

Fadhillah Putri Taha
H071171301

- Bagian A

1.

- a) Data diambil dari bank data *NCBI Sequence Database* dengan alamat <https://www.ncbi.nlm.nih.gov/nuccore/AB237837.1>, merupakan data virus Hepatitis C berkode AB237837.1. Judul data virus ini adalah **Hepatitis C virus full-length replicon pFGR-JFH1 RNA, complete sequence** dengan panjang protein pasangan basa 11.111bp.
- b) Hepatitis merupakan peradangan hati. Hepatitis C merupakan infeksi virus hepatitis C yang menyebabkan peradangan hati hingga kerusakan hati yang begitu serius. Virus hepatitis C dapat menyebar melalui darah yang terkontaminasi. Pada umumnya kasus hepatitis C terjadi pada orang yang menggunakan jarum suntik secara bersamaan. Beberapa orang dapat menghilangkan infeksi dengan sendirinya. Namun, pada beberapa orang tetap memiliki infeksi walaupun tidak merasakan gejala. Infeksi pada hati ini akan memicu terjadinya sirosis hati dan dapat memicu kanker hati.

Virus ini menjadi perhatian dalam tugas MID mata kuliah Pengantar Bioinformatika ini dikarenakan ada keluarga yang terkena salah satu anggota keluarga virus Hepatitis, dan dilihat dari panjang protein pasangan basa sebanyak 11.111 yang memenuhi syarat pengolahan data pada tugas MID mata kuliah Pengantar Bioinformatika pada hari Senin, 23 Maret 2020.

2.

- a) Panjang urutan DNA dari virus Hepatitis C, yaitu 11.111bp
Ukuran panjang urutan DNA dapat dilihat ketika program telah mengetahui data apa yang akan dibaca, yaitu data *hepatitis.fasta*, seperti pada gambar 1

```
#Fadhillah Putri Taha H071171301
#Gene Bank Data : Hepatitis C -- Hepatitis C virus full-length replicon pFGR-JFH1 RNA, complete sequence
#Dengan kode NCBI AB237837.1
#https://www.ncbi.nlm.nih.gov/nuccore/AB237837.1

library("seqinr")
hepatitisC <- read.fasta(file = "D:/dhil/Tugas/Bioinformatika/Bioinformatics/MID/hepatitisC.fasta")
hepatitisC_seq <- hepatitisC[[1]]
```

Gambar 1 Perintah Membaca Panjang urutan DNA virus Hepatitis C


Kemudian akan tampil *output* bahwa data telah dibaca oleh program R pada gambar

```
D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> library("seqinr")
> hepatitisC <- read.fasta(file = "D:/dhil/Tugas/Bioinformatika/Bioinformatics/MID/hepatitisC.fasta")
> hepatitisC_seq <- hepatitisC[[1]]
```

Gambar 2 Pembacaan data

Fadhillah Putri Taha
H071171301

- b) Dan *output* yang dihasilkan merupakan panjang urutan DNA virus Hepatitis C, sepanjang 11.111bp dengan menggunakan perintah *length(nama variabel)*. Dapat dilihat pada gambar 3.



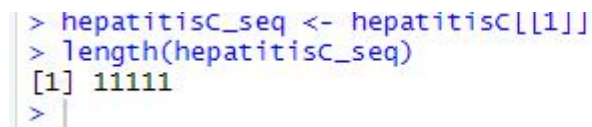
```

D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> GC(hepatitisC_seq)
[1] 0.5797858

```

Gambar 3 Panjang urutan DNA virus Hepatitis C

- c) Basis komposisi dari urutan DNA virus Hepatitis C, A C G T dapat dilihat pada gambar 4, menggunakan perintah *table(nama_variabel)*.

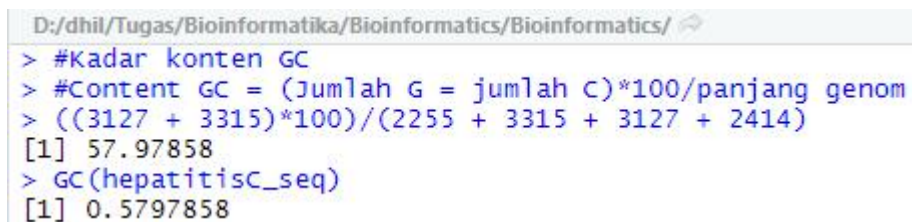


```

> hepatitisC_seq <- hepatitisC[[1]]
> length(hepatitisC_seq)
[1] 11111
>

```

- d) Kadar muatan GC dari *sequence* genom DNA virus Hepatitis C, sebanyak 0.5797858 dengan menggunakan perintah pada gambar 5.



