Computer Assignment 3

781552

10/1/2021

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

##Question 1 #Subtask (a)

```
seqMouse <- read.GenBank("AK080777", as.character = TRUE)[[1]]
seqMouseStr <- paste(seqMouse, sep ="", collapse = "")
DNAMouseStr <- DNAString(seqMouseStr)

seqHuman <- read.GenBank("NM_000520", as.character = TRUE)[[1]]
seqHumanStr <- paste(seqHuman, sep ="", collapse = "")
DNAHumanStr <- DNAString(seqHumanStr)

DNAMouseStr
```

```
## 1839-letter DNAString object
## seq: GCTGCTGGAAGGGGAGCTGGCCGGTGGGCCATGGCC...ACTGGTGTTCAATAAAGATCTATGTGGCATTTTCTC
```

```
DNAHumanStr
```

```
## 4785-letter DNAString object
## seq: CTCACGTGGCCAGCCCCCTCCGAGAGGGGAGACCAG...ATAAATAAACTTTGAAATAAAGGTTGAAAATTAGTA
```

#Subtask (b)

```
findORFs(toupper(DNAMouseStr))
```

```
## IRangesList object of length 1:
## $`1`
## IRanges object with 28 ranges and 0 metadata columns:
##            start       end     width
##        <integer> <integer> <integer>
##    [1]        31      1617      1587
##    [2]      1654      1671        18
##    [3]      1687      1722        36
```

```
##     [4]         2       370       369
##     [5]       392       748       357
##     ...       ...       ...       ...
##    [24]      1317      1367        51
##    [25]      1398      1472        75
##    [26]      1539      1544         6
##    [27]      1587      1595         9
##    [28]      1614      1631        18
```

```
findORFs(toupper(DNAHumanStr))
```

```
## IRangesList object of length 1:
## $`1`
## IRanges object with 111 ranges and 0 metadata columns:
##             start       end     width
##         <integer> <integer> <integer>
##     [1]        43      1632      1590
##     [2]      1669      1974       306
##     [3]      1978      2001        24
##     [4]      2017      2025         9
##     [5]      2077      2313       237
##     ...       ...       ...       ...
##   [107]      4263      4304        42
##   [108]      4308      4445       138
##   [109]      4473      4484        12
##   [110]      4737      4757        21
##   [111]      4761      4769         9
```

#Subtask (c)

```
seqMouseProtein <-  translate(DNAMouseStr)
seqHumanProtein <- translate(DNAHumanStr)

seqMouseProtein
```

```
## 613-letter AAString object
## seq: AAGRGAGRWAMAGCRLWVSLLLAAALACLATALWPW...ASRPGESTPCPCAPVTTEKEAGAGTGVQ*RSMWHFL
```

```
seqHumanProtein
```

```
## 1595-letter AAString object
## seq: LTWPAPSERGDQRAMTSSRLWFSLLLAAAFAGRATA...TFFSTKKKNK*ELGALFIKICISFINKL*NKG*KLV
```

#Subtask (d)

```
a <- pairwiseAlignment(seqHumanProtein,
                       seqMouseProtein,
                       type = "global",
                       substitutionMatrix = "BLOSUM62")

score(a)
```

```
## [1] -1444
```

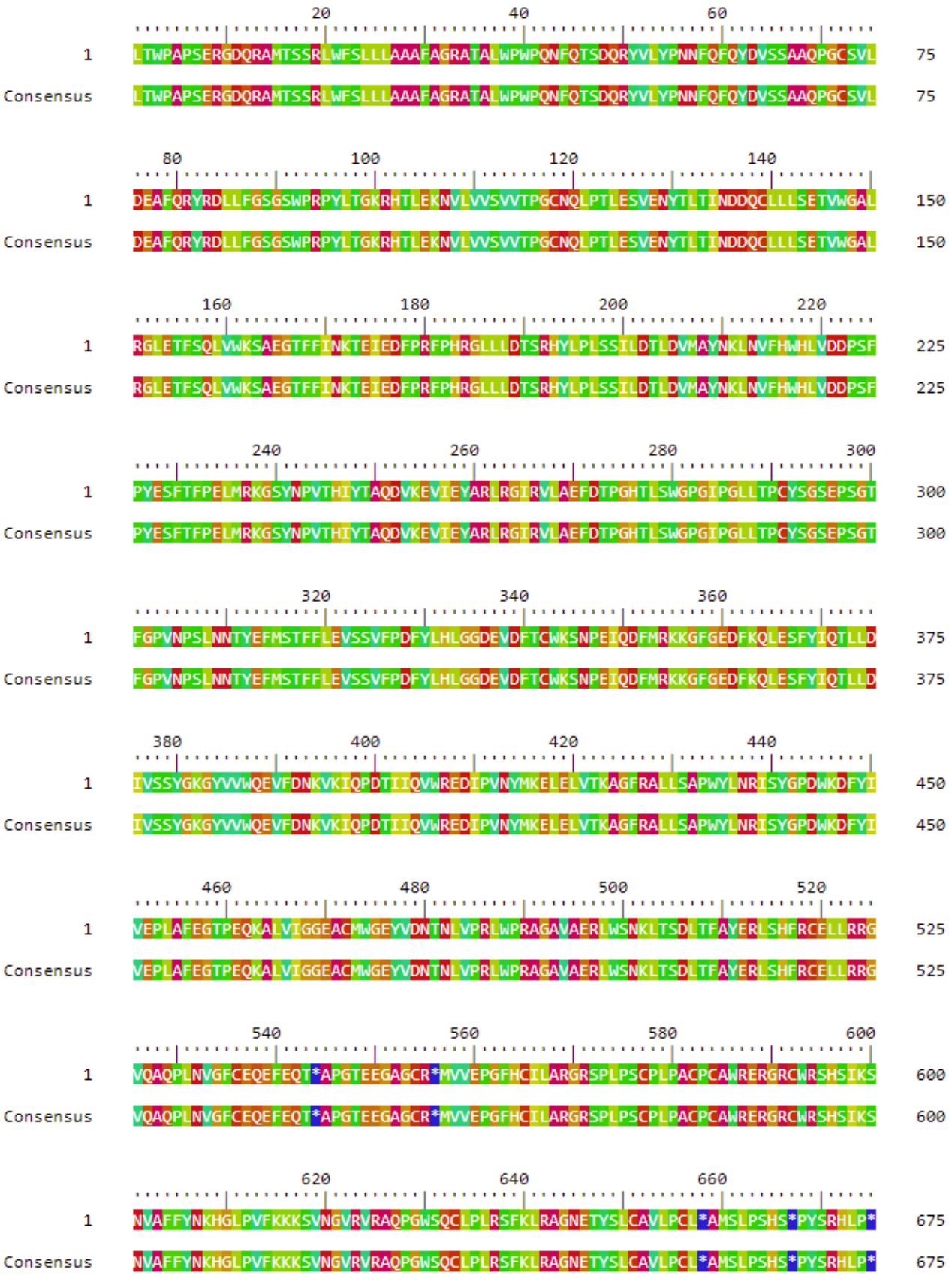
```
alignedPattern(a)
```

```
## AAStringSet object of length 1:
##     width seq
## [1]  1595 LTWPAPSERGDQRAMTSSRLWFSLLLAAAFAGRA...STKKKNK*ELGALFIKICISFINKL*NKG*KLV
```

