Project 4 Membereship detection with profile HMMs

Team K - Minghang Li

March 23, 2023

Problem 8: Estimating match emission probabilities

The count $E_i(a)$ and insert emission probability $e_i(a)$, $a \in \mathcal{A} = \{A, C, G, T\}$ are:

pos	$E_i(a)$	$e_i(a)$
1	$E_1(A) = 4 + 1 = \mathbf{5}$	$e_1(A) = \frac{4+1}{4+4} = \frac{\mathbf{5}}{\mathbf{8}}$
	$E_1(C) = 0 + 1 = \mathbf{1}$	$e_1(C) = \frac{0+1}{4+4} = \frac{\mathbf{1}}{\mathbf{8}}$
	$E_1(G) = 0 + 1 = \mathbf{1}$	$e_1(G) = \frac{0+1}{4+4} = \frac{\mathbf{1}}{\mathbf{8}}$
	$E_1(T) = 0 + 1 = \mathbf{1}$	$e_1(T) = \frac{0+1}{4+4} = \frac{\mathbf{1}}{\mathbf{8}}$
2	$E_2(A) = 0 + 1 = \mathbf{1}$	$e_2(A) = \frac{0+1}{4+4} = \frac{\mathbf{1}}{\mathbf{8}}$
	$E_2(C) = 0 + 1 = \mathbf{1}$	$e_2(C) = \frac{0+1}{4+4} = \frac{\mathbf{1}}{\mathbf{8}}$
	$E_2(G) = 4 + 1 = \mathbf{5}$	$e_2(G) = \frac{4+1}{4+4} = \frac{\mathbf{5}}{\mathbf{8}}$
	$E_2(T) = 0 + 1 = \mathbf{1}$	$e_2(T) = \frac{0+1}{4+4} = \frac{\mathbf{1}}{\mathbf{8}}$
3	$E_3(A) = 0 + 1 = \mathbf{1}$	$e_3(A) = \frac{0+1}{5+4} = \frac{\mathbf{1}}{\mathbf{9}}$
	$E_3(C) = 5 + 1 = \mathbf{6}$	$e_3(C) = \frac{5+1}{5+4} = \frac{\mathbf{2}}{\mathbf{3}}$
	$E_3(G) = 0 + 1 = \mathbf{1}$	$e_3(G) = \frac{0+1}{5+4} = \frac{\mathbf{1}}{\mathbf{9}}$
	$E_3(T) = 0 + 1 = \mathbf{1}$	$e_3(T) = \frac{0+1}{5+4} = \frac{\mathbf{1}}{\mathbf{9}}$

Problem 9: Estimating insert emission probabilities

Since the contiguous insert states have the same position in the model, the count $E_i(a)$ and insert emission probability $e_i(a)$, $a \in \mathcal{A} = \{A, C, G, T\}$ are:

$E(a)$	$e(a)$
$E(A) = 5 + 1 = \mathbf{6}$	$e(A) = \frac{5+1}{5+1+4} = \frac{\mathbf{3}}{\mathbf{5}}$
$E(C) = 0 + 1 = \mathbf{1}$	$e(C) = \frac{0+1}{5+1+4} = \frac{\mathbf{1}}{\mathbf{10}}$
$E(G) = 1 + 1 = \mathbf{2}$	$e(G) = \frac{1+1}{5+1+4} = \frac{\mathbf{1}}{\mathbf{5}}$
$E(T) = 0 + 1 = \mathbf{1}$	$e(T) = \frac{0+1}{5+1+4} = \frac{\mathbf{1}}{\mathbf{10}}$

Problem 10: Estimating transition probabilities

The path of each sequence (bat, rat, cat, gnat, goat) from Begin to end is summarized below,