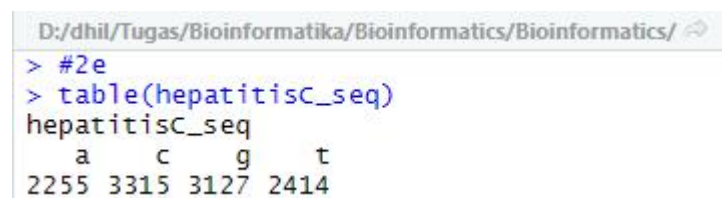
```

D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> #Kadar konten GC
> #Content GC = (Jumlah G = jumlah C)*100/panjang genom
> ((3127 + 3315)*100)/(2255 + 3315 + 3127 + 2414)
[1] 57.97858
> GC(hepatitisC_seq)
[1] 0.5797858

```

Gambar 5 Kadar muatan GC pada virus Hepatitis C

- e) Masing-masing nukleotida A, C, G, dan T secara berurutan berjumlah 2255, 3315, 3127, dan 2414. Dapat dilihat pada gambar 6 dengan perintah



```

D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> #2e
> table(hepatitisC_seq)
hepatitisC_seq
  a    c    g    t
2255 3315 3127 2414

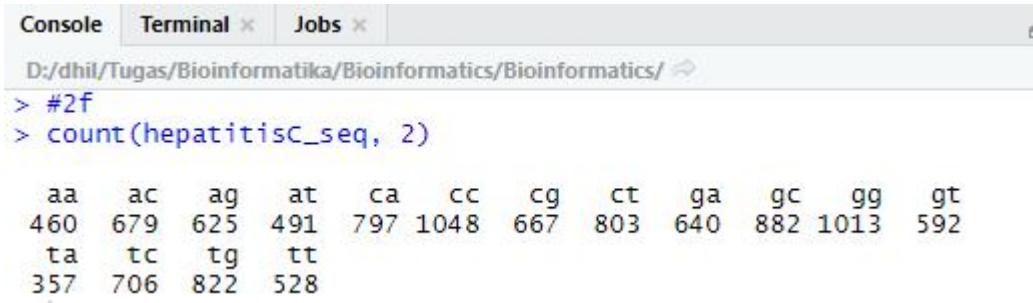
```

Gambar 6 Jumlah nukleotida A, C, G, dan T

table(nama_variabel).

Fadhillah Putri Taha
H071171301

- f) Kemunculan DNA CC muncul sebanyak 1048 kali, CG sebanyak 667 kali, dan GC sebanyak 882 kali. Dengan menggunakan perintah `count(nama_variabel, ukuran_pasangan)` dapat dilihat pada gambar 7.



```

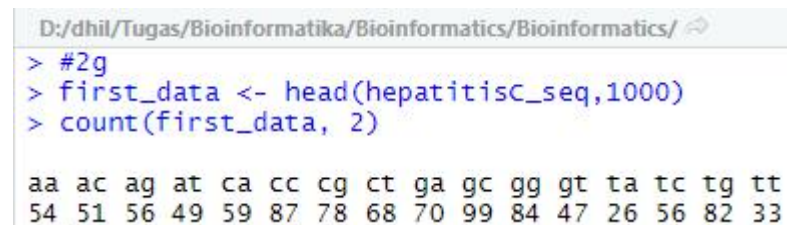
D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> #2f
> count(hepatitisc_seq, 2)

  aa  ac  ag  at  ca  cc  cg  ct  ga  gc  gg  gt
460 679 625 491 797 1048 667 803 640 882 1013 592
  ta  tc  tg  tt
357 706 822 528