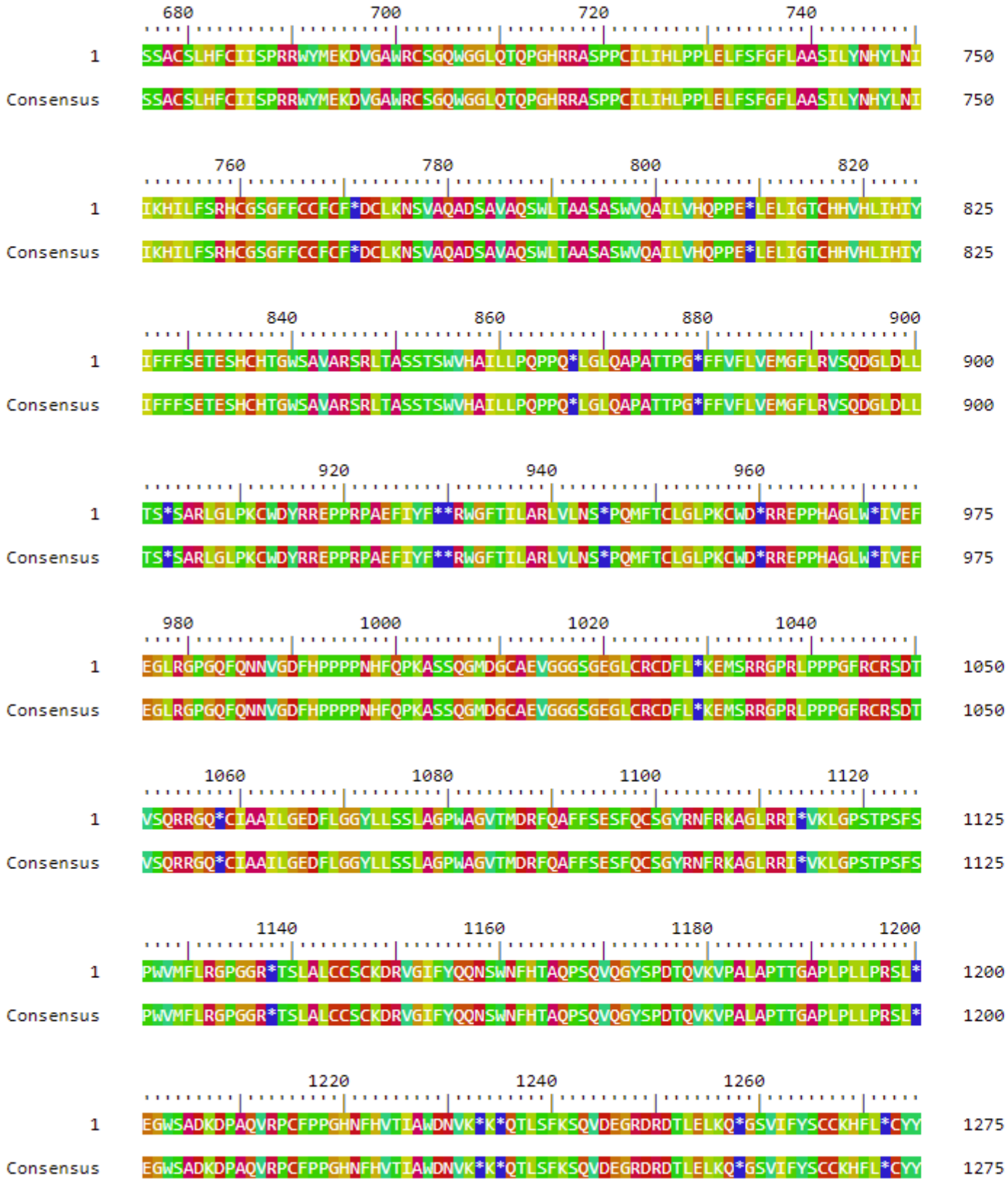
```
alignedSubject(a)
```

```
## AAStringSet object of length 1:
##     width seq
## [1]  1595 ----AAGRGAGRWAMAGCRLWVSLLLAAALACLA...----------------------------------
```

