BIOLOGY TP – GENETIC CODE IN R

Mauricio Salim Gómez Chicre

Introduction

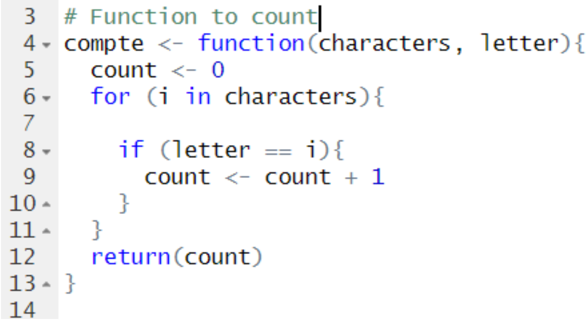
In this TP we will be working with DNA and protein sequences. In particularly with sequences of patients that suffer from polycystic kidney disease (PKD), an autosomal dominant genetic disease. It is characterized by the progressive development of fluid cysts in the kidneys, but it remains asymptomatic form many years. However, when the development of cysts begin it causes lower back and abdominal pain, as well as progressive impairment of renal function, possibly up to the stage of end-stage kidney failure which requires dialysis or kidney transplantation (50% of PKD patients at 62yo), and throughout the disease there can be other serious health complications. Finally, it is worth noting that this pathology is the most common hereditary renal disease (1/1000) and is the consequence of genetic mutations occurring in the PKD1 gene (80% of mutations) or the PKD2 gene (20% of mutations).

Objectives

- Become familiar with the manipulation of sequences and define some of their properties (nucleotide composition, length, GC rate),
- Understand the central dogma of molecular biology to allow the translation of nucleotide sequences into protein sequences,
- Align and compare patient sequences to identify the mutated gene (PKD1 or PKD2) and the mutation (e.g. Ala372Thr or A372T).

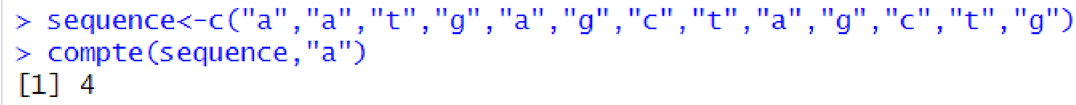
Exercise 2.1

- **Write a function *compte* which takes two arguments: a sequence of characters and a letter, and which returns the number of occurrences of the letter in the sequence.**



```
# Function to count
compte <- function(characters, letter){
  count <- 0
  for (i in characters){

    if (letter == i){
      count <- count + 1
    }
  }
  return(count)
}

```

- **Apply your function to find the number of occurrences of the letter "a" in the biological sequence.**



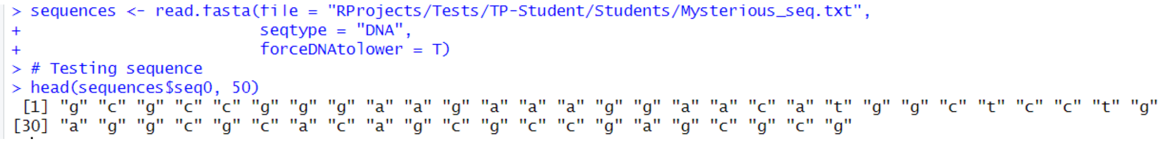
```
> sequence<-c("a","a","t","g","a","g","c","t","a","g","c","t","g")
> compte(sequence,"a")
[1] 4
```

Exercise 3.1

- **What are the *seqtype* and *forceDNAtolower* options for?**

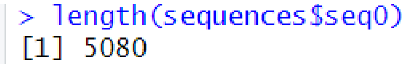
Seqtype refers to the type of the sequence if its DNA or aa (amino acids). On the other hand, *forceDNAtolower* is a Boolean parameter to determine if the DNA sequence should be return in lower case letters.

- **Print the first 50 nucleotides of your sequence.**



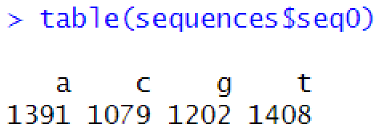
```
> sequences <- read.fasta(file = "RProjects/Tests/TP-Student/Students/Mysterious_seq.txt",
+                         seqtype = "DNA",
+                         forceDNAtolower = T)
> # Testing sequence
> head(sequences$seq0, 50)
 [1] "g" "c" "g" "c" "c" "g" "g" "g" "a" "a" "g" "a" "a" "a" "g" "g" "a" "a" "c" "a" "t" "g" "g" "c" "t" "c" "c" "t" "g"
[30] "a" "g" "g" "c" "g" "c" "a" "c" "a" "g" "c" "g" "c" "c" "g" "a" "g" "c" "g" "c" "g"
```

- **Print the length of the sequence.**



```
> length(sequences$seq0)
[1] 5080
```

- **Print the number of occurences of nucleotides (a, t, c, g). Tip: help(table)**



```
> table(sequences$seq0)

   a    c    g    t 
1391 1079 1202 1408 
```

Exercise 3.2

The GC-content of a DNA sequence is the proportion of bases in this sequence that are either G or C. As G bonds specifially to C (so as A and T in DNA or A and U in RNA), the GC-content of a sequence gives the proportion of GC-bonds in it.

- **Why is the computation of GC-content relevant in molecular biology?**

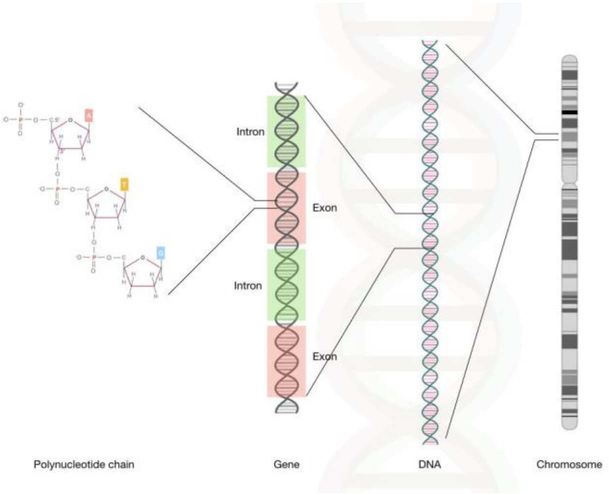


Figure 1. The location of genes on a chromosome. [1]