```

Gambar 7 Jumlah DNA CC, CG, dan GC pada virus Hepatitis C

- g) Jumlah kemunculan kata-kata DNA CC, CG, dan GC pada 1000 nukleotida pertama dan 1000 nukleotida terakhir.
Berikut kemunculan kata-kata DNA CC, CG, dan GC pada 1000 nukleotida pertama. Kata-kata CC sebanyak 87, CG sebanyak 78, dan GC sebanyak 99 kata pada 1000 nukleotida pertama. Perintah dan *output* dapat dilihat pada gambar 8.



```

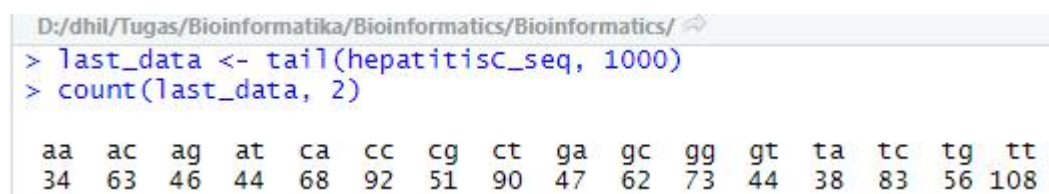
D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> #2g
> first_data <- head(hepatitisc_seq, 1000)
> count(first_data, 2)

  aa ac ag at ca cc cg ct ga gc gg gt ta tc tg tt
54 51 56 49 59 87 78 68 70 99 84 47 26 56 82 33

```

Gambar 8 Jumlah kata-kata CC, CG, dan GC pada 1000 nukleotida pertama

Berikut kemunculan kata-kata DNA CC, CG, dan GC pada 1000 nukleotida terakhir. Kata-kata CC sebanyak 92, CG sebanyak 51, dan GC sebanyak 62 kata pada 1000 nukleotida terakhir. Perintah dan *output* dapat dilihat pada gambar 9.



```

D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> last_data <- tail(hepatitisc_seq, 1000)
> count(last_data, 2)

  aa  ac  ag  at  ca  cc  cg  ct  ga  gc  gg  gt  ta  tc  tg  tt
34  63  46  44  68  92  51  90  47  62  73  44  38  83  56 108

```

Gambar 9 Kemunculan kata-kata DNA CC, CG, dan GC pada 1000 nukleotida terakhir

Fadhillah Putri Taha
H071171301

3. Soal nomor 1 bertujuan agar mahasiswa/penulis dapat jujur pada metode Pengambilan data sekunder, yang diambil pada data bank gen, yaitu NCBI. Dengan mengetahui data penyakit apa yang diambil, sehingga data yang diteliti benar merupakan sesuatu yang akan diteliti, bukan data asal atau Pengambilan secara acak tanpa tahu tujuan maksud dari data yang akan diteliti.

Soal nomor 2 bertujuan agar peneliti tahu benar mengenai data yang akan diteliti secara rinci, dan dalam pengolahannya. Baik, rincian dasar seperti panjang urutan pasangan basa berurutan, jumlah pasangan dengan 2 nukleotida, menghitung pasangan basa berurutan pada seribu urutan pertama dan terakhir, hingga banyak masing-masing protein tunggal A, C, G, dan T.