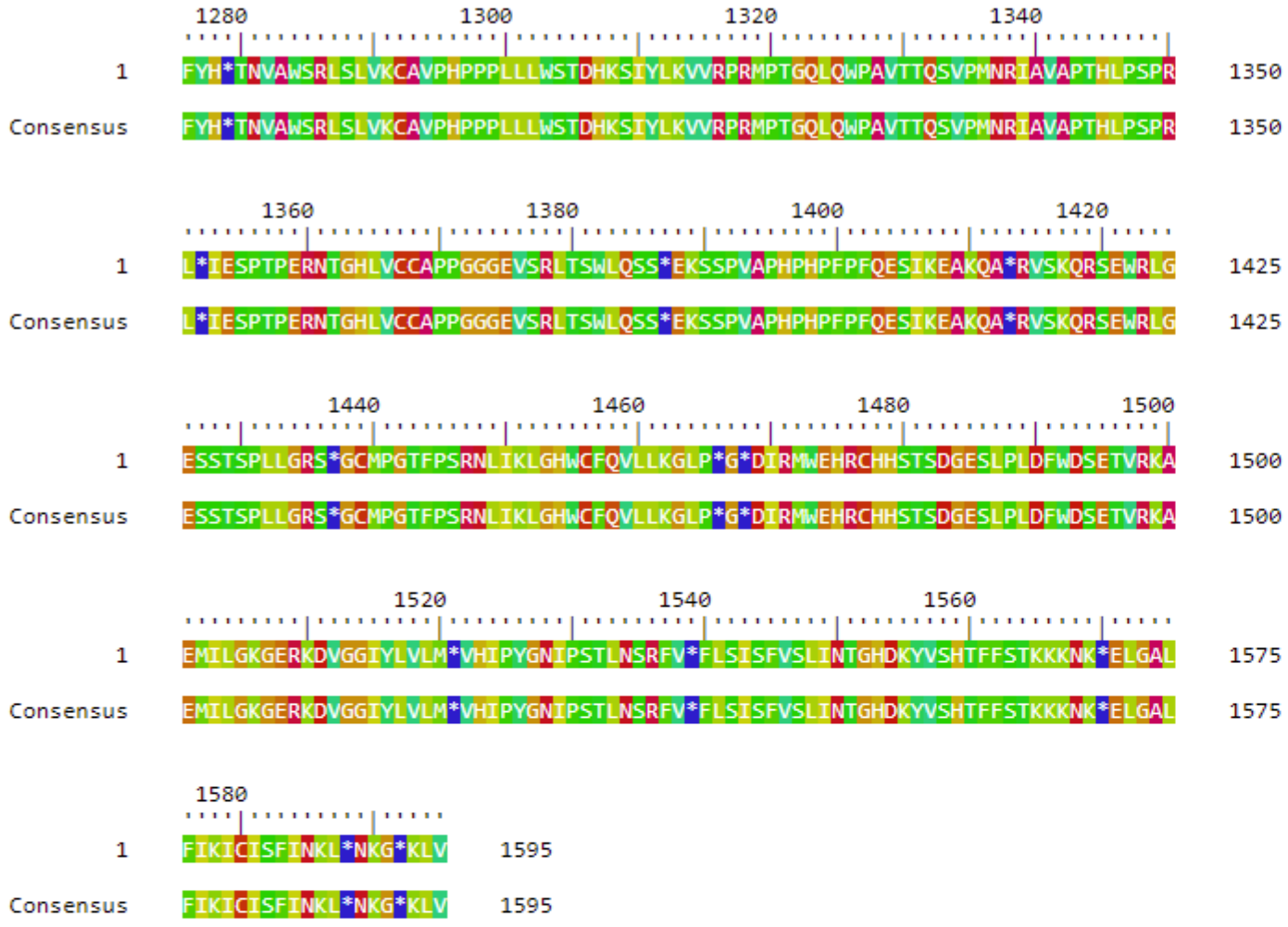
```
b <-
BrowseSeqs(alignedPattern(a), colWidth = 75)

knitr::include_graphics('Pattern1.png')
```


20
40
60
1 LTWPAPSERGDQRAMTSSRLWFSLLLAAAFAGRATALWPWPQNFQTSDQRYVLYPNNFQFQYDVSSAAQPGCSVL 75
Consensus LTWPAPSERGDQRAMTSSRLWFSLLLAAAFAGRATALWPWPQNFQTSDQRYVLYPNNFQFQYDVSSAAQPGCSVL 75
80
100
120
140
1 DEAFQRYRDLLFGSGSWPRPYLTGKRHTLEKNVLVVSVVTPGCNQLPTLESVENYTLTINDDQCLLLSETVWGAL 150
Consensus DEAFQRYRDLLFGSGSWPRPYLTGKRHTLEKNVLVVSVVTPGCNQLPTLESVENYTLTINDDQCLLLSETVWGAL 150
160
180
200
220
1 RGLETFSQLVWKSAEGTFFINKTEIEDFPRFPHRGLLLDTSRHYLPLSSILDTLDVMAYNKLNVFHWHLVDDPSF 225
Consensus RGLETFSQLVWKSAEGTFFINKTEIEDFPRFPHRGLLLDTSRHYLPLSSILDTLDVMAYNKLNVFHWHLVDDPSF 225
240
260
280
300
1 PYESFTFPELMRKGSYNPVTHIYTAQDVKEVIEYARLRGIRVLAEFDTPGHTLSWGPGIPGLLTPCYSGSEPSGT 300
Consensus PYESFTFPELMRKGSYNPVTHIYTAQDVKEVIEYARLRGIRVLAEFDTPGHTLSWGPGIPGLLTPCYSGSEPSGT 300
320
340
360
1 FGPVNPSLNNTYEFMSTFFLEVSSVFPDFYLHLGGDEVDFTCWKSNPEIQDFMRKKGFGEDFKQLESFYIQTLLD 375
Consensus FGPVNPSLNNTYEFMSTFFLEVSSVFPDFYLHLGGDEVDFTCWKSNPEIQDFMRKKGFGEDFKQLESFYIQTLLD 375
380
400
420
440
1 IVSSYGKGYVVWQEVFDNKVKIQPDTIIQVWREDIPVNYMKELELVTKAGFRALLSAPWYLNRISYGPDWKDFYI 450
Consensus IVSSYGKGYVVWQEVFDNKVKIQPDTIIQVWREDIPVNYMKELELVTKAGFRALLSAPWYLNRISYGPDWKDFYI 450
460
480
500
520
1 VEPLAFEGTPEQKALVIGGEACMWGEYVDNTNLVPRLWPRAGAVAERLWSNKLTSDLTFAYERLSHFRCELLRRG 525
Consensus VEPLAFEGTPEQKALVIGGEACMWGEYVDNTNLVPRLWPRAGAVAERLWSNKLTSDLTFAYERLSHFRCELLRRG 525
540
560
580
600
1 VQAQPLNVGFCEQEFEQT*APGTEEGAGCR*MVVEPGFHCILARGRSPLPSCPLPACPCAWRERGRCWRSHSIKS 600
Consensus VQAQPLNVGFCEQEFEQT*APGTEEGAGCR*MVVEPGFHCILARGRSPLPSCPLPACPCAWRERGRCWRSHSIKS 600
620
640
660
1 NVAFFYNKHGLPVFKKKSVNGVRVRAQPGWSQCLPLRSFKLRAGNETYSLCAVLPCL*AMSLPSHS*PYSRHLP* 675
Consensus NVAFFYNKHGLPVFKKKSVNGVRVRAQPGWSQCLPLRSFKLRAGNETYSLCAVLPCL*AMSLPSHS*PYSRHLP* 675