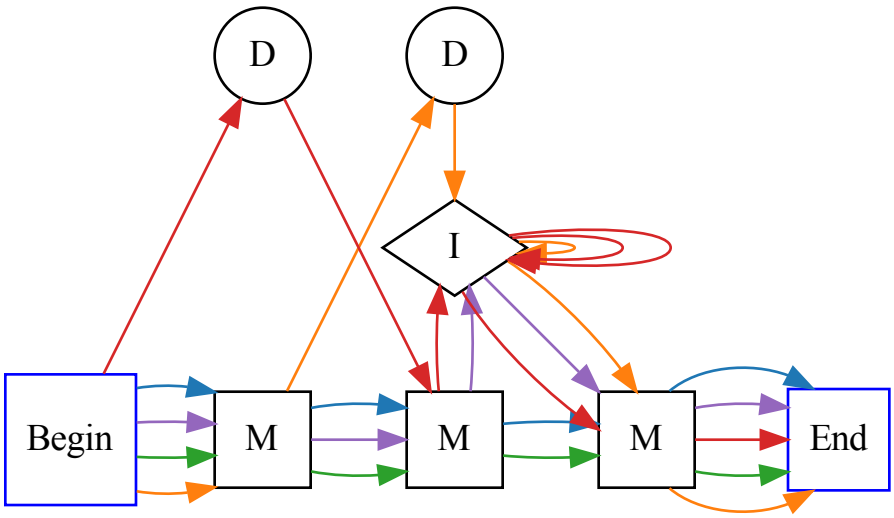


Figure 1: HMM.pdf

Nodes that are not visited in any path have uniform probability of transitting to its next states:

Node	$t_i(k \to l)$
I_0	$\frac{1}{3}$
I_1	$\frac{1}{3}$
I_3	$\frac{1}{2}$
D_3	$\frac{1}{2}$

And the transition probabilities of nodes that are visited are listed in the following table

pos	$T_i(a)$	$e_i(a)$
0	$T_0(M \to M) = 4 + 1 = \mathbf{5}$	$t_0(M \to M) = \frac{4+1}{5+1+2} = \frac{5}{8}$
	$T_0(M \to I) = 0 + 1 = \mathbf{1}$	$t_0(M \to I) = \frac{0+1}{5+1+2} = \frac{1}{8}$
	$T_0(M \to D) = 1 + 1 = 2$	$t_0(M \to D) = \frac{0+1}{5+1+2} = \frac{1}{8}$
1	$T_1(M \to M) = 3 + 1 = \mathbf{4}$	$t_1(M \to M) = \frac{3+1}{4+1+2} = \frac{4}{7}$
	$T_1(M \to I) = 0 + 1 = \mathbf{1}$	$t_1(M \to I) = \frac{0+1}{4+1+2} = \frac{1}{7}$
	$T_1(M \to D) = 1 + 1 = 2$	$t_1(M \to D) = \frac{0+1}{4+1+2} = \frac{1}{7}$
	$T_1(D \to M) = 1 + 1 = 2$	$t_1(D \to M) = \frac{1+1}{2+1+1} = \frac{2}{4}$
	$T_1(D \to I) = 0 + 1 = \mathbf{1}$	$t_1(D \to I) = \frac{0+1}{2+1+1} = \frac{1}{4}$
	$T_1(D \to D) = 0 + 1 = \mathbf{1}$	$t_1(D \to D) = \frac{0+1}{2+1+1} = \frac{1}{4}$
2	$T_2(M \to M) = 2 + 1 = \mathbf{3}$	$t_2(M \to M) = \frac{2+1}{3+3+1} = \frac{3}{7}$
	$T_2(M \to I) = 2 + 1 = \mathbf{3}$	$t_2(M \to I) = \frac{2+1}{3+3+1} = \frac{3}{7}$
	$T_2(M \to D) = 0 + 1 = \mathbf{1}$	$t_2(M \to D) = \frac{0+1}{3+3+1} = \frac{1}{7}$
	$T_2(I \to M) = 3 + 1 = \mathbf{4}$	$t_2(I \to M) = \frac{3+1}{4+4+1} = \frac{4}{9}$
	$T_2(I \to I) = 3 + 1 = \mathbf{4}$	$t_2(I \to I) = \frac{3+1}{4+4+1} = \frac{4}{9}$
	$T_2(I \to D) = 0 + 1 = \mathbf{1}$	$t_2(I \to D) = \frac{0+1}{4+4+1} = \frac{1}{9}$
	$T_2(D \to M) = 0 + 1 = \mathbf{1}$	$t_2(D \to M) = \frac{0+1}{1+2+1} = \frac{1}{4}$
	$T_2(D \to I) = 1 + 1 = 2$	$t_2(D \to I) = \frac{1+1}{1+2+1} = \frac{2}{4}$
	$T_2(D \to D) = 0 + 1 = \mathbf{1}$	$t_2(D \to D) = \frac{0+1}{1+2+1} = \frac{1}{4}$
3	$T_3(M \to M) = 5 + 1 = \mathbf{6}$	$t_3(M \to M) = \frac{5+1}{6+1} = \frac{6}{7}$
	$T_3(M \to I) = 0 + 1 = \mathbf{1}$	$t_3(M \to I) = \frac{0+1}{6+1} = \frac{1}{7}$

All the transition probabilities are summarized in Figure 2.