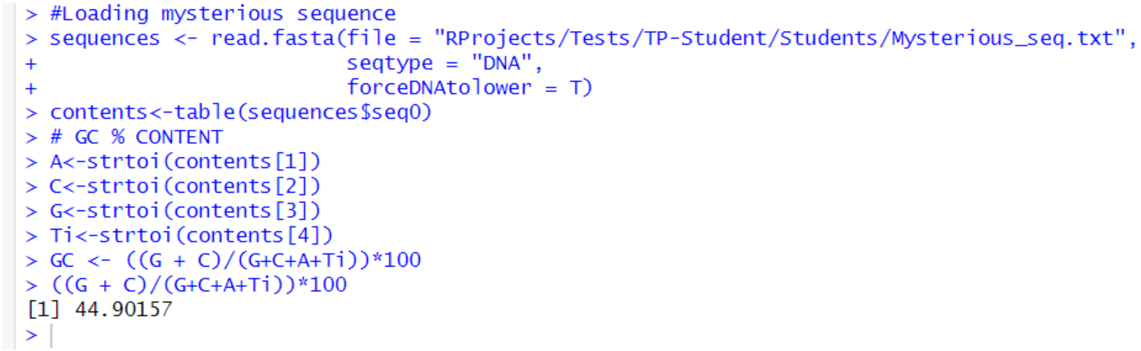
GC rich regions are present both in coding and non-coding regions, as such they represent an important role in synthesis of proteins and gene expression. These GC rich regions are also domains known as "isochores" and they form have form part in some vital genes. They are the darker regions that can be seen in *Figure 1.*

On the other hand, they have an important function in stabilizing the DNA, as higher GC content regions have higher thermal stability while lower ones have low thermostability.

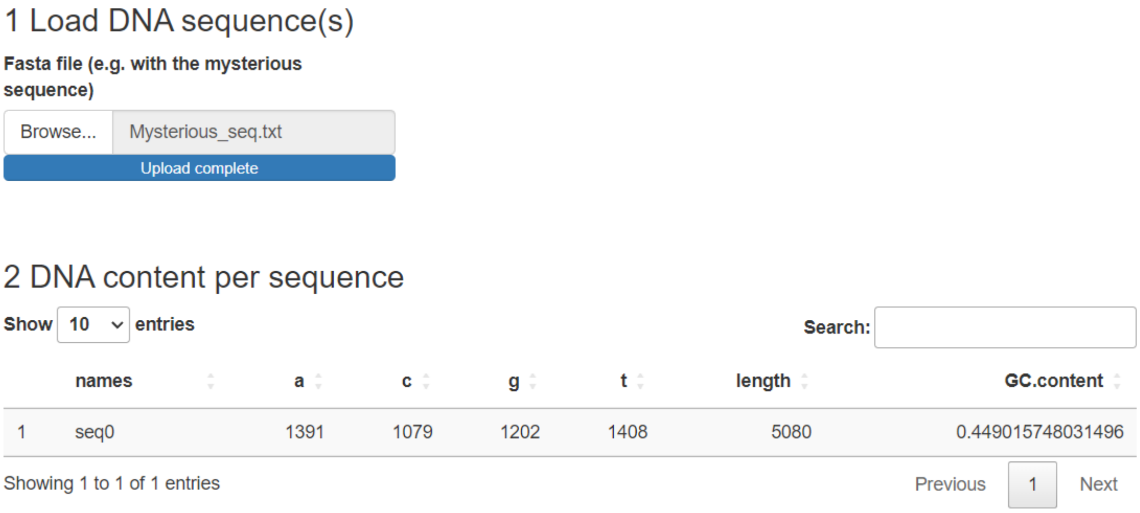
Meaning a DNA with more GC content is highly stable due to the presence of more hydrogen bonds, however, research shows the hydrogen bonds aren't the direct responsible, instead it is caused mainly by molecular interactions of base stacking.

So, taking into account the high thermal stability of GC rich regions, they also have a higher annealing and melting temperature, and because of that they are hard to amplify due to the need of additional optimization steps and elevated annealing temperatures. Increasing temperatures can lead to errors, in the PCR, higher GC templates increase the chances of non-specific bindings and consequently the chances of false-positive results. While selecting the PCR template DNA and designing primers, it is important to be careful. [1]

- **Compute the GC-content of your sequence.**



```
> #Loading mysterious sequence
> sequences <- read.fasta(file = "RProjects/Tests/TP-Student/Students/Mysterious_seq.txt",
+                         seqtype = "DNA",
+                         forceDNAtolower = T)
> contents<-table(sequences$seq0)
> # GC % CONTENT
> A<-strtoi(contents[1])
> C<-strtoi(contents[2])
> G<-strtoi(contents[3])
> Ti<-strtoi(contents[4])
> GC <- ((G + C)/(G+C+A+Ti))*100
> ((G + C)/(G+C+A+Ti))*100
[1] 44.90157
> 
```



1 Load DNA sequence(s)

Fasta file (e.g. with the mysterious sequence)

2 DNA content per sequence

	names	a	c	g	t	length	GC.content
1	seq0	1391	1079	1202	1408	5080	0.449015748031496

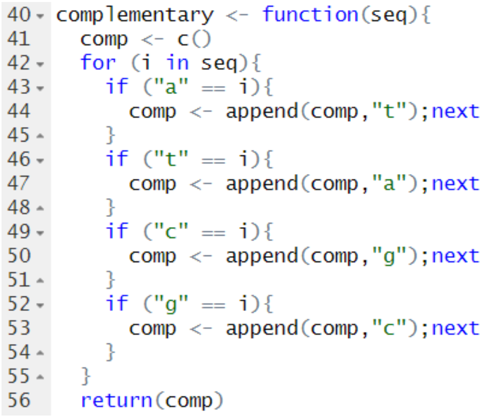
Showing 1 to 1 of 1 entries

Exercise 3.3

- **By convention, in which direction (5'->3' or 3'->5') do we write DNA sequences? What do 5' and 3' represent?**

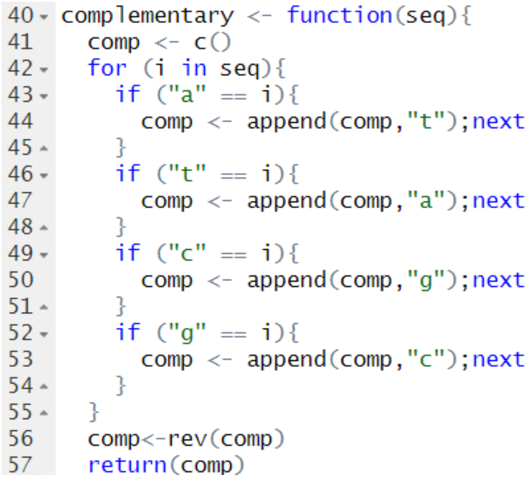
By convention, single strands of DNA and RNA sequences are written in a 5′-to-3′ direction except as needed to illustrate the pattern of base pairing.