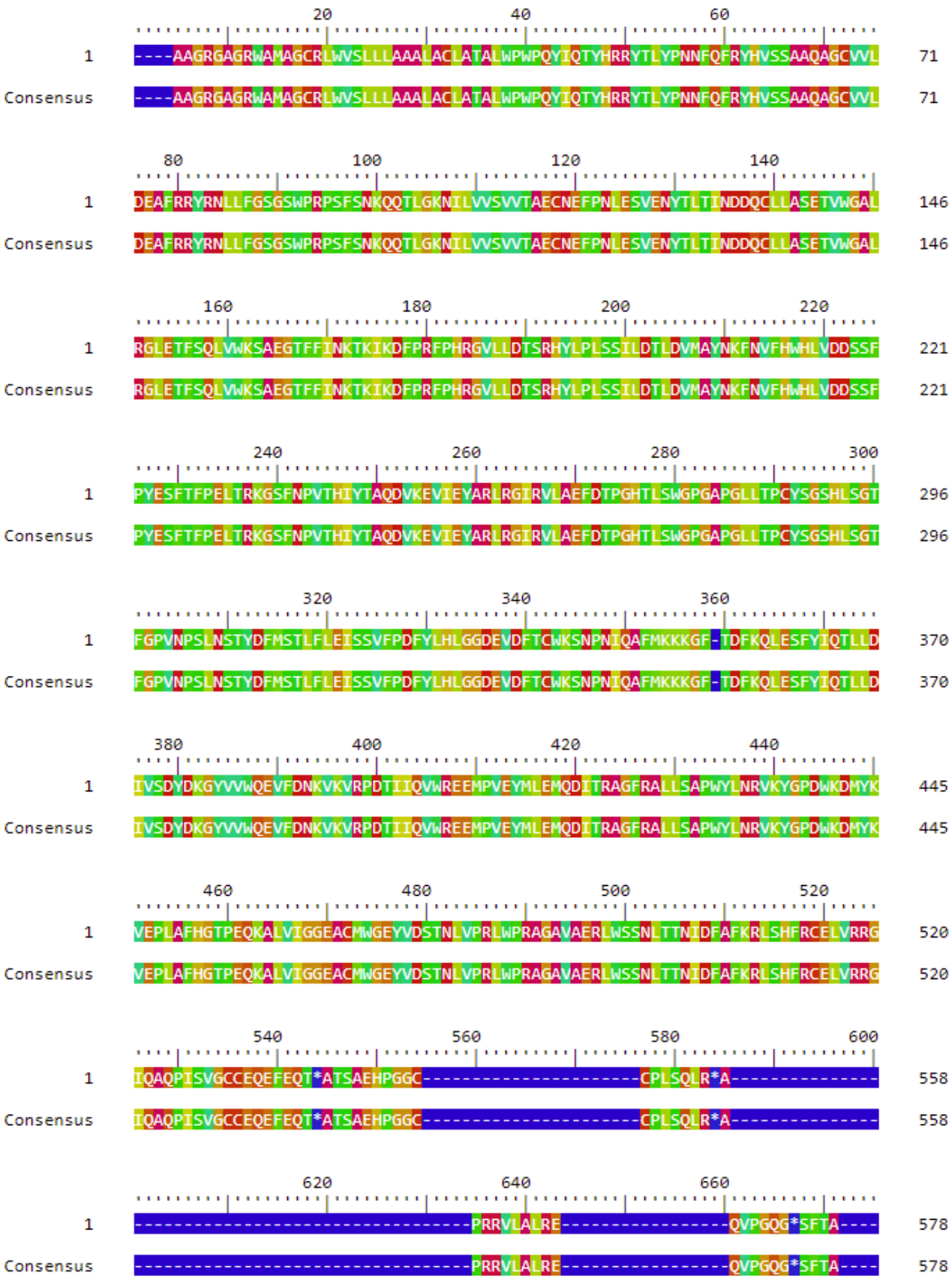
```
knitr::include_graphics('Pattern2.png')
```


680 700 720 740
1 SSACSLHFCIISPRRWYMEKDVGAWRCSGQWGGLQTQPGHRRASPPCILIHLPPLELFSFGFLAASILYNHYLNI 750
Consensus SSACSLHFCIISPRRWYMEKDVGAWRCSGQWGGLQTQPGHRRASPPCILIHLPPLELFSFGFLAASILYNHYLNI 750
760 780 800 820
1 IKHILFSRHCGSGFFCCFCF*DCLKNSVAQADSAVAQSWLTAASASWVQAILVHQPPE*LELIGTCHHVHLIHIY 825
Consensus IKHILFSRHCGSGFFCCFCF*DCLKNSVAQADSAVAQSWLTAASASWVQAILVHQPPE*LELIGTCHHVHLIHIY 825
840 860 880 900
1 IFFFSETESHCHTGWSAVARSRLTASSTSWVHAILLPQPPQ*LGLQAPATTPG*FFVFLVEMGFLRVSQDGLDLL 900
Consensus IFFFSETESHCHTGWSAVARSRLTASSTSWVHAILLPQPPQ*LGLQAPATTPG*FFVFLVEMGFLRVSQDGLDLL 900
920 940 960
1 TS*SARLGLPKCWDYRREPPRPAEFIYF**RWGFTILARLVLNS*PQMFTCLGLPKCWD*RREPPHAGLW*IVEF 975
Consensus TS*SARLGLPKCWDYRREPPRPAEFIYF**RWGFTILARLVLNS*PQMFTCLGLPKCWD*RREPPHAGLW*IVEF 975
980 1000 1020 1040
1 EGLRGPGQFQNNVGDFHPPPPNHFQPKASSQGMDGCAEVGGGSGEGLCRCDFL*KEMSRRGPRLPPPGFRCRSDT 1050
Consensus EGLRGPGQFQNNVGDFHPPPPNHFQPKASSQGMDGCAEVGGGSGEGLCRCDFL*KEMSRRGPRLPPPGFRCRSDT 1050
1060 1080 1100 1120
1 VSQRRGQ*CIAAILGEDFLGGYLLSSLAGPWAGVTMDRFQAFFSESFQCSGYRNFRKAGLRRI*VKLGPSTPSFS 1125
Consensus VSQRRGQ*CIAAILGEDFLGGYLLSSLAGPWAGVTMDRFQAFFSESFQCSGYRNFRKAGLRRI*VKLGPSTPSFS 1125
1140 1160 1180 1200
1 PWVMFLRGPGGR*TSLALCCSCKDRVGIFYQQNSWNFHTAQPSQVQGYSPDTQVKVPALAPTTGAPLPLLPRSL* 1200
Consensus PWVMFLRGPGGR*TSLALCCSCKDRVGIFYQQNSWNFHTAQPSQVQGYSPDTQVKVPALAPTTGAPLPLLPRSL* 1200
1220 1240 1260
1 EGWSADKDPAQVRPCFPPGHNFHVTIAWDNVK*K*QTLSFKSQVDEGRDRDTLELKQ*GSVIFYSCCKHFL*CYY 1275
Consensus EGWSADKDPAQVRPCFPPGHNFHVTIAWDNVK*K*QTLSFKSQVDEGRDRDTLELKQ*GSVIFYSCCKHFL*CYY 1275

