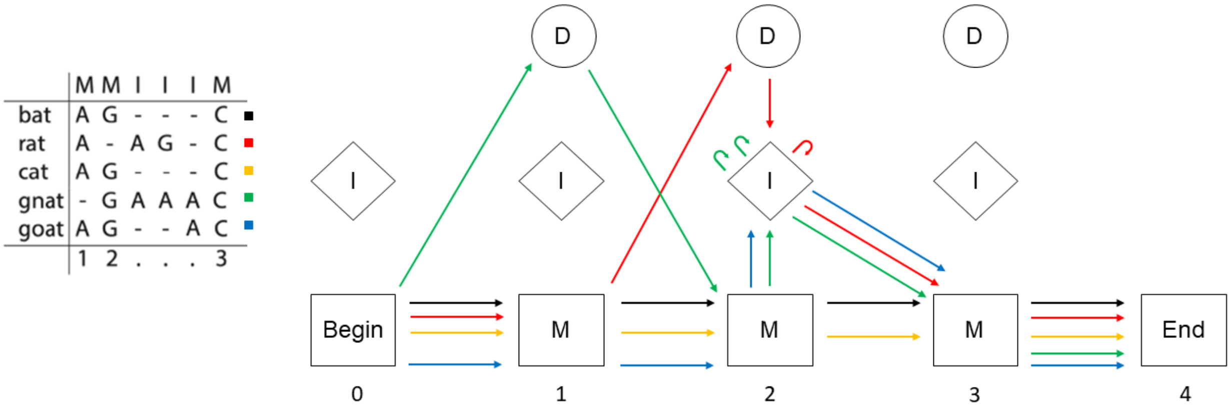
	1	2	3
A	1/4	6/10	1/4
C	1/4	1/10	1/4
G	1/4	2/10	1/4
T	1/4	1/10	1/4

Problem 10: Estimating transition probabilities

The paths of each sequence trough the profile HMM are:

$$\begin{aligned}
\text{bat:} &\quad \text{Begin} \to M_1 \to M_2 \to M_3 \to \text{End}\\
\text{rat:} &\quad \text{Begin} \to M_1 \to D_2 \to I_2 \to I_2 \to M_3 \to \text{End}\\
\text{cat:} &\quad \text{Begin} \to M_1 \to M_2 \to M_3 \to \text{End}\\
\text{gnat:} &\quad \text{Begin} \to D_1 \to M_2 \to I_2 \to I_2 \to I_2 \to M_3 \to \text{End}\\
\text{goat:} &\quad \text{Begin} \to M_1 \to M_2 \to I_2 \to M_3 \to \text{End}
\end{aligned}$$

With this we can draw the following diagram of the paths



	M	M	I	I	I	M	
bat	A	G	-	-	-	C	■
rat	A	-	A	G	-	C	🟥
cat	A	G	-	-	-	C	🟨
gnat	-	G	A	A	A	C	🟩
goat	A	G	-	-	A	C	🟦
	1	2	.	.	.	3	

Figure 1: profile JMM for each sequence

We can use these paths to count $T_i(k \to l)$. The count table is the following

Table 5: $T_i(k \to l)$

	0	1	2	3
$M \to M$	4	3	2	5
$M \to I$	0	0	2	0
$M \to D$	1	1	0	0
$I \to M$	0	0	3	0
$I \to I$	0	0	3	0
$I \to D$	0	0	0	0
$D \to M$	0	1	0	0
$D \to I$	0	0	1	0
$D \to D$	0	0	0	0

With this we can estimate $t_i(k \to l)$ as:

Table 6: $t_i(k \to l)$

	0	1	2	3
$M \to M$	5/8	4/7	3/7	6/8
$M \to I$	1/8	1/7	3/7	1/8
$M \to D$	2/8	2/7	1/7	1/8
$I \to M$	1/3	1/3	4/9	1/3
$I \to I$	1/3	1/3	4/9	1/3
$I \to D$	1/3	1/3	1/9	1/3
$D \to M$	1/3	2/4	1/4	1/3
$D \to I$	1/3	1/4	2/4	1/3
$D \to D$	1/3	1/4	1/4	1/3