- **Write a function *complementary* that takes a sequence as input and returns its complementary sequence (i.e., a becomes t; t, a; c, g; and g, c). Tip: You can use the *chartr* function instead of if* statements if you want.***

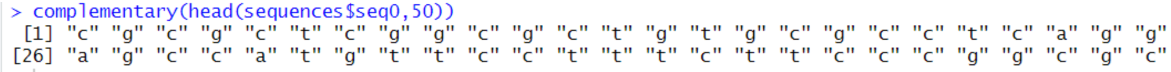


```
complementary <- function(seq){
  comp <- c()
  for (i in seq){
    if ("a" == i){
      comp <- append(comp,"t");next
    }
    if ("t" == i){
      comp <- append(comp,"a");next
    }
    if ("c" == i){
      comp <- append(comp,"g");next
    }
    if ("g" == i){
      comp <- append(comp,"c");next
    }
  }
  return(comp)
```

- **Compute and print the reverse complementary of the reference sequence. Remember that sequences are always written in 5'–>3' sense. The reversed version of a sequence *seq* can be obtained with *rev(seq)*.**



```
complementary <- function(seq){
  comp <- c()
  for (i in seq){
    if ("a" == i){
      comp <- append(comp,"t");next
    }
    if ("t" == i){
      comp <- append(comp,"a");next
    }
    if ("c" == i){
      comp <- append(comp,"g");next
    }
    if ("g" == i){
      comp <- append(comp,"c");next
    }
  }
  comp<-rev(comp)
  return(comp)
```



```
> complementary(head(sequences$seq0,50))
 [1] "c" "g" "c" "g" "c" "t" "c" "g" "g" "c" "g" "c" "t" "g" "t" "g" "c" "g" "c" "c" "t" "c" "a" "g" "g"
[26] "a" "g" "c" "c" "a" "t" "g" "t" "t" "c" "c" "t" "t" "t" "c" "t" "t" "c" "c" "c" "g" "g" "c" "g" "c"
```

Exercise 5.1

- **What is the transcription mechanism for? Briefly describe the main steps of transcription.**

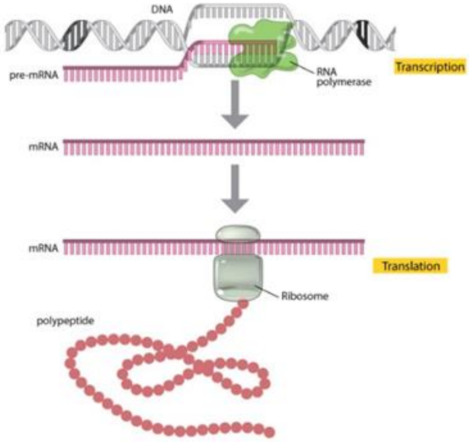


Figure 2. Process of synthesis of proteins from DNA. Transcription and Translation.

One of the strands of DNA within a gen gets transcribed with the help of RNA polymerase (See *Figure 2.*), that binds to a specific codon sequence called a promoter (where the gen starts). This first step is called **initiation** and RNA polymerase is helped also by proteins called transcription factors. It doesn't need a primer.

In the next step, **elongation**, RNA polymerase separates the 2 strands apart. One of them will be the template strand/ or antisense strand, meaning it will be used to generate the mRNA, the other is the nontemplate/ or sense strand. The process in which the RNA polymerase moves along the template strands and produces the mRNA is called elongation. It's always read in the 3' to 5' sense, generating the mRNA in the 5' to 3' sense. RNA polymerase zips DNA back, keeping only 10 to 20 bases exposed at a time. Until RNA polymerase receives a stop signal it will not stop. When stop signal is received it stops and finished the transcription, this step is called **termination**. [2]

- **What is(are) the difference(s) between mRNA and cDNA?**

Complementary DNA (cDNA) is a DNA copy of a messenger RNA (mRNA) molecule produced by reverse transcriptase, a DNA polymerase that can use either DNA or RNA as a template.

cDNA is a more convenient way to work with the coding sequence than mRNA because RNA is very easily degraded by omnipresent RNases. This the main reason cDNA is sequenced rather than mRNA. Likewise, investigators conducting DNA microarrays often convert the mRNA into cDNA to produce their probes. [3]

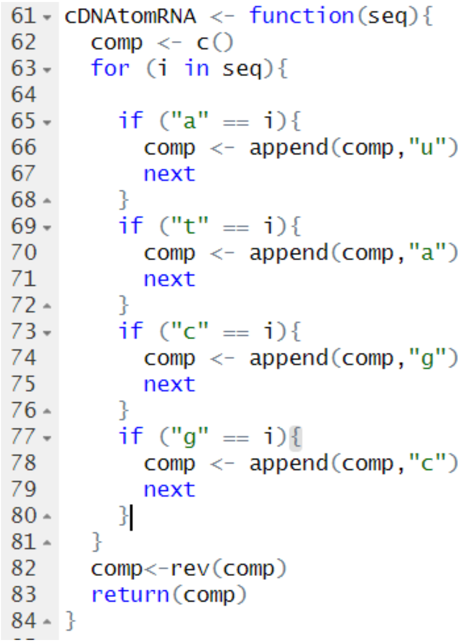
- **What enzyme catalyzes the mRNA to cDNA reaction?**

A reverse transcriptase (RT), it can be found in retrovirus such as the HIV that causes the AIDS disease.