Problem 11: Protein family membership classification

Import functions and read alignments

```
# import functions
source("code/profileHMM.R", local = knitr::knit_global())
```

```
# read alignments
GTPase <- parseAlignment("./data/GTP_binding_proteins.txt")
ATPase <- parseAlignment("./data/ATPases.txt")
```

Learn HMM from two protein families

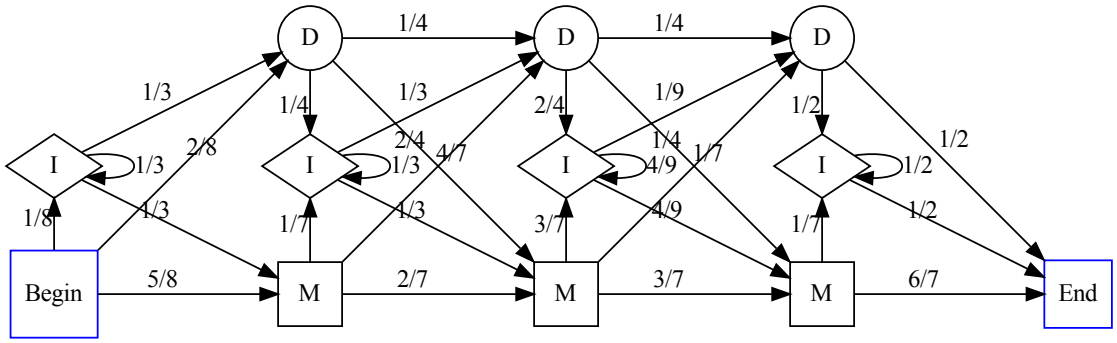


Figure 2: HMM profile

```
HMM_GTPase <- learnHMM(GTPase)
HMM_ATPase <- learnHMM(ATPase)
```

Identify position(s) with the highest match and insert emission frequencies over all symbols.

```
alphabet <- c("A","C","D","E","F","G","H","I","K","L","M", "N","P","Q","R","S","T","V","W","Y")
```

ATPase

```
mE_max_ATPase <- which(HMM_ATPase$mE == max(HMM_ATPase$mE, na.rm = TRUE), arr.ind = TRUE)
iE_max_ATPase <- which(HMM_ATPase$iE == max(HMM_ATPase$iE, na.rm = TRUE), arr.ind = TRUE)
print(paste("The position(s) with the highest match emission frequency over all symbols is/are:",
        mE_max_ATPase[2]))
```