Fadhillah Putri Taha
H071171301

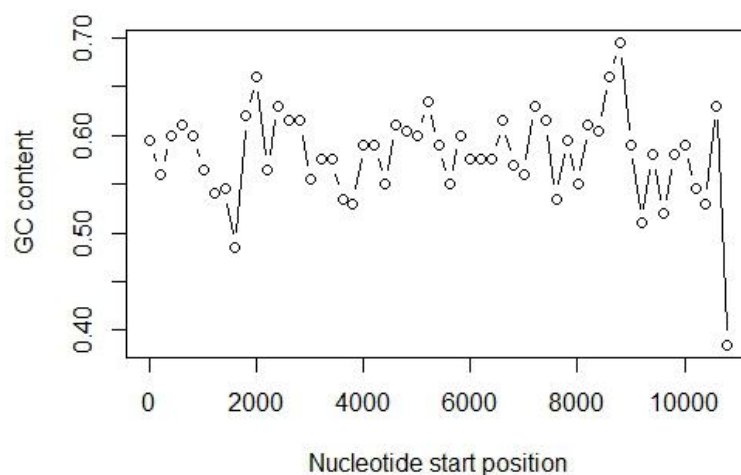
- Bagian B
- *Sliding plot* konten GC pada virus Hepatitis C dengan ukuran 200 nukleotida. Dimana fungsi dan perintah dapat dilihat pada gambar 10 dan plot dapat dilihat pada gambar 11 dan titik puncak mencapai 0.695 dan titik terendah senilai 0.385.

```
D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> # B_1
> slidingwindowplot <- function(windowsize, inputseq)
+ {
+   starts <- seq(1, length(inputseq)-windowsize, by = windowsize)
+   n <- length(starts)
+   hepatitisC_GC <- numeric(n)
+   for (i in 1:n) {
+     hepatitisC <- inputseq[starts[i]:(starts[i]+windowsize-1)]
+     hepatitisC_GC <- GC(hepatitisC)
+     hepatitisC_GC[i] <- hepatitisC_GC
+   }
+   plot(starts,hepatitisC_GC,type="b",xlab="Nucleotide start position",ylab="GC content")
+   cat("\npuncak= ", max(hepatitisC_GC))
+   cat("\npalung= ", min(hepatitisC_GC))
+ }
> #a sliding window plot with a window size of 50 nucleotides
> slidingwindowplot(50, hepatitisC_seq)
```

Gambar 11 Perintah membuat sliding plot konten GC

```
> #a sliding window plot with a window size of 200 nucleotides
> slidingwindowplot(200, hepatitisC_seq)
```

```
puncak= 0.695
palung= 0.385
```



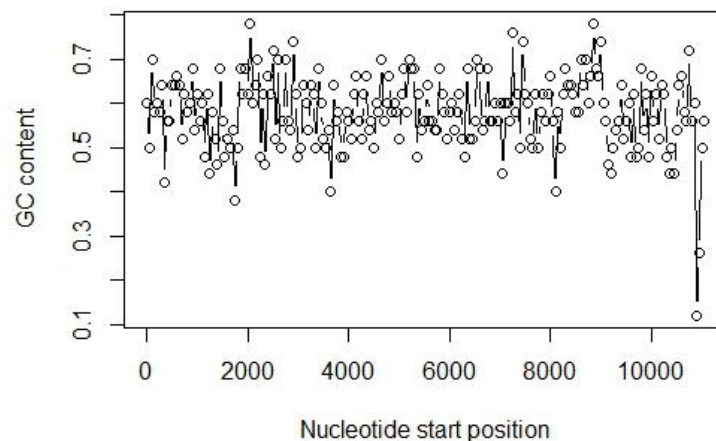
Gambar 10 Sliding plot dengan ukuran 200 nukleotida

Fadhillah Putri Taha
H071171301

- *Sliding plot* konten GC dalam urutan genom untuk virus Hepatitis C dengan ukuran 200 dapat dilakukan, sedangkan untuk ukuran 20.000 dan 200.000 tidak dapat dilakukan karena total panjang pasangan basa berurutan hanya 11.111. Sehingga penulis mengganti ukuran jendela sebesar 50, 5000, dan 11000. *Sliding plot* berukuran 50, 5000 dapat dilihat pada gambar 11, 12, dan 13.

```
> #a sliding window plot with a window size of 50 nucleotides
> slidingwindowplot(50, hepatitisC_seq)
```

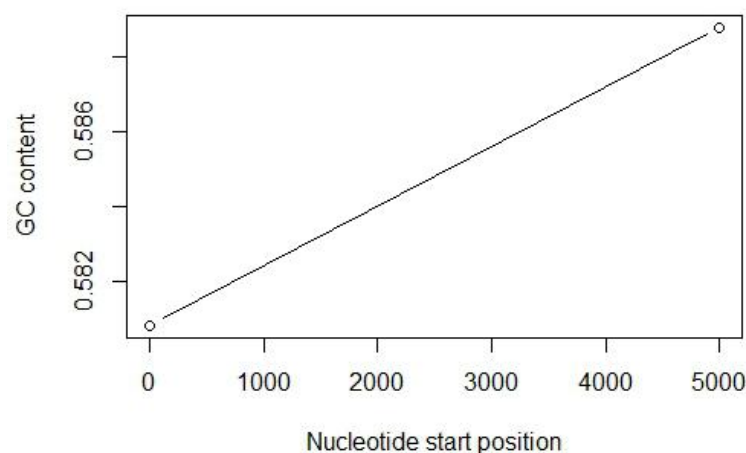
```
puncak= 0.78
palung= 0.12
```



Gambar 13 Sliding plot dengan window berukuran 50

```
> #a sliding window plot with a window size of 5000 nucleotides
> slidingwindowplot(5000, hepatitisC_seq)
```

```
puncak= 0.5888
palung= 0.5808
```