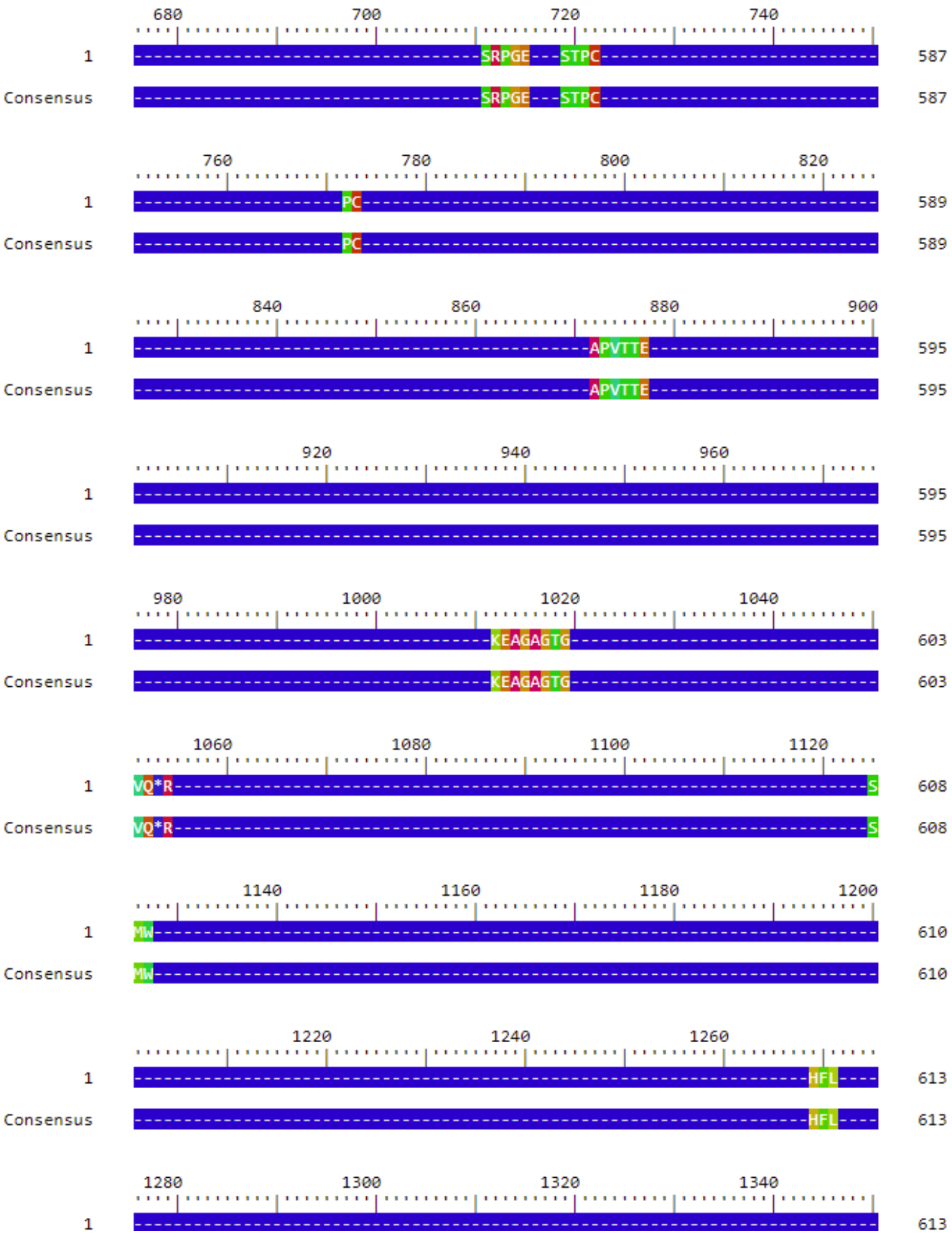
```
knitr::include_graphics('Pattern3.png')
```