Problem 11: Protein family membership classification

1. Run source("profileHMM.R")

```
source("profileHMM.R")
```

2. Read the two alignments 'GTP_binding_proteins.txt' and 'ATPases.txt' into memory using the function parseAlignment().

```
GTP_data = parseAlignment("GTP_binding_proteins.txt")
ATP_data = parseAlignment("ATPases.txt")
```

3. Use the function learnHMM() to parametrise two profile HMMs: one for each protein family (multiple alignment).

```
GTP_profile = learnHMM(GTP_data)
ATP_profile = learnHMM(ATP_data)
```

4. Identify the position(s) with the highest match and with the highest insert emission frequencies over all symbols. Plot the respective match and insert emission frequencies for the identified positions.

```
library(dplyr)
GTP_profile$mE %>% which.max() %>% arrayInd(dim(GTP_profile$mE)) -> GTP_idx
ATP_profile$mE %>% which.max() %>% arrayInd(dim(ATP_profile$mE)) -> ATP_idx

cat(
  "For GTP binding proteins the pos. with the highest match emission frequency is position",
  colnames(GTP_profile$mE)[GTP_idx[2]], "\n (with highest frequency of ",
  GTP_profile$mE[GTP_idx], ")\n")
```

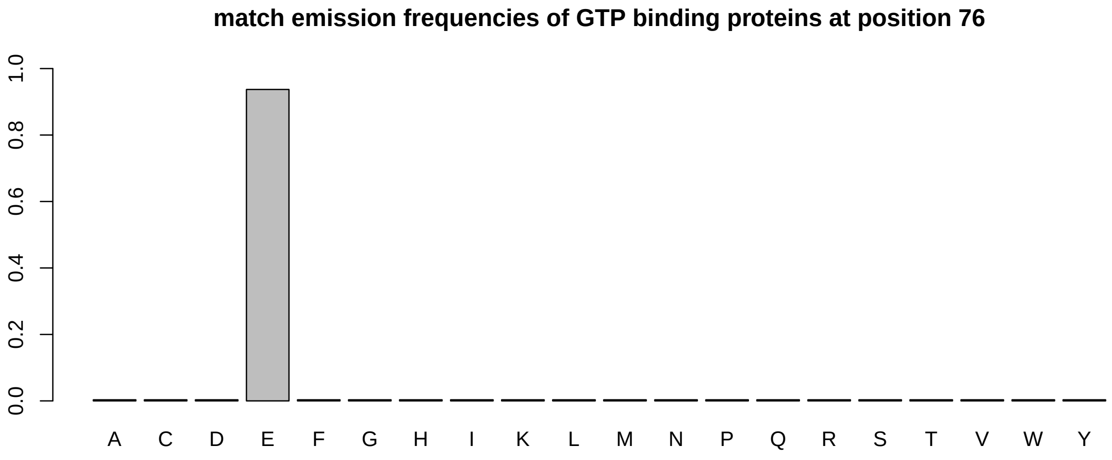
match frequencies:

```
## For GTP binding proteins the pos. with the highest match emission frequency is position 76
##  (with highest frequency of  0.9370861 )
```