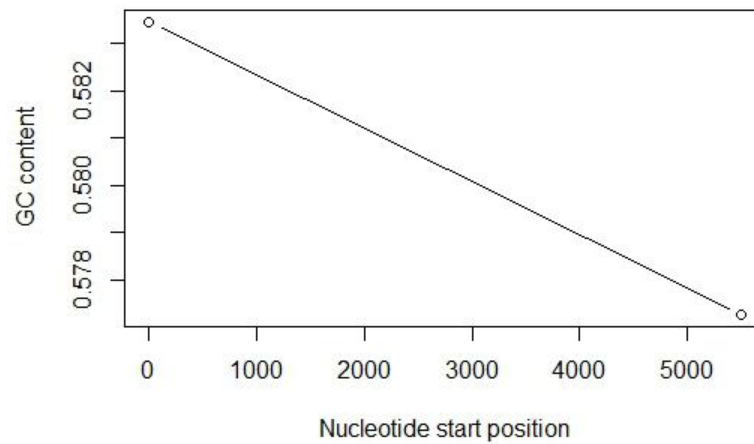
Gambar 12 Sliding plot dengan window berukuran 5000

-
-

Fadhillah Putri Taha
H071171301

```
> #a sliding window plot with a window size of 5500 nucleotides  
> slidingwindowplot(5500, hepatitisC_seq)
```

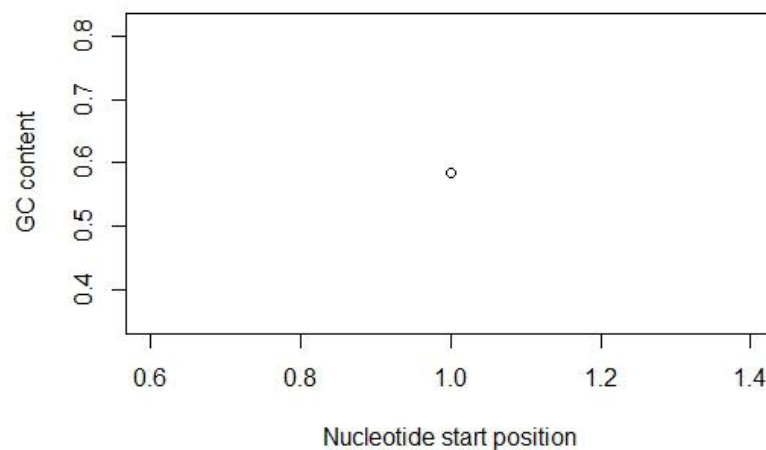
```
puncak= 0.5834545  
palung= 0.5772727
```



Gambar 15 Sliding plot dengan window berukuran 5500

```
> #a sliding window plot with a window size of 6000 nucleotides  
> slidingwindowplot(6000, hepatitisC_seq)
```

```
puncak= 0.5831667  
palung= 0.5831667
```



Gambar 14 Sliding plot dengan window berukuran 6000

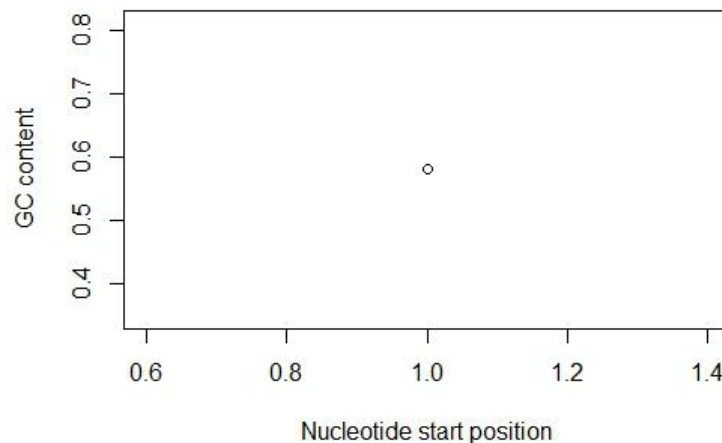
-
-

Fadhillah Putri Taha
H071171301

-

```
> #a sliding window plot with a window size of 11000 nucleotides
> slidingwindowplot(11000, hepatitisC_seq)
```

```
puncak= 0.5803636
palung= 0.5803636
```