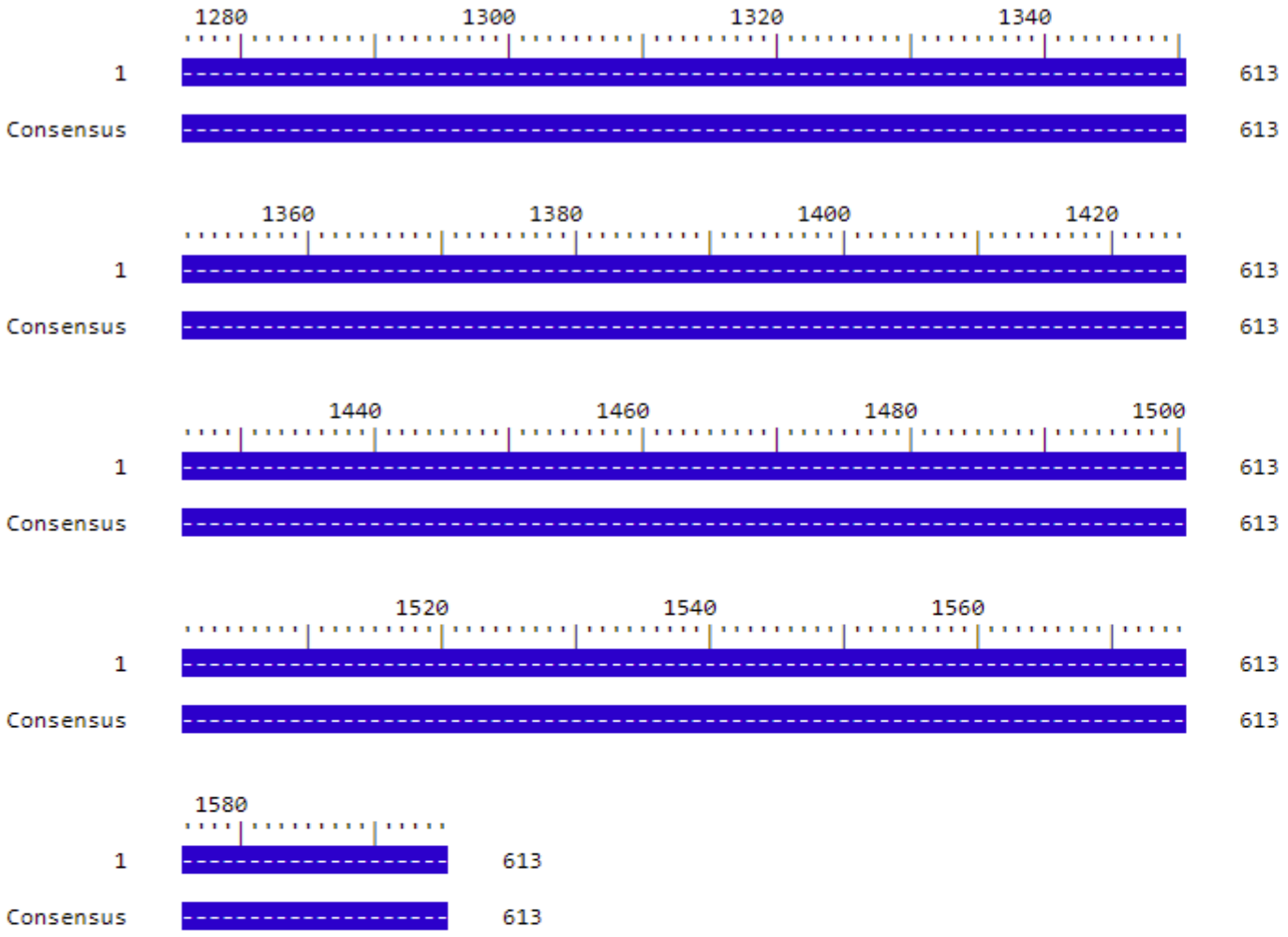
```
BrowseSeqs(alignedSubject(a), colWidth = 75)

knitr::include_graphics('Subject1.png')
```


20 40 60
1 ----AAGRGAGRWAMAGCRLWVSLLLAAALACLATALWPWPQYIQTYHRRYTLYPNNFQFRYHVSSAAQAGCVVL 71
Consensus ----AAGRGAGRWAMAGCRLWVSLLLAAALACLATALWPWPQYIQTYHRRYTLYPNNFQFRYHVSSAAQAGCVVL 71
80 100 120 140
1 DEAFRRYRNLLFGSGSWPRPSFSNKQQTLGKNILVVSVVTAECNEFPNLESVENYTLTINDDQCLLASETVWGAL 146
Consensus DEAFRRYRNLLFGSGSWPRPSFSNKQQTLGKNILVVSVVTAECNEFPNLESVENYTLTINDDQCLLASETVWGAL 146
160 180 200 220
1 RGLETFSQLVWKSAEGTFFINKTKIKDFPRFPHRGVLLDTSRHYLPLSSILDTLDVMAYNKFNVFHWHLVDDSSF 221
Consensus RGLETFSQLVWKSAEGTFFINKTKIKDFPRFPHRGVLLDTSRHYLPLSSILDTLDVMAYNKFNVFHWHLVDDSSF 221
240 260 280 300
1 PYESFTFPELTRKGSFNPVTHIYTAQDVKEVIEYARLRGIRVLAEFDTPGHTLSWGPGAPGLLTPCYSGSHLSGT 296
Consensus PYESFTFPELTRKGSFNPVTHIYTAQDVKEVIEYARLRGIRVLAEFDTPGHTLSWGPGAPGLLTPCYSGSHLSGT 296
320 340 360
1 FGPVNPSLNSTYDFMSTLFLEISSVFPDFYLHLGGDEVDFTCWKSNPNIQAFMKKKGF-TDFKQLESFYIQTLLD 370
Consensus FGPVNPSLNSTYDFMSTLFLEISSVFPDFYLHLGGDEVDFTCWKSNPNIQAFMKKKGF-TDFKQLESFYIQTLLD 370
380 400 420 440
1 IVSDYDKGYVVWQEVFDNKVKVRPDTIIQVWREEMPVEYMLEMQDITRAGFRALLSAPWYLNRVKYGPDWKDMYK 445
Consensus IVSDYDKGYVVWQEVFDNKVKVRPDTIIQVWREEMPVEYMLEMQDITRAGFRALLSAPWYLNRVKYGPDWKDMYK 445
460 480 500 520
1 VEPLAFHGTPEQKALVIGGEACMWGEYVDSTNLVPRLWPRAGAVAERLWSSNLTTNIDFAFKRLSHFRCELVRRG 520
Consensus VEPLAFHGTPEQKALVIGGEACMWGEYVDSTNLVPRLWPRAGAVAERLWSSNLTTNIDFAFKRLSHFRCELVRRG 520
540 560 580 600
1 IQAQPISVGCCEQEFEQT*ATSAEHPGGC----------CPLSQLR*A---------- 558
Consensus IQAQPISVGCCEQEFEQT*ATSAEHPGGC----------CPLSQLR*A---------- 558
620 640 660
1 ----------PRRVLALRE----------QVPGQG*SFTA---- 578
Consensus ----------PRRVLALRE----------QVPGQG*SFTA---- 578

```
knitr::include_graphics('Subject2.png')
```


680
700
720
740
1
SRPGE
STPC
587
Consensus
SRPGE
STPC
587
760
780
800
820
1
PC
589
Consensus
PC
589
840
860
880
900
1
APVTTE
595
Consensus
APVTTE
595
920
940
960
1
595
Consensus
595
980
1000
1020
1040
1
KEAGAGTG
603
Consensus
KEAGAGTG
603
1060
1080
1100
1120
1
VQ*R
S
608
Consensus
VQ*R
S
608
1140
1160
1180
1200
1
MW
610
Consensus
MW
610
1220
1240
1260
1
HFL
613
Consensus
HFL
613
1280
1300
1320
1340
1
613

```
knitr::include_graphics('Subject3.png')
```