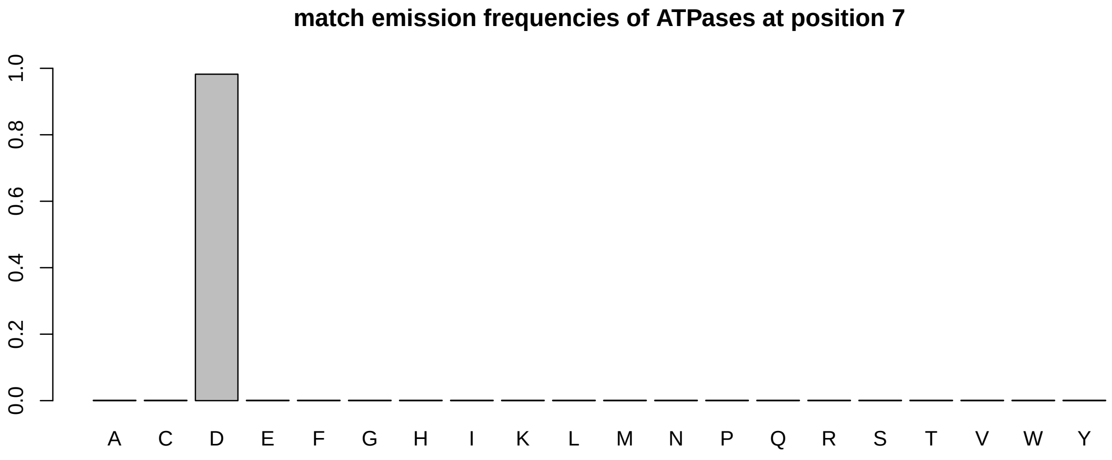
```
cat("For ATPases the pos. with the highest match emission frequency is position",
    colnames(ATP_profile$mE)[ATP_idx[2]], "\n (with highest frequency of ",
    ATP_profile$mE[ATP_idx], ")\n")
```

```
## For ATPases the pos. with the highest match emission frequency is position 7
##  (with highest frequency of  0.9823091 )
```

```
barplot(GTP_profile$mE[,GTP_idx[2]], ylim = c(0,1),
        main = "match emission frequencies of GTP binding proteins at position 76")
```



```
barplot(ATP_profile$mE[,ATP_idx[2]], ylim = c(0,1),
        main = "match emission frequencies of ATPases at position 7")
```



```
library(dplyr)
GTP_profile$iE %>% which.max() %>% arrayInd(dim(GTP_profile$iE)) -> GTP_idx
ATP_profile$iE %>% which.max() %>% arrayInd(dim(ATP_profile$iE)) -> ATP_idx

cat(
  "For GTP binding proteins the pos. with the highest insert emission frequency is position",
  colnames(GTP_profile$mE)[GTP_idx[2]], "\n (with highest frequency of ",
  GTP_profile$iE[GTP_idx], ")\n")
```

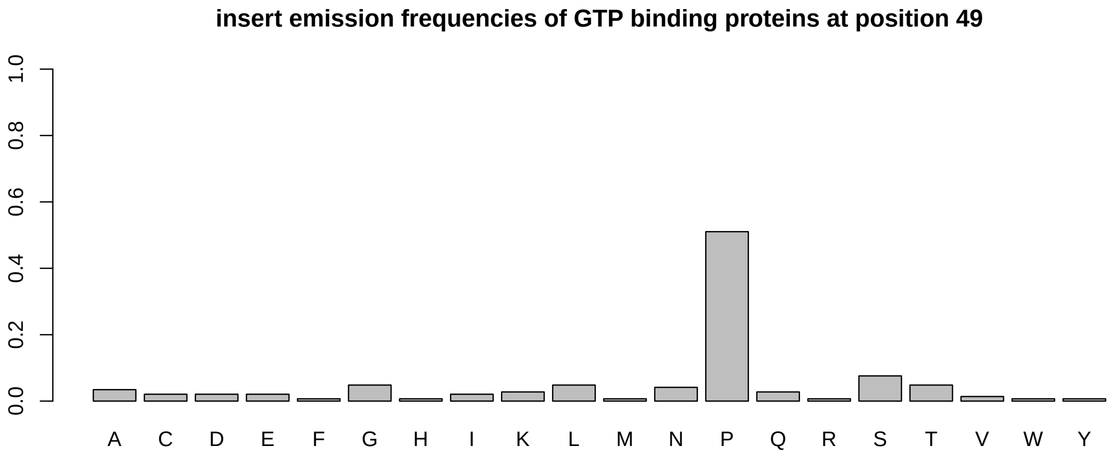
insert frequencies:

```
## For GTP binding proteins the pos. with the highest insert emission frequency is position 49
##  (with highest frequency of  0.5103448 )
```