Gambar 16 Sliding plot dengan window berukuran 11.000

-

-

- Fungsi menghitung konten AT urutan DNA pada virus Hepatitis C.

```
> #B_3
> AT <- function(inputseq)
+ {
+   mytable <- count(inputseq, 1) # make a table with the count of As, Cs, Ts, and Gs
+   mylength <- length(inputseq) # find the length of the whole sequence
+   myAs <- mytable[[1]] # number of As in the sequence
+   myTs <- mytable[[4]] # number of Ts in the sequence
+   myAT <- (myAs + myTs)/mylength
+   return(myAT)
+ }
```

Gambar 17 Fungsi penghitung konten AT

- Jumlah konten AT pada urutan DNA virus Hepatitis C.
- Hubungan antara konten AT dan konten GC pada genom Hepatitis C.

Konten AT harus kurang dari konten GC, dimana jika dilakukan operasi penjumlahan antara konten AC dan konten GC akan menghasilkan 1.

```
> GC(hepatitisC_seq)
[1] 0.5797858
> 0.4202142 + 0.5797858
[1] 1
```

Gambar 19 Jumlah AT dan GC = 1

hepatitis C

Fadhillah Putri Taha
H071171301

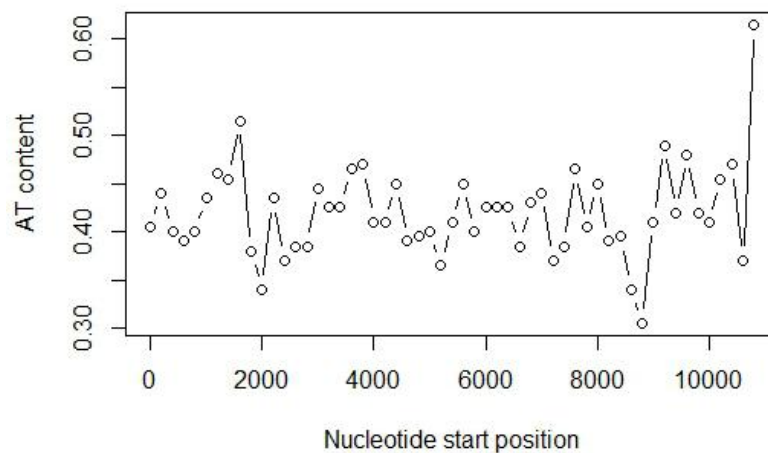
4. Fungsi menggambar *sliding plot* konten AT dengan panjang nukleotida 200.

```
> #B_4
> slidingwindowplotAT <- function(windowsize, inputseq)
+ {
+   starts <- seq(1, length(inputseq)-windowsize, by = windowsize)
+   n <- length(starts)
+   hepatitisC_ATs <- numeric(n)
+   for (i in 1:n) {
+     hepatitisC <- inputseq[starts[i]:(starts[i]+windowsize-1)]
+     hepatitisC_AT <- AT(hepatitisC)
+     hepatitisC_ATs[i] <- hepatitisC_AT
+   }
+   plot(starts,hepatitisC_ATs,type="b",xlab="Nucleotide start position",ylab="AT content")
+   cat("\npuncak= ", max(hepatitisC_ATs))
+   cat("\npalung= ", min(hepatitisC_ATs))
+ }
```

Gambar 20 Fungsi menggambar *sliding plot* konten AT pada virus Hepatitis C

```
> #a sliding window plot with a window size of 200 nucleotides
> slidingwindowplot(200, hepatitisC_seq)
```

```
puncak= 0.615
palung= 0.305
```



Gambar 21 *Sliding plot* konten AT dengan 200 nukleotida

Fadhillah Putri Taha
H071171301

5.