- **What is the difference between cDNA and the original DNA sequence (if any)?**

The main difference is that the cDNA doesn't have the introns (non-coding sequence) in the sequence, it only has the exons (coding sequence). Additionally, in nature cDNA is single stranded while double DNA is double stranded.

- **Write a function *transcribe* that converts the mysterious cDNA sequence into RNA.**



```r
cDNAtomRNA <- function(seq){
  comp <- c()
  for (i in seq){

    if ("a" == i){
      comp <- append(comp,"u")
      next
    }
    if ("t" == i){
      comp <- append(comp,"a")
      next
    }
    if ("c" == i){
      comp <- append(comp,"g")
      next
    }
    if ("g" == i){
      comp <- append(comp,"c")
      next
    }
  }
  comp<-rev(comp)
  return(comp)
}
```

Exercise 6.1

- **What is the translation mechanism for? Briefly describe the main steps of translation.**

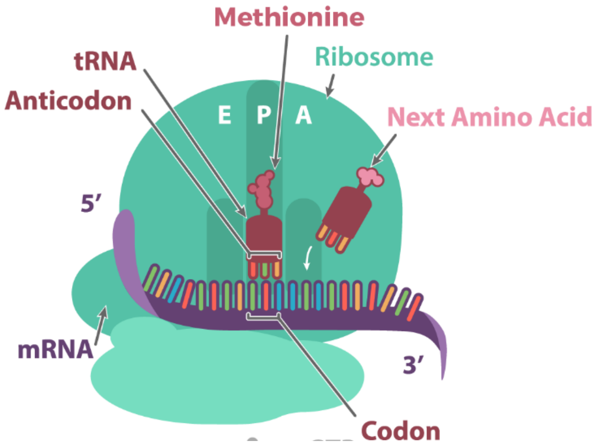


Figure 3. Translation mechanism initiation complex. [4]

During translation in the ribosome, mRNA acts as a code for specific proteins, because each codon in the mRNA will code for a specific anticodon carried by tRNA. tRNA transport the amino acids that then are used by the ribosome to synthetize the protein.

So, during the **initiation** step the small ribosomal subunit joins the mRNA and an initiator tRNA, then comes the large ribosomal subunit and joins to complete the translation initiation complex (*Figure 3*). Then during the **elongation** step, more tRNAs enter the ribosomal translation complex and the aminos acid are covalently bonded. The first tRNA detaches, the second moves to give space for another tRNA and so forth. The polypeptide will begin the **termination** step when the initiation complex finds a stop codon in the mRNA (UAA, UAG, or UGA), it will finish the translation and the protein will float away for folding and further modifications. [4]

- **What is the genetic code? What are its main characteristics (number of nucleotides used for decoding, sliding properties of the ribosome, etc.)?**

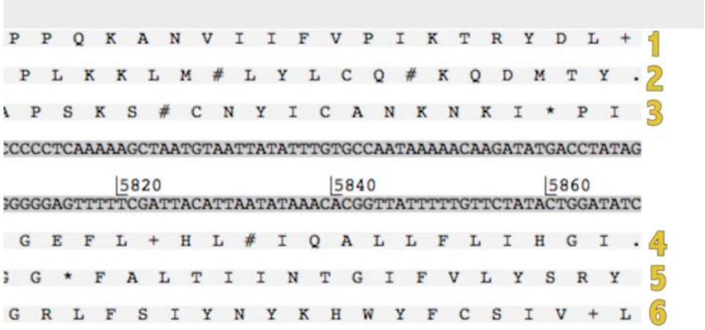
The genetic code is a set of rules defining how the four-letter code of DNA is translated into the 20-letter code of amino acids, which are the building blocks of proteins. The genetic code is a set of three-letter combinations of nucleotides called codons, each of which corresponds to a specific amino acid or stop signal. [5]

- **How many amino acids are there in the standard genetic code? Why is the genetic code defined as “redundant’?**

In the standard genetic code, there are 20-letter code of amino acids. However, there are 64 possible permutations, or combinations, of three-letter nucleotide sequences that can be made from the four nucleotides. Of these 64 codons, 61 represent amino acids, and three are stop signals. Although each codon is specific for only one amino acid (or one stop signal), the genetic code is described as degenerate, or redundant, because a single amino acid may be coded for by more than one codon. It is also important to note that the genetic code does not overlap, meaning that each nucleotide is part of only one codon-a single nucleotide cannot be part of two adjacent codons. Furthermore, the genetic code is nearly universal, with only rare variations reported. For instance, mitochondria have an alternative genetic code with slight variations. [5]

- **What is a “reading frame”? How many reading frames are there for one sequence?**

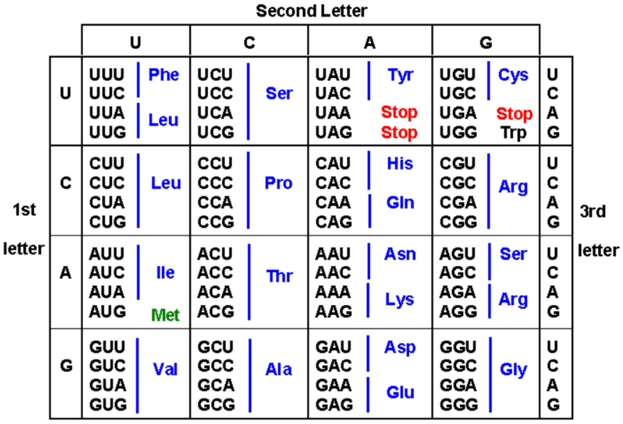
A reading frame is how a sequence of nucleotides is read (DNA or RNA). For a 2 strand DNA there a 6 possible reading frame to translate the genes, an example can be seen in *Figure 4.*



```
P P Q K A N V I I F V P I K T R Y D L + 1
 P L K K L M # L Y L C Q # K Q D M T Y .2
A P S K S # C N Y I C A N K N K I * P I 3
CCCCCTCAAAAAGCTAATGTAATTATATTTGTGCCAATAAAAACAAGATATGACCTATAG
                  |5820                |5840                 |5860
GGGGGAGTTTTTCGATTACATTAATATAAACACGGTTATTTTTGTTCTATACTGGATATC
 G E F L + H L # I Q A L L F L I H G I .4
G G * F A L T I I N T G I F V L Y S R Y 5
G R L F S I Y N Y K H W Y F C S I V + L 6
```

Figure 4. Reading frames.

