WEEK 12. PRACTICE

Finding genes: going ashore at CpG islands

Basic Shell Commands

```
$ cd [User_Folder]
$ mkdir Session12
$ cd Session12
```

Basic Shell Commands

```
$ time
```

```
Wed May 25 14:14:19 [kyoungwoo@biglab-master Chapter15]$ time

real    0m0.000s
user    0m0.000s
sys     0m0.000s
```

```
$ time python [script]
```

```
Wed May 25 14:16:13 [kyoungwoo@biglab-master Chapter15]$ time python cpg.py short.fa > normal

real    0m0.105s
user    0m0.025s
sys     0m0.006s
Wed May 25 14:16:36 [kyoungwoo@biglab-master Chapter15]$
```

Basic Shell Commands

```
$ date                              # Current time and date
```

```
[biguser@biglab-master ~]$ date
Mon Nov 19 22:57:21 KST 2018
```

```
$ date '+%F %r'                     # %F : YYYY-MM-DD
                                    # %r : 12 hour format
```

```
[biguser@biglab-master ~]$ date '+%F %r'
2018-11-19 10:57:58 PM
```

```
$ date '+%Y-%m-%d'                  # YYYY-MM-DD (same as%F)
```

```
[biguser@biglab-master ~]$ date '+%Y-%m-%d'
2018-11-19
```

Basic Shell Commands

```
$ time python [script] > $(date '+%Y-%m-%d').log
```

```
Wed May 25 14:16:36 [kyoungwoo@biglab-master Chapter15]$ time python cpg.py short.fa > $(date '+%Y-%m-%d').log

real    0m0.067s
user    0m0.021s
sys     0m0.009s
Wed May 25 14:18:43 [kyoungwoo@biglab-master Chapter15]$ ll
total 19716
-rw-rw-r-- 1 kyoungwoo kyoungwoo   30999 May 25 14:18 2016-05-25.log
```

Code 15.1

cpg.py

```
#!/usr/bin/python
import sys
import re

win = 500
step = 10
seq = ''

input_file = sys.argv[1]
for line in open(input_file):  # a shorter sequence
    if not re.search('>', line):
        line = line.rstrip()
        seq = seq + line

print 'pos\tcpg\tcg_ratio\tcg_obs_exp'

for i in range(0, len(seq) - win+1, step):

    testseq = seq[i:i + win]
    c = float(testseq.count('C'))
    g = float(testseq.count('G'))
    cg = float(testseq.count('CG'))
    cg_ratio = (c + g) * 100 / len(testseq)
    cg_obs_exp = cg * len(testseq) / (c * g)
    pos = i + win / 2
    if cg_ratio >= 55 and cg_obs_exp >= 0.65:
        print str(pos)+ '\t'+ "1"+ '\t'+ str(cg_ratio)+'\t'+ str(cg_obs_exp)
    else:
        print str(pos)+ '\t'+ "0"+ '\t'+ str(cg_ratio)+'\t'+ str(cg_obs_exp)
```

$$\frac{\frac{cg}{len(testseq)}}{\frac{c}{len(testseq)} \times \frac{g}{len(testseq)}}$$

CG ratio → Observed ratio

C ratio × G ratio → Expected ratio

Code 15.1
cpg.py

```
$ cp /home/biguser/tutor/Session12/short.fa .
$ python cpg.py short.fa > cpg.short.out
$ less cpg.short.out
```

pos	cpg	cg_ratio	cg_obs_exp
250	0	36.2	0.133547008547
260	0	36.4	0.131492439185
270	0	36.8	0.127567291746
280	0	37.0	0.189729319504
290	0	37.4	0.18315018315
300	0	36.6	0.191815856777
310	0	36.8	0.192604006163
320	0	37.0	0.192307692308
330	0	37.2	0.192110655738
340	0	37.2	0.256147540984
350	0	37.6	0.248385494287
360	0	38.4	0.294568163073
370	0	38.2	0.296982656213
380	0	38.2	0.298900047824
390	0	38.2	0.296982656213
400	0	38.6	0.290360046458
410	0	38.0	0.297619047619