- Menghitung urutan 3 nukleotida

Kata 3 nukleotida GAC tidak muncul terlalu sering, namun memang kata GAC cukup banyak pada virus Hepatitis C ini. Untuk mengetahui modus kemunculan katanya, dapat

```
> #B_5
> count(hepatitisC_seq, 3)

aaa aac aag aat aca acc acg act aga agc
110 126 145 79 169 226 126 158 149 165
agg agt ata atc atg att caa cac cag cat
207 104 79 146 189 77 164 229 208 196
cca ccc ccg cct cga cgc cgg cgt cta ctc
269 317 199 263 113 192 212 150 125 241
ctg ctt gaa gac gag gat gca gcc gcg gct
257 180 132 199 196 113 173 298 201 210
gga ggc ggg ggt gta gtc gtg gtt taa tac
217 302 310 184 95 172 212 112 54 124
tag tat tca tcc tcg tct tga tgc tgg tgt
76 103 186 207 141 172 161 223 284 154
tta ttc ttg ttt
58 147 164 159
```

Gambar 22 Jumlah urutan GAC

```
> sum(count(hepatitisC_seq,3))
[1] 11109
```

Gambar 23 Total kata dengan 3 nukleotida

digunakan perintah `count(nama_variabel)` pada gambar 22.

- Jumlah kemunculan kata GACs sebanyak 199 kali dari 11109 yang dapat dilihat pada gambar 22 dan gambar 23. Sehingga frekuensi kemunculan GACs dapat dihitung dengan $199/11109$, dapat dilihat pada gambar 24 atau gambar 25.

```
D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> #B_5c
> count(hepatitisC_seq,1)

a      c      g      t
2255 3315 3127 2414
> 2255/11111 #frekuensi A
[1] 0.202952
> 3315/11111 #frekuensi C
[1] 0.298353
> 3127/11111 #frekuensi G
[1] 0.2814328
> 0.202952 * 0.298353 * 0.2814328 #frekuensi GAC
[1] 0.01704113
```

Gambar 24 Rincian frekuensi kemunculan GACs

```
> 199/11109
[1] 0.0179134
```

Gambar 25 Frekuensi kemunculan GACs

Fadhillah Putri Taha
H071171301

- Nilai *rho* untuk kata GACs

```
D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> #menghitung nilai rho
> (199/11109)/(0.202952 * 0.298353 * 0.2814328)
[1] 1.051186
```

Gambar 26 Nilai rho konten GACs

```
D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> #menghitung panjang rho untuk urutan 3 nukleotida
> rho(hepatitisC_seq, 3)
```

aaa	aac	aag	aat	aca	acc	acg	act	aga	agc
1.1845067	0.9229505	1.1259821	0.7946597	1.2379256	1.1261061	0.6655751	1.0811208	1.1570437	0.8715864
agg	agt	ata	atc	atg	att	caa	cac	cag	cat
1.1591844	0.7544077	0.7946597	0.9990103	1.3709909	0.7235260	1.2013006	1.1410544	1.0987271	1.3411372
cca	ccc	ccg	cct	cga	cgc	cgg	cgt	cta	ctc
1.3403652	1.0744671	0.7150602	1.2241535	0.5969046	0.6899073	0.8075716	0.7401625	0.8553171	1.1217529
ctg	ctt	gaa	gac	gag	gat	gca	gcc	gcg	gct
1.2681450	1.1505326	1.0250320	1.0511861	1.0975852	0.8196930	0.9138452	1.0707937	0.7656693	1.0362274
gga	ggc	ggg	ggg	gta	gtc	gtg	gtt	taa	tac
1.2151836	1.1504086	1.2518795	0.9625189	0.6891224	0.8487196	1.1089892	0.7589271	0.5431851	0.8484745
tag	tat	tca	tcc	tcg	tct	tga	tgc	tgg	tgt
0.5512979	0.9678334	1.2727118	0.9634973	0.6957527	1.0993978	1.1678812	1.1003749	1.4856270	1.0435248
tta	ttc	ttg	ttt						
0.5449936	0.9396016	1.1112862	1.3956284						