- **What is the amino acid acting as the Start codon? Which codon correspond to this amino acid?**



Second Letter

1st letter		U		C		A		G		3rd letter
U		UUU	Phe	UCU		UAU	Tyr	UGU	Cys	U
		UUC		UCC	Ser	UAC		UGC		C
		UUA	Leu	UCA		UAA	Stop	UGA	Stop	A
		UUG		UCG		UAG	Stop	UGG	Trp	G
C		CUU		CCU		CAU	His	CGU		U
		CUC	Leu	CCC	Pro	CAC		CGC	Arg	C
		CUA		CCA		CAA	Gln	CGA		A
		CUG		CCG		CAG		CGG		G
A		AUU		ACU		AAU	Asn	AGU	Ser	U
		AUC	Ile	ACC	Thr	AAC		AGC		C
		AUA		ACA		AAA	Lys	AGA	Arg	A
		AUG	Met	ACG		AAG		AGG		G
G		GUU		GCU		GAU	Asp	GGU		U
		GUC	Val	GCC	Ala	GAC		GGC	Gly	C
		GUA		GCA		GAA	Glu	GGA		A
		GUG		GCG		GAG		GGG		G

Figure 5. The Genetic code.

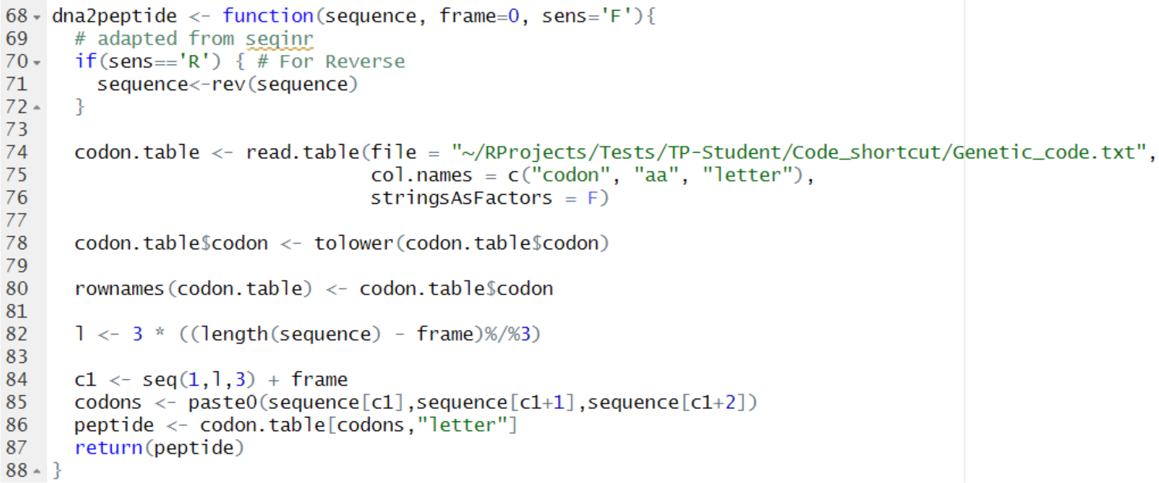
A start codon is made up of the letters ATG (or AUG in RNA), which codes for the amino acid methionine (*Figure 5*).

- **Which are the Stop codons? Which one-letter code is used here to describe Stop codons?**

The stop codons are UAA, UAG and UGA, the one-letter code used here to describe is an X.

Exercise 6.2

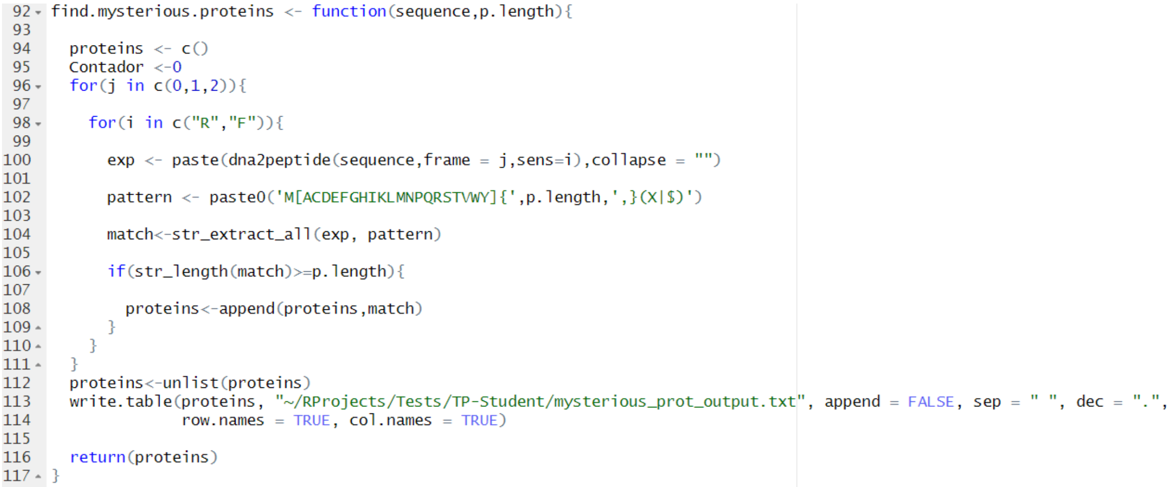
In your working R script, write a function *dna2peptide* that translates a DNA sequence into an amino acid chain. The function returns the amino acid sequence.



```r
dna2peptide <- function(sequence, frame=0, sens='F'){
  # adapted from seqinr
  if(sens=='R') { # For Reverse
    sequence<-rev(sequence)
  }

  codon.table <- read.table(file = "~/RProjects/Tests/TP-Student/Code_shortcut/Genetic_code.txt",
                            col.names = c("codon", "aa", "letter"),
                            stringsAsFactors = F)

  codon.table$codon <- tolower(codon.table$codon)

  rownames(codon.table) <- codon.table$codon

  l <- 3 * ((length(sequence) - frame)%/%3)

  c1 <- seq(1,l,3) + frame
  codons <- paste0(sequence[c1],sequence[c1+1],sequence[c1+2])
  peptide <- codon.table[codons,"letter"]
  return(peptide)
}
```