Visualization of CPG landscape with R

```
$ cp /home/biguser/tutor/Session12/cpg_short.r .
$ vi cpg_short.r
```

```
# plot the results of cpg island prediction

# define some colours
rgb <- c("#000000", "#E69F00", "#56B4E9", "#009E73",
    "#F0E442", "#0072B2", "#D55E00", "#CC79A7")

# read file being output from Perl script
data <- read.table("cpg_short.out", sep = "\t", header = TRUE)

# Specify the number of lines of margin to the four sides
# of the plot. In this case we want to make room for text
# at the right axis.

pdf("cpg_plot.pdf")
par(mar = c(5, 4, 4, 5) + 0.1)

# make first plot, which is empty
plot(data$pos, data$cg_obs_exp, type = "n", xaxt = "n",
    yaxt = "n", xlab = "", ylab = "")

# plot lines to indicate where CpG islands are predicted
for (i in 1:length(data$pos)) {
    if (data$cpg[i] == 1) {
        lines(c(data$pos[i], data$pos[i]), c(0, 1), col = rgb[2])
    }
```

⋮

Visualization of CPG landscape with R

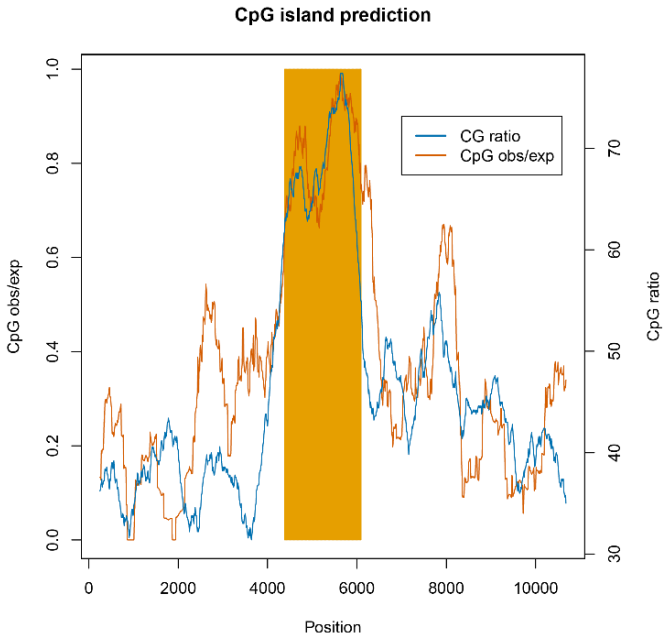
```
# before another plot, prevent R from clearing
# the graphics device
par(new = TRUE)

# make second plot with the cg_obs_exp data
plot(data$pos, data$cg_obs_exp, type = "l", main = "CpG island prediction",
    xlab = "Position", ylab = "CpG obs/exp", col = rgb[7])

# print legend
legend(7000, 0.9, c("CG ratio", "CpG obs/exp"), col = c(rgb[6],
    rgb[7]), lwd = 2)

par(new = TRUE)

# make third plot
plot(data$pos, data$cg_ratio, type = "l", xaxt = "n",
    yaxt = "n", xlab = "", ylab = "", col = rgb[6])

# print ticks for the 2nd y axis
axis(4)

# print text to 2nd y axis
mtext("CpG ratio", side = 4, line = 3)
dev.off()
```