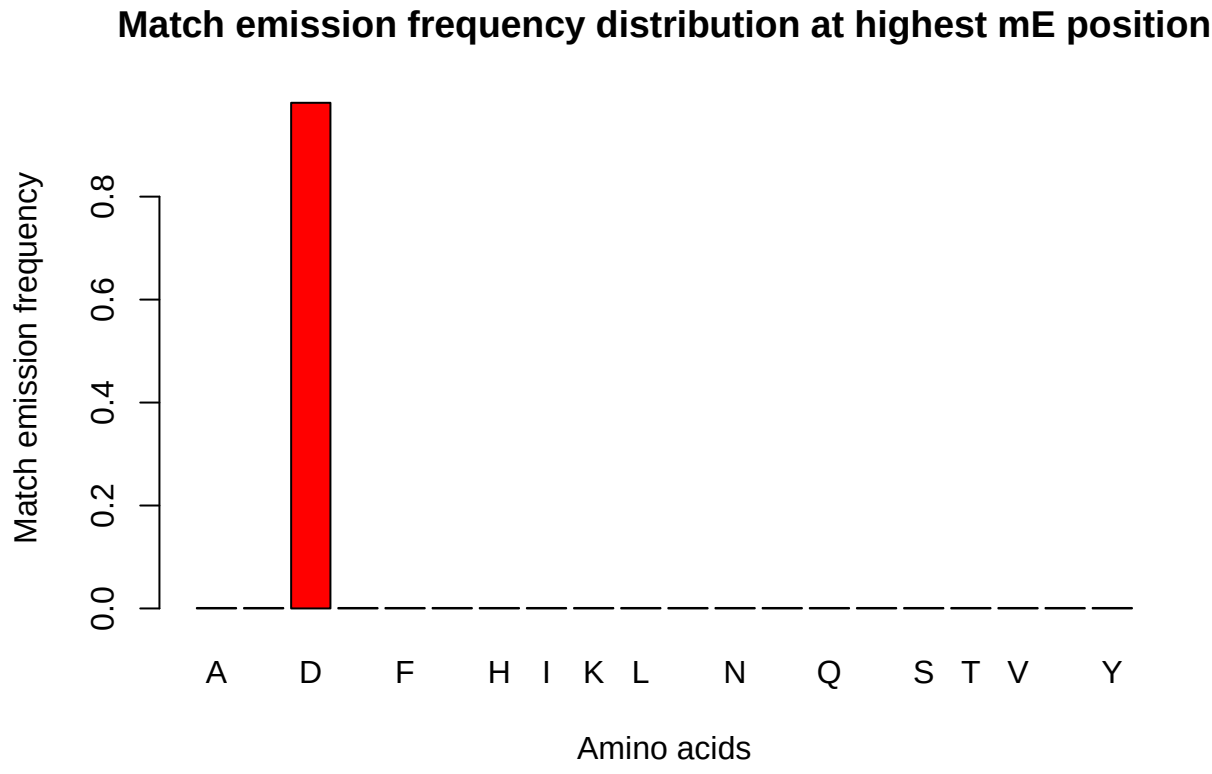
```
## [1] "The position(s) with the highest match emission frequency over all symbols is/are: 8"
```

```
print(paste("The position(s) with the highest insert emission frequency over all symbols is/are:",
        iE_max_ATPase[2]))
```

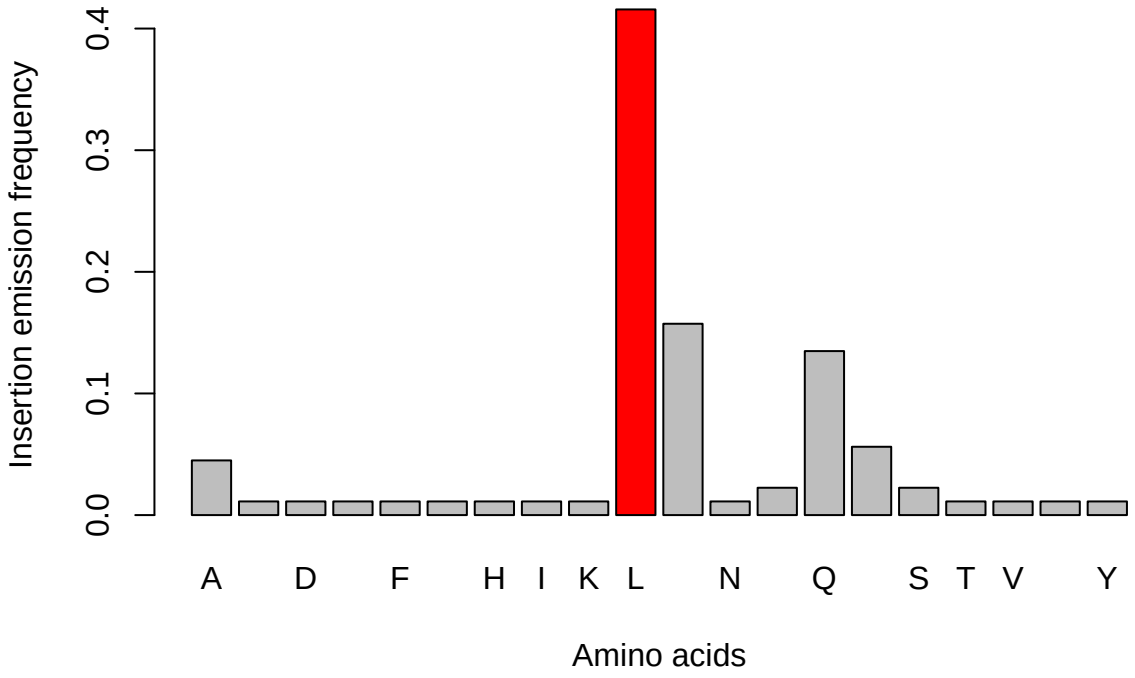
```
## [1] "The position(s) with the highest insert emission frequency over all symbols is/are: 71"
```

```
# generate barplot color sequence c() with length of alphabet
# all colors are gray except for position at mE_max_ATPase[1] is red
color_mE_max_ATPase <- c(rep("gray", length(alphabet)))
color_mE_max_ATPase[mE_max_ATPase[1]] <- "red"
barplot(HMM_ATPase$mE[, mE_max_ATPase[2]],
     col = color_mE_max_ATPase,
```

```
     names.arg = alphabet,
     main = "Match emission frequency distribution at highest mE position",
     xlab = "Amino acids",
     ylab = "Match emission frequency")
```




```
# generate barplot color sequence c() with length of alphabet
# all colors are gray except for position at mE_max_ATPase[1] is red
color_iE_max_ATPase <- c(rep("gray", length(alphabet)))
color_iE_max_ATPase[iE_max_ATPase[1]] <- "red"
barplot(HMM_ATPase$iE[, iE_max_ATPase[2]],
     col = color_iE_max_ATPase,
     names.arg = alphabet,
     main = "Insertion emission frequency distribution at highest iE position",
     xlab = "Amino acids",
     ylab = "Insertion emission frequency")
```