Gambar 27 Menghitung panjang rho untuk urutan 3 nukleotida

- Mengecek fungsi *rho*

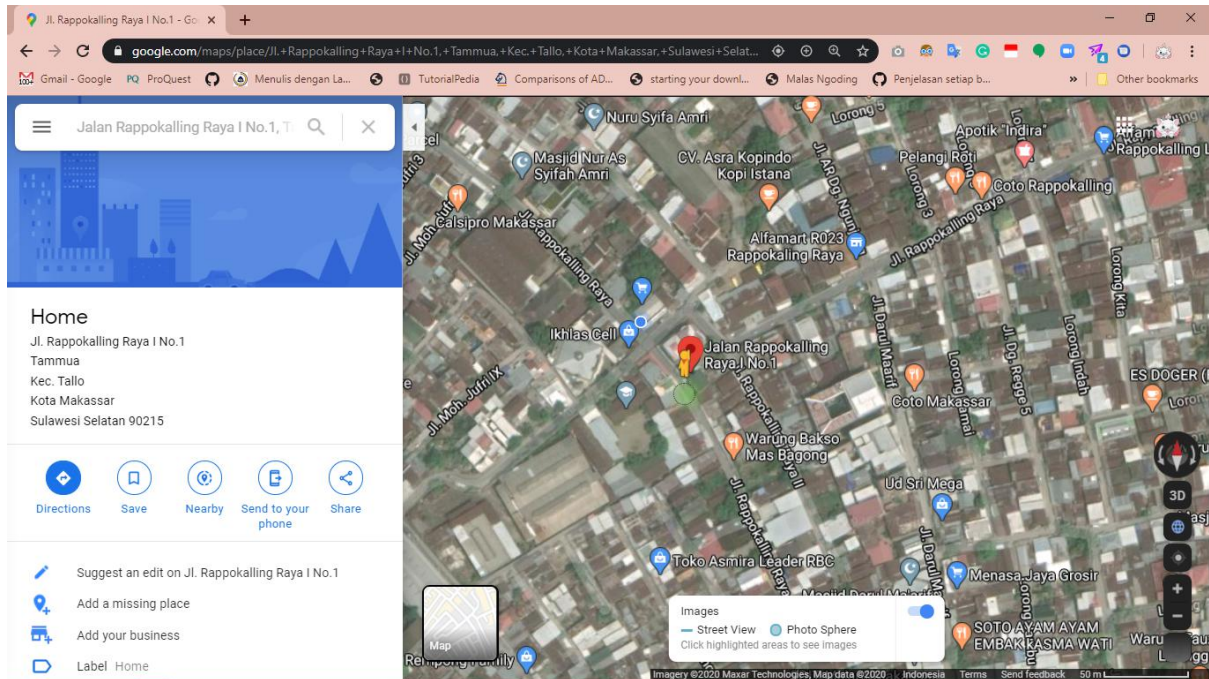
```
D:/dhil/Tugas/Bioinformatika/Bioinformatics/Bioinformatics/
> #5d
> help.search("rho")
> base::getHook
function (hookName)
get0(hookName, envir = .userHooksEnv, inherits = FALSE, ifnotfound = list())
<bytecode: 0x000001a88bb99f88>
<environment: namespace:base>
> seqinr::rho
function (sequence, wordsize = 2, alphabet = s2c("acgt"))
{
  wordcount <- count(sequence, wordsize, freq = FALSE, alphabet = alphabet)
  uni <- count(sequence, 1, freq = TRUE, alphabet = alphabet)
  expected_wordfreq <- function(wordsize, uni) {
    if (wordsize == 1)
      return(uni)
    else kronecker(uni, expected_wordfreq(wordsize - 1, uni))
  }
  expected_wordcount <- sum(wordcount) * expected_wordfreq(wordsize,
    uni)
  return(wordcount/expected_wordcount)
}
<bytecode: 0x000001a89aed46e8>
<environment: namespace:seqinr>
```

Gambar 28 Mengecek fungsi rho

Fadhillah Putri Taha
H071171301

- Bagian C

Tugas dikerjakan di kediaman Bapak Salihu Taha yang beralamat di Jl. Rappokalling Raya I. Lokasi dapat dilihat dengan mengunjungi alamat https://bit.ly/STaha_House atau



<https://maps.app.goo.gl/yGnnjx4B64B3hHoU6>