Exercise 6.3

Define a function *find.mysterious.proteins* function that takes a cDNA sequence as input and returns a list of proteins as defined above. Apply this function to the mysterious sequence, save the resulting sequences in a text file and join it to your report.



```r
find.mysterious.proteins <- function(sequence,p.length){

  proteins <- c()
  Contador <-0
  for(j in c(0,1,2)){

    for(i in c("R","F")){

      exp <- paste(dna2peptide(sequence,frame = j,sens=i),collapse = "")

      pattern <- paste0('M[ACDEFGHIKLMNPQRSTVWY]{',p.length,',}(X|$)')

      match<-str_extract_all(exp, pattern)

      if(str_length(match)>=p.length){

        proteins<-append(proteins,match)
      }
    }
  }
  proteins<-unlist(proteins)
  write.table(proteins, "~/RProjects/Tests/TP-Student/mysterious_prot_output.txt", append = FALSE, sep = " ", dec = ".",
              row.names = TRUE, col.names = TRUE)

  return(proteins)
}
```

Text File Output:

```
"1"
"MQTLVCSNKELCVAYSRESKDRDEAMRKNLQKPSMFHFQPGKVYLLMFSLEIGLTFYQL
NEQLVKLKYSIKRQDSLGSPSEGYSTALHALLLISAQNVTLLNISVKEDFFFX"
```

"2"
"MVNSSRVQPQQPGDAKRPPAPRAPDPGRLMAGCAAVGASLAAPGGLCEQRGLEIEMQRI
RQAAARDPPAGAAASPSPPLSSCSRQAWSRDNPGFEAEEEEEEVEGEEGGMVVEMDVEWR
PGSRRSAASSAVSSVGARSRGLGGYHGAGHPSGRRRRREDQGPPCPSPVGGGDPLHRHLPL
EGQPPRVAWAERLVRGLRGLWGTRLMEESSTNREKYLKSVLRELVTYLLFLIVLCILTYGM
MSSNVYYYTRMMSQLFLDTPVSKTEKTNFKTLSSMEDFWKFTEGSLLDGLYWKMQPSNQ
TEADNRSFIFYENLLLGVPRIRQLRVRNGSCSIPQDLRDEIKECYDVYSVSSEDRAPFGPRNG
TAWIYTSEKDLNGSSHWGIIATYSGAGYYLDLSRTREETAAQVASLKKNVWLDRGTRATFI
DFSVYNANINLFCVVRLLVEFPATGGVIPSWQFQPLKLIRYVTTFDFFLAACEIIFCFFIFYYV
VEEILEIRIHKLHYFRSFWNCLDVVIVVLSVVAIGINIYRTSNVEVLLQFLEDQNTFPNFEHL
AYWQIQFNNIAAVTVFFVWIKLFKFINFNRTMSQLSTTMSRCAKDLFGFAIMFFIIFLAYAQ
LAYLVFGTQVDDFSTFQECIFTQFRIILGDINFAEIEEANRVLGPIYFTTFVFFMFFILLNMFLA
IINDTYSEVKSDLAQQKAEMELSDLIRKGYHKALVKLKLKKNTVDDISESLRQGGGKLNFD
ELRQDLKGKGHTDAEIEAIFTKYDQDGDQELTEHEHQQMRDDLEKEREDLDLDHSSLPRP
MSSRSFPRSLDDSEEDDDEDSGHSSRRRGSISSGVSYEEFQVLVRRVDRMEHSIGSIVSKIDA
VIVKLEIMERAKLKRREVLGRLLDGVAEDERLGRDSEIHREQMERLVREELERWESDDAAS
QISHGLGTPVGLNGQPRPRSSRPSSSQSTEGMEGAGGNGSSNVHVX"

"3"
"MWYPCVSTMCHISIMVTRKSLYDTFYCTVFHFTCKILQNSSFLPINYIYFSSLVMENSPPHL
FPIIFPCVLVLLKRHYIRKSFSTIKNVINVX"

"4"
"MQRVPAADWAGARRALVLSPAPPPARVARAVVAPKPPAPRAHGAHGRGGGRPPAARAP
LYVHLHHHSAFFPFHLLLLLLGLEAGVIAAPRLPGARRERRRRGGRGSGRGVPRGRLPDAL
HLDLQAPLLAEAARGGEAGAHGRAASHQPARVRRAGRGRPLGVPGLLRLHATGVHHRGH
WRPARGCAAPRSALCASGAMFLSSRR"

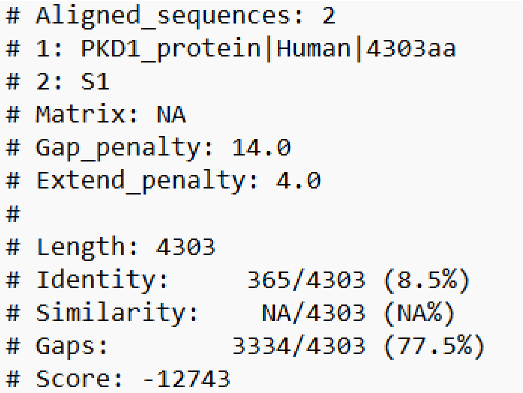
"5"
"MHKTMRKRRYVTSSRKTLLRYFSRLVLLSSMSLVPQRPRSPRTSLSAQATRGGCPSRGRW
RCSGSPPPTGLGHGGPWSSRRRRLPLGWPAPWX"

Exercise 7.1

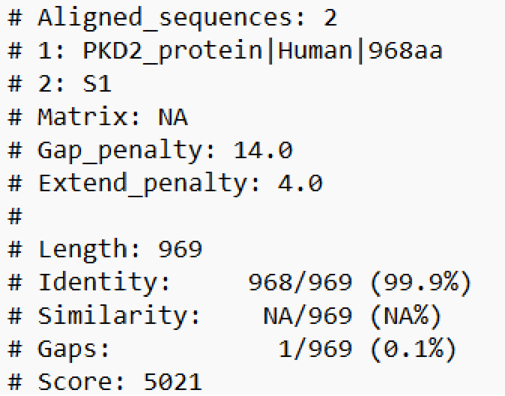
- **What are the pros and cons of aligning amino acids instead of nucleotides?**