GTPase

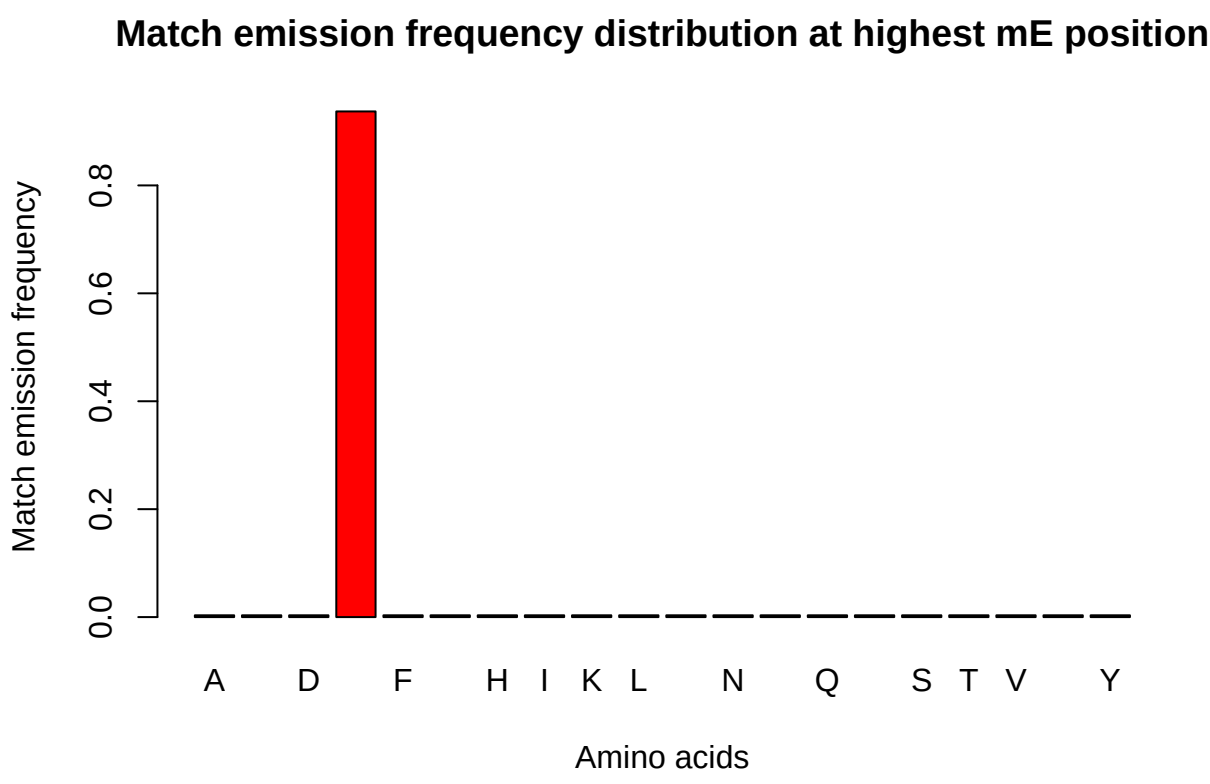
```
mE_max_GTPase <- which(HMM_GTPase$mE == max(HMM_GTPase$mE, na.rm = TRUE), arr.ind = TRUE)
iE_max_GTPase <- which(HMM_GTPase$iE == max(HMM_GTPase$iE, na.rm = TRUE), arr.ind = TRUE)
print(paste("The position(s) with the highest match emission frequency over all symbols is/are:",
        mE_max_GTPase[2]))
```

```
## [1] "The position(s) with the highest match emission frequency over all symbols is/are: 77"
```