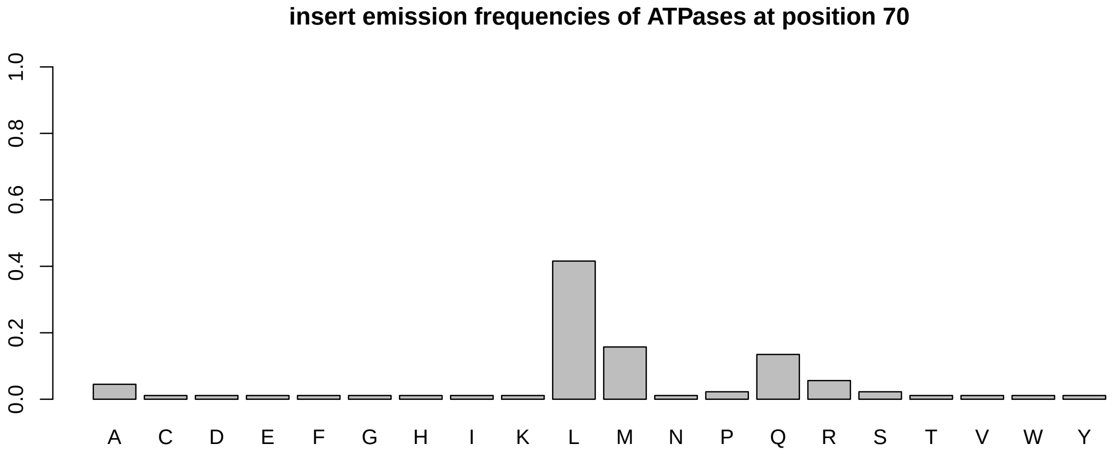
```
cat("For ATPases the pos. with the highest insert emission frequency is position",
    colnames(ATP_profile$mE)[ATP_idx[2]], "\n (with highest frequency of ",
    ATP_profile$iE[ATP_idx], ")\n")
```

```
## For ATPases the pos. with the highest insert emission frequency is position 70
##  (with highest frequency of  0.4157303 )
```

```
barplot(GTP_profile$iE[,GTP_idx[2]], ylim = c(0,1),
        main = "insert emission frequencies of GTP binding proteins at position 49")
```



```
barplot(ATP_profile$iE[,ATP_idx[2]], ylim = c(0,1),
        main = "insert emission frequencies of ATPases at position 70")
```



5. The file Unclassified proteins.txt contains 31 protein sequences from unknown families. Load the protein sequences into a list using the `parseProteins()` function.

```
Unclassified_data = parseProteins("Unclassified_proteins.txt")
```

6. The function forward() takes as input a profile HMM $\mathcal{M}$ and a sequence x. It returns the log odds ratio

$$\log \frac{P(x \mid \mathcal{M})}{P(x \mid \mathcal{R})}$$

of the probability of observing the sequence x given the model $\mathcal{M}$ versus the probability of observing the sequence x given the random model $\mathcal{R}$. For each unclassified protein $x^{(i)}$ in the list, apply the forward algorithm for both models M_1 and M_2 to obtain the log odds ratio