There are 4 main reasons as to why aligning amino acids is better than nucleotides. First, the redundancy in codons ~1/3 of DNA mutations many times don’t matter. However, Nucleotide alignment doesn’t consider the redundancy in amino acids codons. For example, TCT, TCC, TCA, TCG, AGT, AGC code for serine but in a nucleotide sequence alignment a three-nucleotide mutation from TCA to AGT (a silent mutation that has no impact on the protein sequence) would score less than a single nucleotide mutation from TCA to TAA (a Ser-Stop codon mutation that would vastly alter the protein. A silent mutation rarely has any impact on evolution, so this kind of mutation should not count much to the score. Second, A smaller sized alphabet requires more matches. Nucleotide sequences are made with an alphabet of four nucleotides while most proteins have twenty. It is easier to achieve statistically significant alignment by comparing a larger alphabet of characters because you are much less likely to get a match by chance. Third, The DNA database is cluttered with non-coding sequences, since DNA sequences not only contain coding sequences, when trying to align two proteins it makes much more sense to limit the alignment to the coding sequence than just scanning the whole genome. This is especially important when searching for short sequences that by chance might align with non-coding regions of the DNA. Fourth, not all amino acid mutations are equally harmful to the protein structure. In nucleotide sequences it isn't considered the similarity of amino acids, in their structure and role in the protein. For example, isoleucine and valine have similar structures, both have hydrophobic side chains and differ only by the addition of an extra carbon on isoleucine. A mutation from one to the other is not as likely to substantially change the protein structure as some other mutations would, but a Nucleotide sequence alignment will treat this mutation the same as any other. However, it is also true that if we are studying recent radiation in sequences, it will probably be better to align nucleotides sequences to see the differences between the sequences. [6]

- **Back to your working R script (not in Shiny), align the longest mysterious translated protein from the previous part against the two reference proteins, PKD1 and PKD2 - you had generated the PKD1 and PKD2 sequences in the fasta format during the Unix session (exercise 2). What was the mysterious sequence?**



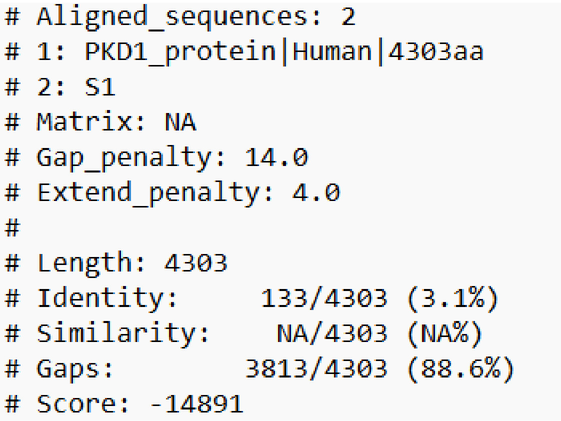
```
# Aligned_sequences: 2
# 1: PKD1_protein|Human|4303aa
# 2: S1
# Matrix: NA
# Gap_penalty: 14.0
# Extend_penalty: 4.0
#
# Length: 4303
# Identity:     365/4303 (8.5%)
# Similarity:    NA/4303 (NA%)
# Gaps:        3334/4303 (77.5%)
# Score: -12743
```



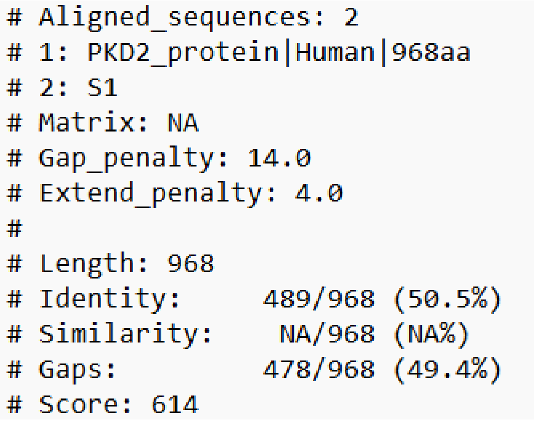
```
# Aligned_sequences: 2
# 1: PKD2_protein|Human|968aa
# 2: S1
# Matrix: NA
# Gap_penalty: 14.0
# Extend_penalty: 4.0
#
# Length: 969
# Identity:     968/969 (99.9%)
# Similarity:    NA/969 (NA%)
# Gaps:           1/969 (0.1%)
# Score: 5021
```

It was a PKD2_protein|Human|968aa, the score results si the highest in that protein.

- **Align your pair's sequence against the two reference proteins and identify which of the two reference proteins was assigned to you (use *pairwiseAlignment*). Write the result in a file (*writePairwiseAlignments*).**



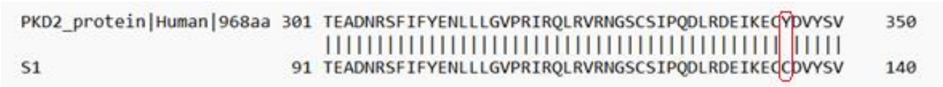
```
# Aligned_sequences: 2
# 1: PKD1_protein|Human|4303aa
# 2: S1
# Matrix: NA
# Gap_penalty: 14.0
# Extend_penalty: 4.0
#
# Length: 4303
# Identity:     133/4303 (3.1%)
# Similarity:    NA/4303 (NA%)
# Gaps:        3813/4303 (88.6%)
# Score: -14891
```



```
# Aligned_sequences: 2
# 1: PKD2_protein|Human|968aa
# 2: S1
# Matrix: NA
# Gap_penalty: 14.0
# Extend_penalty: 4.0
#
# Length: 968
# Identity:     489/968 (50.5%)
# Similarity:    NA/968 (NA%)
# Gaps:         478/968 (49.4%)
# Score: 614
```

I was assigned the PKD2_protein|Human|968aa.