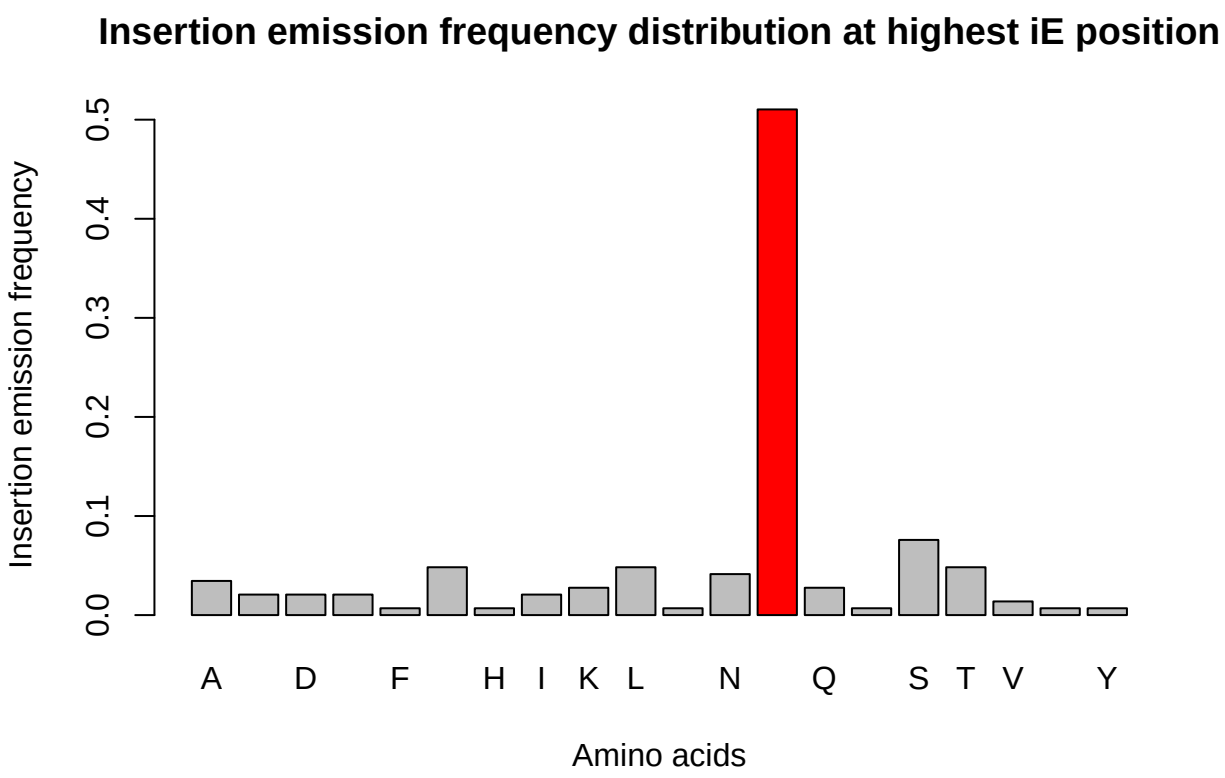
```
print(paste("The position(s) with the highest insert emission frequency over all symbols is/are:",
        iE_max_GTPase[2]))
```

```
## [1] "The position(s) with the highest insert emission frequency over all symbols is/are: 50"
```

```
# generate barplot color sequence c() with length of alphabet
# all colors are gray except for position at mE_max_ATPase[1] is red
color_mE_max_GTPase <- c(rep("gray", length(alphabet)))
color_mE_max_GTPase[mE_max_GTPase[1]] <- "red"
barplot(HMM_GTPase$mE[, mE_max_GTPase[2]],
     col = color_mE_max_GTPase,
     names.arg = alphabet,
     main = "Match emission frequency distribution at highest mE position",
     xlab = "Amino acids",
     ylab = "Match emission frequency")
```




```r
# generate barplot color sequence c() with length of alphabet
# all colors are gray except for position at mE_max_ATPase[1] is red
color_iE_max_GTPase <- c(rep("gray", length(alphabet)))
color_iE_max_GTPase[iE_max_GTPase[1]] <- "red"
barplot(HMM_GTPase$iE[, iE_max_GTPase[2]],
     col = color_iE_max_GTPase,
     names.arg = alphabet,
     main = "Insertion emission frequency distribution at highest iE position",
     xlab = "Amino acids",
     ylab = "Insertion emission frequency")
```