#Subtask (e)

```
matchPattern("*", seqMouseProtein)
```

```
## Views on a 613-letter AAString subject
## subject: AAGRGAGRWAMAGCRLWVSLLLAAALACLATALW...RPGESTPCPCAPVTTEKEAGAGTGVQ*RSMWHFL
## views:
##       start end width
##   [1]   539 539     1 [*]
##   [2]   557 557     1 [*]
##   [3]   574 574     1 [*]
##   [4]   606 606     1 [*]
```

```
matchPattern("*", seqHumanProtein)
```

```
## Views on a 1595-letter AAString subject
## subject: LTWPAPSERGDQRAMTSSRLWFSLLLAAAFAGRA...FSTKKKNK*ELGALFIKICISFINKL*NKG*KLV
## views:
##        start  end width
##    [1]   544  544     1 [*]
##    [2]   556  556     1 [*]
##    [3]   658  658     1 [*]
##    [4]   667  667     1 [*]
##    [5]   675  675     1 [*]
##    ...   ...  ...   ... ...
##   [32]  1521 1521     1 [*]
##   [33]  1539 1539     1 [*]
##   [34]  1570 1570     1 [*]
##   [35]  1588 1588     1 [*]
##   [36]  1592 1592     1 [*]
```

#Subtask (f)

```
newSeqHumanProtein <- subseq(seqHumanProtein, start = 1, end = 543)
newSeqMouseProtein <- subseq(seqMouseProtein, start = 1, end = 538)

newSeqHumanProtein
```

```
## 543-letter AAString object
## seq: LTWPAPSERGDQRAMTSSRLWFSLLLAAAFAGRATA...TFAYERLSHFRCELLRRGVQAQPLNVGFCEQEFEQT
```

```
newSeqMouseProtein
```

```
## 538-letter AAString object
## seq: AAGRGAGRWAMAGCRLWVSLLLAAALACLATALWPW...DFAFKRLSHFRCELVRRGIQAQPISVGCCEQEFEQT
```

#Subtask (g)

```
d <- pairwiseAlignment(newSeqHumanProtein,
                       newSeqMouseProtein,
                       type = "global",
                       substitutionMatrix = "BLOSUM62")

score(d)
```

```
## [1] 2400
```

```
#The score is marginally better for the
#shortened sequence than the original sequence
```

#Subtask (h)

```
e <- pairwiseAlignment(seqHumanProtein,
                       seqMouseProtein,
                       type = "local",
                       substitutionMatrix = "BLOSUM62")

score(e)
```

```
## [1] 2450
```

```
#The score using the Smith-Waterman local alignment is marginally better
#than the global alignment because the global alignment yielded
#a negative score, which means that there are many badly aligning regions,
#or many gaps in the alignment. The local alignment ignores those.
```

##Question 2 #Subtask (a)

```
string1 <- AAString(x = "HEAGAWGHEE")
string2 <- AAString(x = "PAWHEAE")

f <- pairwiseAlignment(string1,
                       string2,
                       type = "global",
                       substitutionMatrix = "BLOSUM50",
                       gapOpening = 1,
                       gapExtension = 2)
score(f)
```

```
## [1] 27
```

#Subtask (b)

```
randomSequences <- function(seq, n){
# Generates "n" random amino acid sequences for a given sequence
#
# Inputs:
# seq - the sequence from which to sample
# n - number of generated random sequences
#
# Output: a cell array of n random sequences
# Identifies the unique amino acids
  s <- uniqueLetters(seq)
# Calculates the length of given amino acid sequence
  l <- length(seq)
# Calculates the proportions of amino acids of the given sequence
```

```
  t <- letterFrequency(seq, letters = s)/l
# Generates n random amino acid sequences
  randseqs <- c(1:n)
    for (i in 1:n){
  randseqs[i] <- paste(sample(s, length(seq), replace = T, prob = t), collapse="")
}
# Returns generated random sequences
return(randseqs)
}

string3 <- randomSequences(string2, 1000)
```

#Subtask (c)