- **By looking at the alignment, identify by eye the mutation/deletion/insertion between your sequence and the reference sequence for your protein by looking at the alignment result. Mutations are annotated in this way: G7602T means that a Glycine is mutated into a Threonine at the 7602th reference position.**



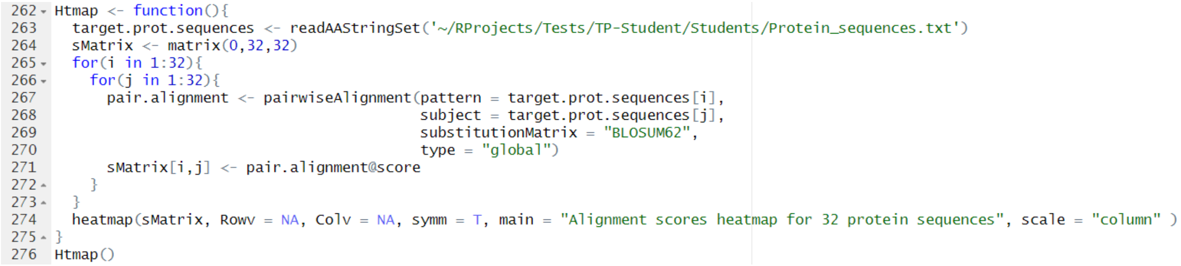
```
PKD2_protein|Human|968aa 301 TEADNRSFIFYENLLLGVPRIRQLRVRNGSCSIPQDLRDEIKECYDVYSV    350
                             |||||||||||||||||||||||||||||||||||||||||||| |||||
S1                        91 TEADNRSFIFYENLLLGVPRIRQLRVRNGSCSIPQDLRDEIKECCDVYSV    140
```

Figure 6. Mutation in the assigned sequence (seq31).

The mutation between my sequence (seq31) and the PKD2_protein|Human|968aa is Y345C (*Figure 6*).

Exercise 7.2

- **Compute the pairwise alignment for each of the 32 AA sequences against all the others in *Protein_sequences.txt.***
- **Build a 32 by 32 scores matrix. (Hint: Alignments scores are accessible with *<PairwiseAlignments>@score*)**
- **Use heatmap to plot the clustered scores matrix and comment the results. Find the correct parameters for pooling the sequences by group and facilitating the interpretation.**



```r
Htmap <- function(){
  target.prot.sequences <- readAAStringSet('~/RProjects/Tests/TP-Student/Students/Protein_sequences.txt')
  sMatrix <- matrix(0,32,32)
  for(i in 1:32){
    for(j in 1:32){
      pair.alignment <- pairwiseAlignment(pattern = target.prot.sequences[i],
                                          subject = target.prot.sequences[j],
                                          substitutionMatrix = "BLOSUM62",
                                          type = "global")
      sMatrix[i,j] <- pair.alignment@score
    }
  }
  heatmap(sMatrix, Rowv = NA, Colv = NA, symm = T, main = "Alignment scores heatmap for 32 protein sequences", scale = "column" )
}
Htmap()
```

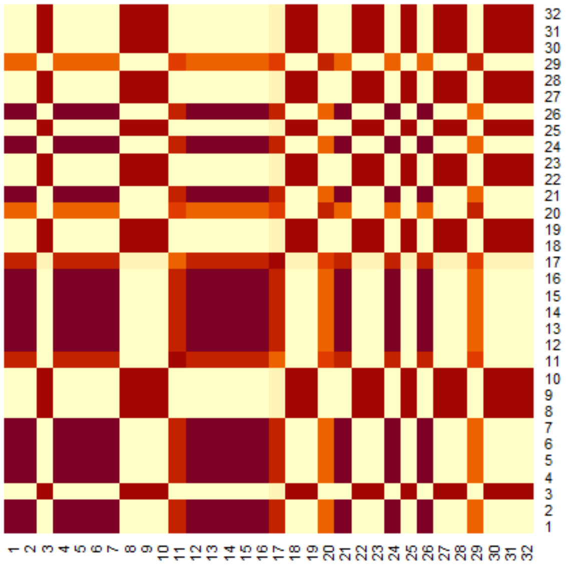


Figure 7. Alignment scores heatmap for 32 protein sequences.

- **Do you identify clusters? How can you retrieve the sequences? What is your interpretation?**

There are several red areas (clusters) that means the ones that had a higher score in the two-protein alignment. The conclusion that can be drawn from *Figure 7*. is even if there are some mutations between the proteins that are PKD1 or the ones that are PKD2, they still can be very similar. That's why choosing a scoring system and an alignment method (amino acid or nucleotide alignment) that consider the objective of the study has a significant effect on the results.

Bibliography

[1] D. T. Chauhan, "What is the importance of GC content?," Genetic Education, 13-Sep-2021. [Online]. Available: https://geneticeducation.co.in/what-is-the-importance-of-gc-content/. [Accessed: 04-Feb-2023].

[2] "Stages of transcription: Initiation, elongation &amp; termination (article)," Khan Academy. [Online]. Available: https://www.khanacademy.org/science/biology/gene-expression-central-dogma/transcription-of-dna-into-rna/a/stages-of-transcription. [Accessed: 04-Feb-2023].

[3] "cDNA Production," CDNA production. [Online]. Available: https://bio.davidson.edu/genomics/method/cDNAproduction.html#:~:text=cDNA%20is%20a%20more%20convenient,order%20to%20produce%20their%20probes. [Accessed: 04-Feb-2023].

[4] "Stages of translation (article)," Khan Academy. [Online]. Available: https://www.khanacademy.org/science/biology/gene-expression-central-dogma/translation-polypeptides/a/the-stages-of-translation. [Accessed: 04-Feb-2023].

[5] "genetic code," Nature news. [Online]. Available: https://www.nature.com/scitable/definition/genetic-code-13/#:~:text=The%20genetic%20code%20is%20a%20set%20of%20three%2Dletter%20combinations,and%20his%20colleagues%20in%201961. [Accessed: 04-Feb-2023].

[6] Lakna, “Difference between DNA and cdna: Definition, characteristics, synthesis, use,” Pediaa.Com, 07-Sep-2017. [Online]. Available: https://pediaa.com/difference-between-dna-and-cdna/. [Accessed: 04-Feb-2023].