Visualization of CPG landscape with R

```
$ Rscript cpg_short.r
```

```
[kyungtae@biglab-master scripts]$ Rscript cpg_short.r
null device
          1
```

cpg_plot.pdf



Visualization of CPG landscape with R

```
$ cp /home/biguser/tutor/Session12/chr4_region.fa .
$ python cpg.py chr4_region.fa > cpg_chr4.out
$ cp /home/biguser/tutor/Session12/cpg_chr4.r .
$ cp /home/biguser/tutor/Session12/chr4_annotation.txt .
$ less -S chr4_annotation.txt
```

chr	name	category	beg	end	z	strand	none	gene_id			
chr4	hg19_knownGene	exon	72053003	72053118	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72102293	72102366	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72120937	72121116	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72205087	72205222	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72215629	72215789	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72222725	72222904	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72263294	72263370	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72306333	72306490	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72313363	72313450	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72316106	72316260	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72316905	72317018	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72319212	72319386	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72332161	72332294	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72338416	72338687	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72352665	72352735	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72363218	72363409	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72397779	72397892	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72399944	72400105	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72412067	72412245	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72413365	72413437	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";
chr4	hg19_knownGene	exon	72420857	72420925	0.000000	+	.	gene_id "uc003hfy.2"; transcript_id "uc003hfy.2";

Visualization of CPG landscape with R

Step 1: CpG island plot

```
$ vi cpg_chr4.r
```

```
# plot the results of CpG island prediction

#define some rgb colours
rgb <- c("#000000", "#E69F00", "#56B4E9", "#009E73",
    "#F0E442", "#0072B2", "#D55E00", "#CC79A7")

# read data which is output of Perl script
data <- read.table("cpg_chr4.out", sep = "\t", header = TRUE)

# make an emtpy plot
plot(0, type = "n", xlim = c(72009075, 72009075 +
    1520980), ylim = c(1.6, 2.3), xlab = "Position", ylab = "",
    yaxt = "n", main = "CpG island prediction")

# plot the predicted CpG islands
for (i in 1:length(data$pos)) {
    if (data$cpg[i] == 1) {
        # convert the position numbers to chromosomal positions
        data$pos[i] <- data$pos[i] + 72009075
        #print (data$pos[i])
        lines(c(data$pos[i], data$pos[i]), c(1.7, 1.8), col = rgb[7])
    }
}
```

Visualization of CPG landscape with R

Step 2: Gene structure plot

```
# Read file with chr4:72,009,075-73,530,562 region
# annotation. This information was obtained with
# the Table Browser of the UCSC browser

annot <- read.table("chr4_annotation.txt", sep = "\t",
    header = TRUE)

color <- 1
prevname <- ""
lines <- length(annot$chr)  # number of lines in the annotation file

for (i in (1:lines)) {

    if (annot$category[i] == "exon") {
        # if we consider an exon
        # if it a different gene as compared to the previous line,
        # change the colour
        if (annot$gene_id[i] != prevname) {
            color <- color + 1
        }
        prevname <- annot$gene_id[i]

        # draw rectangles for the exons
        rect(annot$beg[i], 1.9, annot$end[i], 2.1, border = rgb[color],
            col = rgb[color])
    }
```

rect() : plot region 안에 네모 모양(상자)을 그리는 함수.

```
function (xleft, ybottom, xright, ytop, density = NULL, angle = 45,
 col = NA, border = NULL, lty = par("lty"), lwd = par("lwd"), ...)
```

xleft : 사각형의 왼쪽 x좌표.
ybottom : 사각형의 아래쪽 y좌표.
xright : 사각형의 오른쪽 x좌표.
ybottom : 사각형의 위쪽 y좌표.
col : 사각형의 내부 색상.
border : 사각형의 테두리 색상.
lty : 선의 종류.(테두리 및 내부 빗금)
lwd : 선의 굵기.(테두리 및 내부 빗금)
density : 내부 선들의 밀도.(내부 빗금)
angle : 내부 선들의 기울기. (내부 빗금. default=45)
main : plot의 제목, 이름.