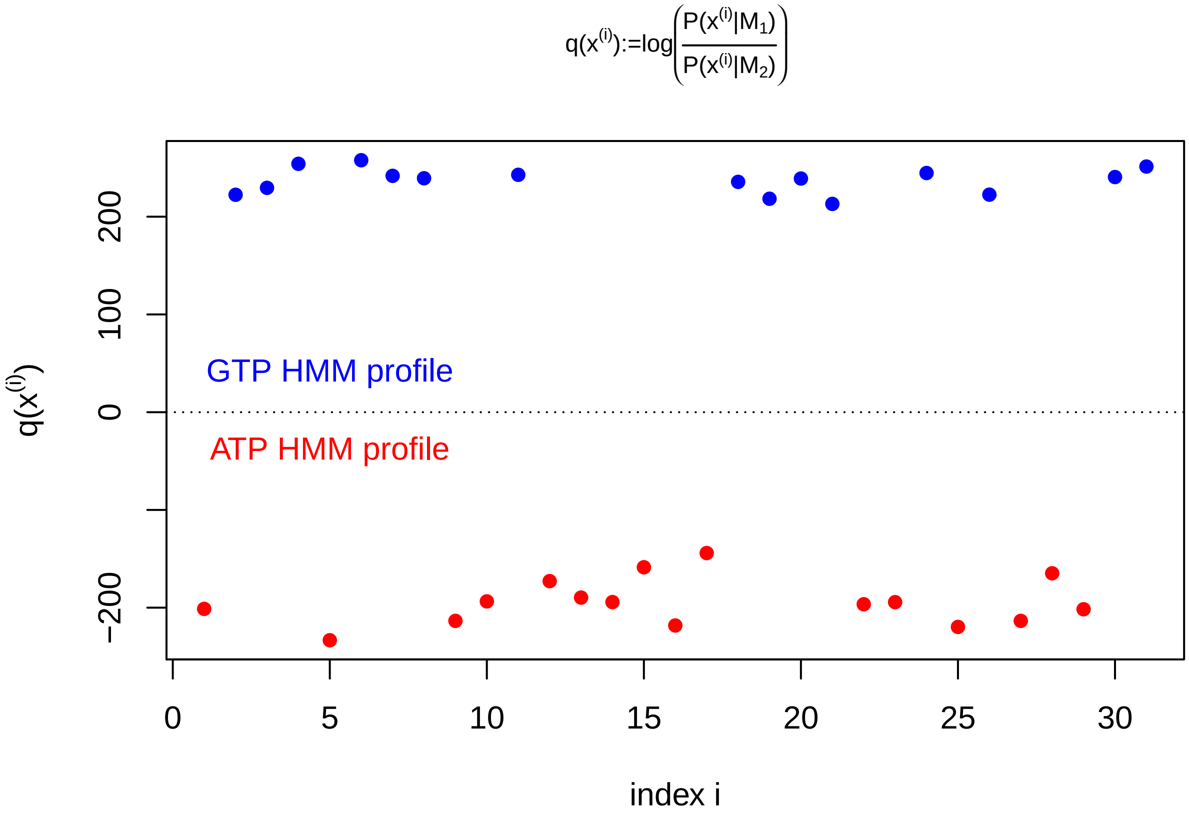
$$q(x^{(i)}) := \log \left(\frac{P(x^{(i)} \mid M_1)}{P(x^{(i)} \mid M_2)} \right)$$

Plot the values $q(x^i)$ and include this in your report. Which proteins in the list belong to which family? Can you clearly decide for each protein?

```
lapply(Unclassified_data, forward, HMM = GTP_profile) %>% unlist() -> GTP_log
lapply(Unclassified_data, forward, HMM = ATP_profile) %>% unlist() -> ATP_log
q = GTP_log - ATP_log
names(q) = 1:length(q)
print(q)
```

```
##         1         2         3         4         5         6         7         8
## -201.2764  222.4483  229.5424  254.1149 -233.3239  257.7991  241.8135  239.3519
##         9        10        11        12        13        14        15        16
## -213.5402 -193.5920  242.8638 -172.8615 -189.6986 -194.3196 -158.6686 -218.2638
##        17        18        19        20        21        22        23        24
## -144.1315  235.6511  218.3957  239.0511  213.1032 -196.5237 -194.3479  244.6956
##         25          26          27          28          29          30          31
## -219.7358  222.5847 -213.5467 -164.8467 -201.6350  240.5883  251.3384
```

```
library(latex2exp)
ylab_string = TeX(r"($q(x^{(i)}))")
q_string =
  TeX(r"($q(x^{(i)}):= \log \left(\frac{P(x^{(i)} | M_1)}{P(x^{(i)} | M_2)} \right)$)")
par(mar = c(4, 5, 5, 1))
plot(q, xlab = "index i",ylab = ylab_string, main = q_string, pch = 16,
     col = ifelse(q < 0,'red','blue'), cex.main = 0.75)
abline(h = 0, lty = 3)
text(5, 40, labels = "GTP HMM profile", col = "blue")
text(5, -40, labels = "ATP HMM profile", col = "red")
```



We can see in the plot that we have a nice linear separation between the two models. Sequences with a positive q have a higher probability to belong to the GTP binding protein family than to belong to the ATPase family. For sequences with a negative q it is the other way around. Since for no sequence q is close to zero we can clearly decide for each protein.