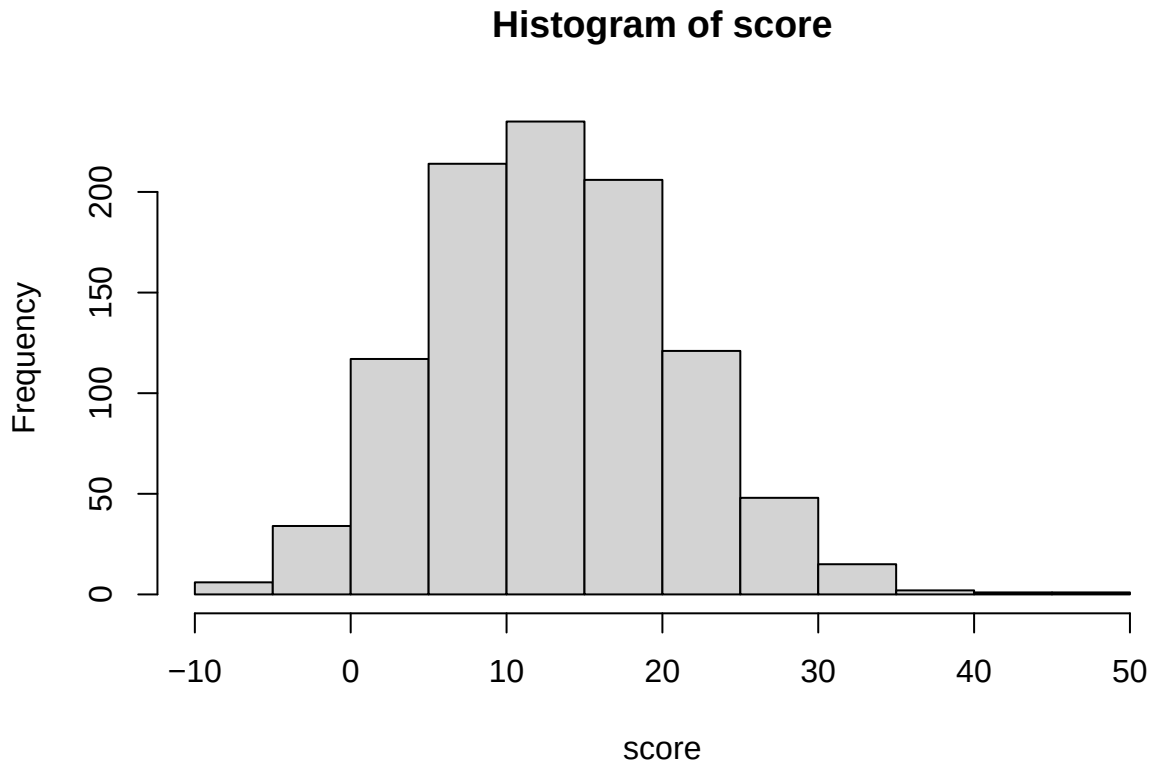
```
score = vector()

for (i in 1:length(string3)) {
  score[i] <- score(pairwiseAlignment(string1, string3[i], type = "global",
                                      substitutionMatrix = "BLOSUM50",
                                      gapOpening = 1,
                                      gapExtension = 2))
}
score
```

```
##    [1] 18 20 22 12 30 20 17  3 16  4 24  8 -3 22 20 25 23 16 13 18 10  9 29 18
##   [25] 11 23  5 11 21  7 31 22 12 20  7 10 33 19  7 22  4  7  5 14 23  1 15 12
##   [49]  2 19  8 18 14 19 12 -2 -2 12 20 25  9 -5 10  0  7 15 16 10  9 27 10 17
##   [73]  5 28  3 20 16 23  2 16 21 18 15 26  7 43 25  7 19  1 12 19  4 25  4 11
##   [97] 10  7 16 30 13 24 11 -1  9 15 22 28 12 19 17 23  2  8 22  8 17 20 25 17
##  [121]  9 14 13 20 16 13  4 22 16  0 17 13 13  6 18 19 22  7 13  4 19 23 11 13
##  [145]  1 11 12 18 13 26 21  4 11  6 10  6 14 16 -3 11  6  5 -2  0  3 13  8 12
##  [169] 14 18 22 13 18  1  5 -2 11 12  8 14 10 23  3 23 19 19  9 10 12 10 18 15
##  [193] 19  7  2 13 16 10 19 20 30 15  4 10  7 21  6 25 22  2 10 27 13 11 11  5
##  [217] 12  9 21 14 21  3 20 10  2  4  8 10 18  5 22 10  6 20 23  8 20  9 13 12
##  [241] 18 13  8 28  8 19  9 11  6  9  8  6 12 21 17  4  2 12  8 15 24  3  9 23
##  [265] 26 13 11  6 30 18  9 20 24 11 16 28 13 13 19 10  6 20 13 20 17 20  1 15
##  [289] 10 11 24 13  3 19 14 12 11 22 21  4 14 28 11  5 17 10 19 17  8 16  9 15
##  [313] 14  7 -1  4  5 -1 19 18 12  5  7 21 13 18  7  7 -3 17 16 11  8  3  6 17
##  [337] 38 24 -3 21  6 33  0 19 12  7 31 21 10 24 10 15  6 11  7 27 11 14 17 15
##  [361] 33 12 15 17 21 19 26  3 14 18 17 25  3 12 21 12 12 13 16  8 13  6 10 12
##  [385]  5  4 15 10  3  5  6 19 18  5 12 13 24 12 18 14 28 10 22 20  6  9 22 12
##  [409]  3 19 11 12 15 13  1 13 48  9 11 24  7 16 23  3 22  9  9  8  7  4 13 -2
##  [433] 16  3  6  7 21  6 22 18 -4 26 13  8 20 18 13 16  9 20 13 17  3 21 25 11
##  [457]  4 14 10 15  9 18 12 18  9  5 25  9 25 12 11  9 21 20 15 17 22 19 29 13
##  [481] 13 27 10 13  8 14 17 12 32 19 19 14 16 18 11  8  8 11 11 14 24  5  3 10
##  [505] 10  5 16 18 25  2  7 13  3  7 16 14  0 -6 13  4 11  5  3  7  5 22 20  3
##  [529]  6 11 16  4 12 11 22  4  8 30 17 19 10 -5 29 26 12 12 14  3 20 15 10 18
##  [553]  6  2 13 32 15  9 10 33 10 24 18 15 22 -1 24  4  5 10  7 27 26 33 18  2
##  [577] 18 21  7 21 23 23  8  6  4 10 10  7 14 10 24 15  9 16 22 10 15 12  2 11
##  [601] 12 11  7 18 21 14  7 19 22  0 12 30 16 17  0 26 11 -5  8 18  6 -2 24 17
##  [625] 12 15 21 13 13 19  5 13 11  7  9  9  2  8 17 10 28 22 19 17  5 18 13 15
##  [649] 11 17 12  8  8 28 13 16 18  9  9 16  5 23 11  2 31 21 36 21  6 15 20 16
##  [673]  6 21  5  7 14  9  4 28  1  9 10  6 15 21  5  9  5 13 22 10 13  3  6 12
```

```
##  [697] 26  5 17 20 30 18 21 13 28 11 19  9 13  1  9  0  2 17 -3 16 11 13 21 -5
##  [721] 22 19 16  5 14  8 17 21 14  5 33 14 19  7 14 20 22 27 21 16  6 27 19 18
##  [745]  9 21 20  0  0 11 15 22  6 13 25 26  7 21 18 12 22 15 13 10 11 11  6 15
##  [769] 16 12 -1 12 29 16 19  5 19 12  2 11 14 10 14  5  7  3 11 10  5 19 14 17
##  [793] -1 22 23 17  1 16  9 18 10 28  3 33 11  0 13  9 18 12 29 13  6 10 11 28
##  [817] 19 22 10  6  0 16 18 21 33 15  3 16 18 18 12 11 27 27 12  8 16  4  8 13
##  [841] 27 13 20 18 24 19  9  9 14 11 16  0 11 10 20  9 18 20 23 24 16 23 11  1
##  [865]  9 20  7 20  4 14 26 18  1 11 11 15 15  5 18  5  6 16 30 14 19 11 -2 19
##  [889] 21  5  3 18 12 21 21  9 10 10 10  7  9 13  3 23 22 10 10  9 19 20 10 34
##  [913] 24 13 10 19  2  6 13  7 24 13 20 17 14 20 30  6 10  5 23 -2  8 19 26  1
##  [937] 15 13 16  8  8  4 14 20 -5 13 14 14 17  1 23 12  9 15 19 22  8 21 20 10
##  [961] 16  7  6  1 15 18 21 -1  9 35  9 17 12  3 16 19 20  8 15 15  8 15 16 16
##  [985] 11 10 16 12 25 15 18 22 21 10  9  6  5  7 16 18
```

#Subtask (d)

```
hist(score)
```




```
pvalue <- length(which(score >= 27)) / length(score)

pvalue
```

```
## [1] 0.055
```

```
#The alignment is almost statiscally significant
#since the p-value is larger than 0.05, which means that
#6.9% of random sequences would have as large score as our result.
```