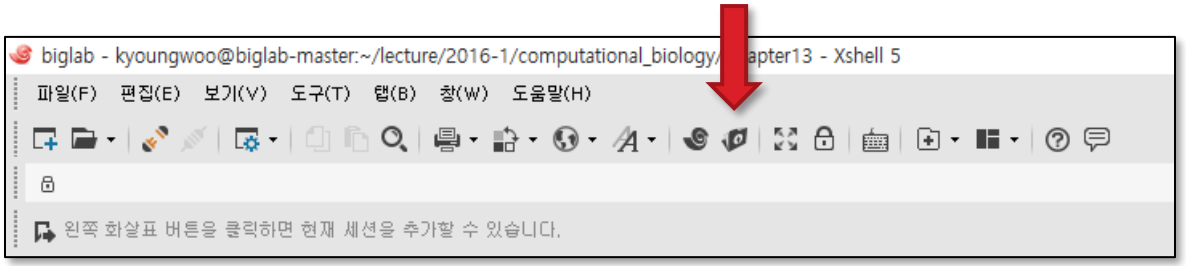
Visualization of CPG landscape with R

Step 2: Gene structure plot

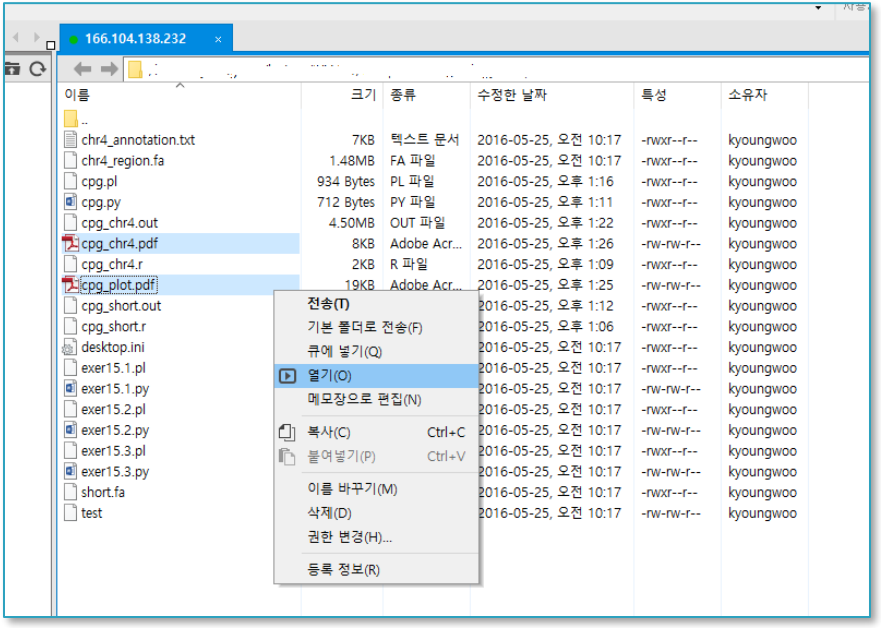
```
    if (annot$category[i] == "trans") {
        # to identify the end points of transcripts
        lines(c(annot$beg[i], annot$end[i]), c(2, 2), col = "grey",
            lw = 2)
        direction <- annot$strand[i]
        if (direction == "+") {
            dir <- 2
        }
        if (direction == "-") {
            dir <- 1
        }
        # print arrows to indicate the location
        # and direction of transcript

        arrows(annot$beg[i], 1.85, annot$end[i], 1.85, col = "grey",
            code = dir, lw = 5, length = 0.1)
    }
}

# print names of genes (this information
# can not be extracted from the
# chr4_annotation.txt file)
text(72009075 + 2e+05, 2.2, "SLC4A4")
text(72009075 + 610000, 2.2, "GC")
text(72009075 + 950000, 2.2, "NPFFR2")
text(72009075 + 1300000, 2.2, "ADAMTS3")
```

Visualization of CPG landscape with R

```
$ Rscript cpg_chr4.r
```



Visualization of CPG landscape with R



Mouse right click and click open from “cpg_plot.pdf” and “cpg_chr4.pdf” file

Visualization of CPG landscape with R

cpg_chr4.pdf