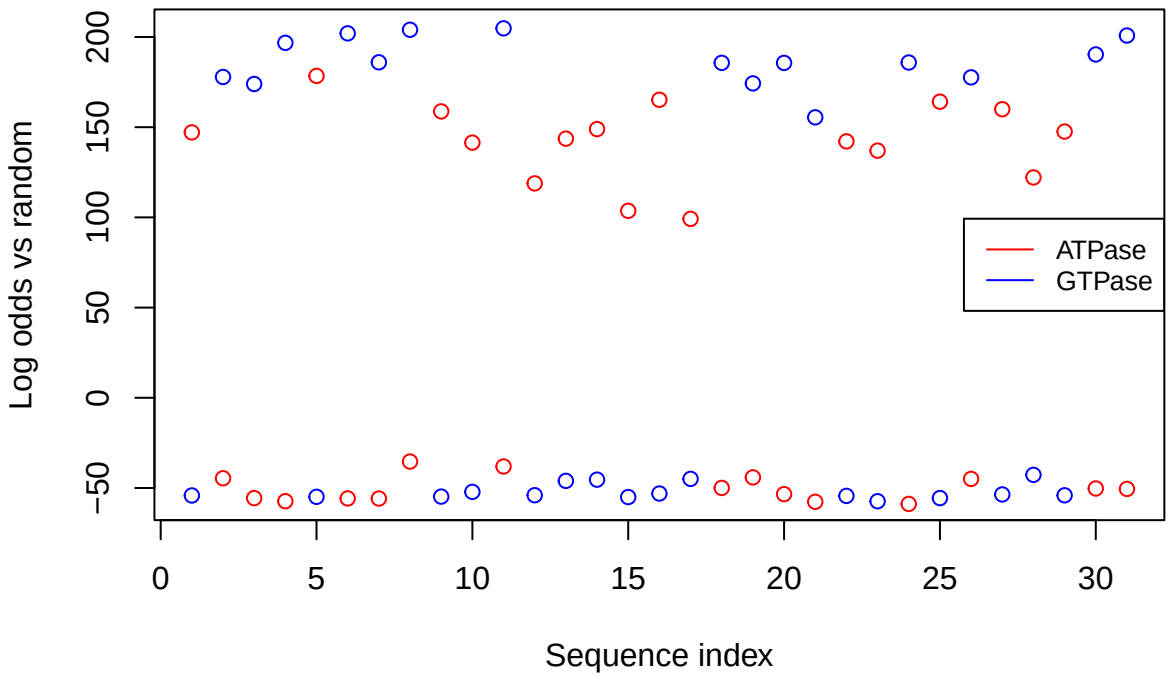


Classify unclassified proteins

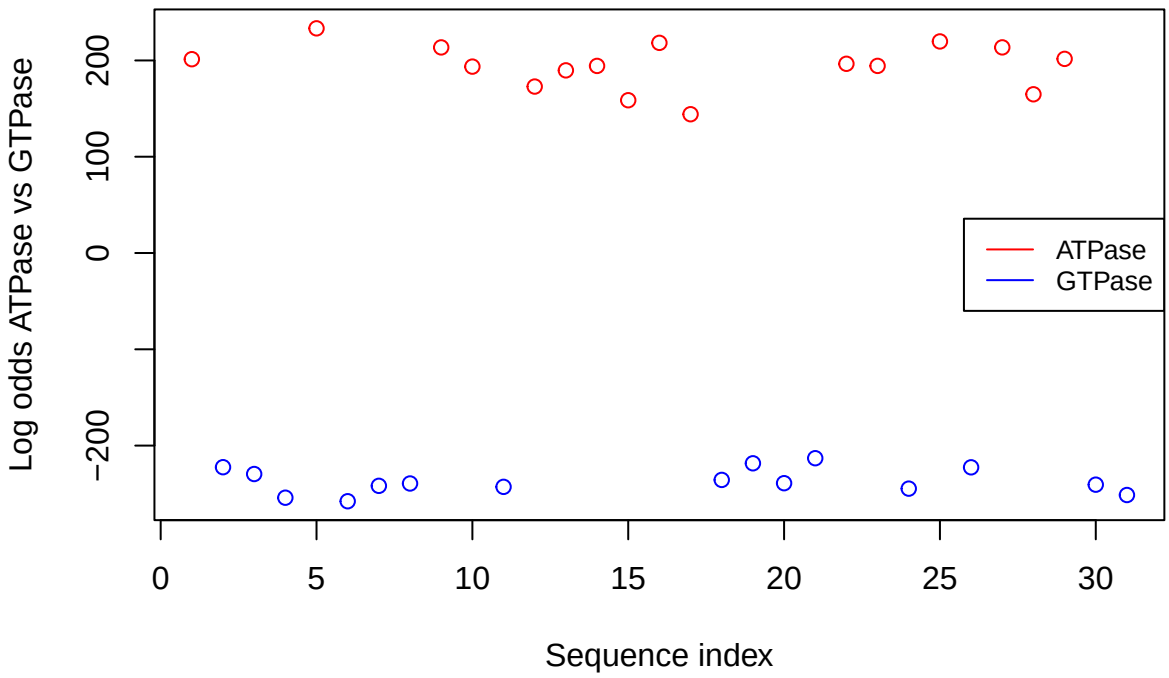
```
unclassified_proteins <- parseProteins(proteinsFile = "./data/Unclassified_proteins.txt")
```

```
num_cores <- detectCores()
registerDoParallel(num_cores)
log_odds_ATPase <- foreach(seq=unclassified_proteins, .combine = c) %dopar% {
  forward(HMM = HMM_ATPase, seq = seq)
}
log_odds_GTPase <- foreach(seq=unclassified_proteins, .combine = c) %dopar% {
  forward(HMM = HMM_GTPase, seq = seq)
}
stopImplicitCluster()
```

```
# plot the log odds ratio of ATPase (red) and GTPase (blue) on the same plot
plot(log_odds_GTPase, col="blue",
    xlab = "Sequence index",
    ylab = "Log odds vs random")
points(log_odds_ATPase, col="red")
# add legend
legend("right", legend=c("ATPase", "GTPase"), col=c("red", "blue"), lty=1, cex=0.8)
```




```r
q <- log_odds_ATPase - log_odds_GTPase
plot(q, col=ifelse(q >= 0, "red", "blue"),
    xlab = "Sequence index",
    ylab = "Log odds ATPase vs GTPase")
legend("right", legend=c("ATPase", "GTPase"), col=c("red", "blue"), lty=1, cex=0.8)
```



We can see a clear separation between points that are classified as ATPases and those classified as GTPases. The log ratio of ATPase vs GTPase is always way larger than 0 or way smaller than 0, indicating high likelihood.

```
# print the index of proteins that are classified as ATPases
print("The index of proteins that are classified as ATPases are:")
```

```
## [1] "The index of proteins that are classified as ATPases are:"
```

```
print(which(q >= 0))
```

```
## MM MM MM MM MM MM MM MM MM MM MM MM MM MM MM MM
##  1  5  9 10 12 13 14 15 16 17 22 23 25 27 28 29
```

```
# print the index of proteins that are classified as GTPases
print("The index of proteins that are classified as GTPases are:")
```

```
## [1] "The index of proteins that are classified as GTPases are:"
```

```
print(which(q < 0))
```

```
## MM MM MM MM MM MM MM MM MM MM MM MM MM MM MM
##  2  3  4  6  7  8 11 18 19 20 21 24 26 30 31
```