Analiza transkryptomu - projekt

Justyna Ostapiuk

Celem projektu było wykonanie analizy RNAseq dla bioprojektu PRJNA313294. Do tego celu wykorzystano dwie próbki - zika (zakażone), będące czynnikiem badanym, dwie próbki - mock (zakażone), będace próbą kontrola, komórki ludzkie - neutral progenitorl cell (hNPCc). Ogólny projekt badał czy zakażenie ZIKV zwiększa śmiertelność komórek i zaburza postęp cyklu komórkowego, powodując osłabiony wzrost komórek ludzkich hNPC oraz czy analiza sekwencjonowania RNA zainfekowanych hNPC ujawnia zaburzenia regulacji transkrypcji. Celem analizy danych z powyższego projektu jest przeanalizowanie jakie zmiany transkryptomiczne zostały wywołane czynnikiem badanym, czyli wirusem ZIKV w ludzkich liniach komórkowych. Posłużono się odczytami pair end z urządzenia MiSeq oraz odczytami single end z urządzenia NextSeq.

	SINGLE - NextSeq	PAIRED - MiSeq
ZIKA	SRR3194430, SRR3194431	SRR3191544, SRR3191545
MOCK	SRR3194428, SRR3194429	SRR3191542, SRR3191543

Całość projektu została wykonana lokalnie na komputerze (zbyt mała ilość miejsca na serwerze oraz spowolnienie serwera uniemożliwiło pracę). Kolejno przeprowadzone etapy projektu:

1. Pobieranie danych z bazy SRA

Pierwszym etapem projektu było utworzenie pliku SRR_ACC_List.txt zawierającego numery SRA dla bioprojektu PRJNA313294. Do pobrania danych z archiwum użyto narzędzia SRA Toolkit, w szczególności polecenia fastq-dump. Uruchomiono przygotowany skrypt do pobierania odczytów z plikiem SRR_ACC_List.txt . Polecenie split-files umożliwiło rozdzielenie danych pair end na oddzielne odczyty. Za pomocą polecenia sra-stat wygenerowano raport o danych w postaci pliku XML.

```
#!/bin/bash
while IFS='' read -r line || [[ -n "$line" ]]; do
AR=${line};
echo ${AR};
echo "*** Pobieranie: ${AR}";
fastq-dump -X 10 ${AR} --split-files
done < $1
```

2. Kontrola jakości plików fastq QC, filtrowanie odczytów

W drugim etapie dokonano kontroli jakości danych za pomocą programu FastQC.

```
#!/bin/bash
for j in home/justyna/projekt/SRR/*.fastq
do
fastqc $j -o
done
```

Wykres 1. Wykres jakości odczytów przed oczyszczaniem (single-end)

Wykres 2. Wykres jakości odczytów przed oczyszczaniem (paired-end)

Dane okazały się być dobrej jakości. Pojawił się tylko problem w zakładce „Per Base Sequence Content”. Na końcach 3’ zaobserwowane zostały odchylenia w rozkładzie nukleotydów. W wyniku

tego odczyty zostały przycięte w tym miejscu. Ponieważ zostały wykryte również nadreprezentowane sekwencje w odczytach, z wysoką zawartością N, z tego powodu uwzględniono ten parametr również podczas czyszczenia danych. Celem edycji danych posłużono się programem Trimmomatic.

```
#!/bin/bash

for i in SRR*_1.fastq
do
ii=$(basename $i | sed 's/_1/_2/')
output1=$i.paired.fastq
output2=$i.unpaired.fastq
output3=$ii.paired.fastq
output4=$ii.unpaired.fastq
java -jar trimmomatic-0.39.jar PE -phred33 $i $ii $output1 $output2 $output3 $output4
ILLUMINACLIP:hg19.fa:2:30:5 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:20 MINLEN:36
done
```

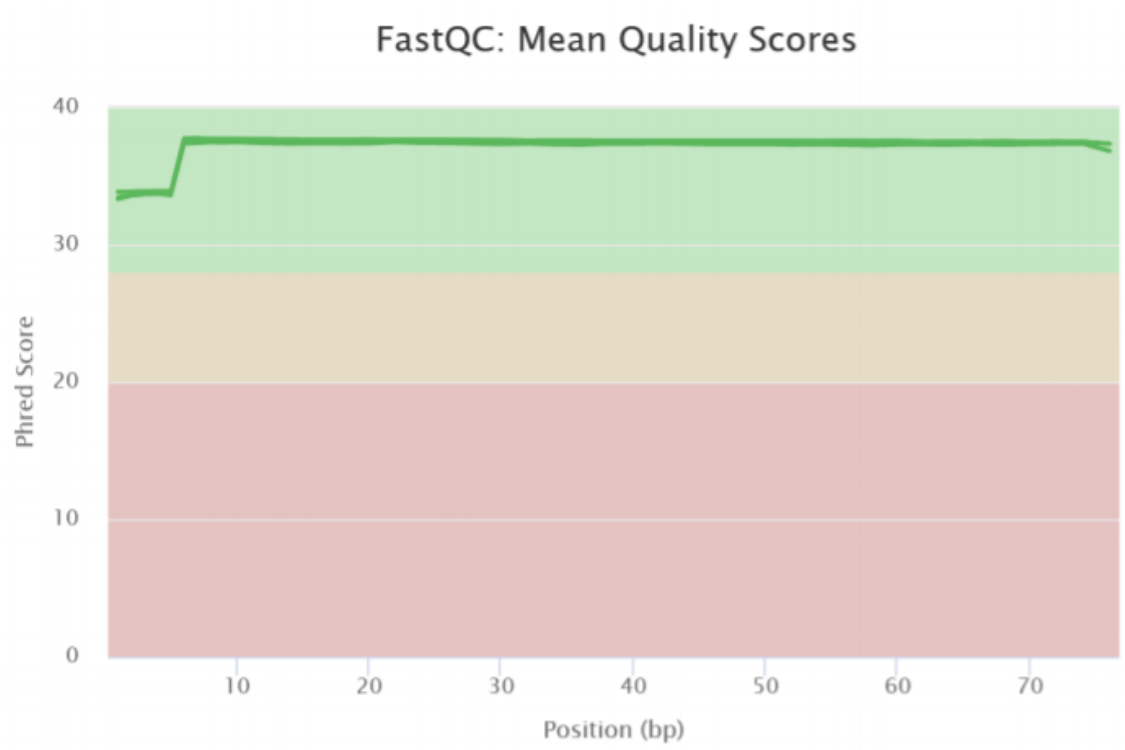
Użyto następujących ustawień:

- LEADING:3 - Usuwa wszystkie zasady poniżej progu jakości (3). Zaczynając od końca 5’
- TRAILING:3 - Usuwa wszystkie zasady poniżej progu jakości (3). Zaczynając od końca 3’
- SLIDINGWINDOW:4:20 - ‘Wędruje’ po sekwencji, odcinając, gdy średnia jakość w ‘oknie’ spadnie poniżej progu (20), zaczyna skanowanie od końca 5’. Parametr windowSize - ‘wielkość okna’ określa liczbę zasad do uśrednienia (4)
- MINLEN:35 - Usunie odczyty, jeśli są poniżej określonej długości (35)

W przypadku danych paired end do dalszej analizy wykorzystano tylko pliki sparowane (z czterech plików wynikowych trimmomatica dla danych paired-end).

Wykres 3. Wykres jakości odczytów przed oczyszczaniem (single-end)

Wykres 4. Wykres jakości odczytów przed oczyszczaniem (single-end)



3. Mapowanie odczytów

W trzecim etapie po pobraniu genomu referencyjnego hg19 (GRCh37), dokonano mapownia za pomocą programu hisat2.

```
#!/bin/bash

for i in home/justyna/projekt/trimmomatic/MISEQ/*_1_paired.fastq.gz;

do

plik1=$i

plik2=$(echo $i | sed 's/_1/_2')

hisat2 --dpad --gbar --mp 6,2 --np 1 --n-ceil L,0,0.15 --no-mixed --no-discordant
-x /home/justyna/projekt/mapowanie/index -1 $plik1 -2 $plik2 -S
MISEQ/$i.out.sam

done


for j in home/justyna/projekt/trimmomatic/NEXTSEQ/*.fastq.gz;

do

plik3=$j

hisat2 --dpad --gbar --mp 6,2 --np 1 --n-ceil L,0,0.15 --no-mixed --no-
discordant -x /home/justyna/projekt/mapowanie/index -U $plik3 -S
SRR_SAM/$j.out.sam;

done

#!/bin/bash

wget --timestamping
'ftp://hgdownload.cse.ucsc.edu/goldenPath/hg19/bigZips/hg19.fa.gz' -P mapowanie cd
mapowanie
gunzip hg19.fa.gz hg19.fa

hisat2-build hg19.fa index
```

Zastosowano zarówno program w trybie Paired-end przy wykorzystaniu slidingwindow. Tak samo zastosowano program dla danych Single-End, jednak zdecydowano o zachowaniu
odczytów o większej długości niż dane Paired-end. Ponownie wykonano kontrolę jakości za pomocą FastQC, by zobaczyć czy Trimmomatic poprawnie dokonał edycji danych. do obóbki danych (sortowania i indeksowania plików BAM) użyto narzędzia samtools.

```
#!/bin/bash
for i in home/justyna/projekt/trimmomatic/MISEQ/*.bam;
do
samtools view -Sb -@ 2 ${i}>${i}.bam
```

```
samtools sort ${i}.bam -o ${i}.sorted.bam
samtools index ${i}.sorted.bam
done

for i in home/justyna/projekt/trimmomatic/NEXTSEQ/*.bam;
do
samtools view -Sb -@ 2 ${i}>${i}.bam
samtools sort ${i}.bam -o ${i}.sorted.bam
samtools index ${i}.sorted.bam
done
```

4. Zliczanie odczytów

W tym etapie posłużono się funkcją featurecounts. Cały proces przeprowadzono za pomocą Rstudio, używając do tego celu biblioteki Rsubread .

```
# wczytanie biblioteki
library(Rsubread)
# wczytanie plików bam z danymi
NextSeq_ZIKA_1 <- 'home/justyna/projekt/BAM/SRR3194430.bam'
NextSeq_ZIKA_2 <- 'home/justyna/projekt/BAM/SRR3194431.bam'
NextSeq_MOCK_1 <- 'home/justyna/projekt/BAM/SRR3194428.bam'
NextSeq_MOCK_2 <- 'home/justyna/projektBAM/SRR3194429.bam'
MiSeq_ZIKA_1 <- 'home/justyna/projekt/BAM/SRR3191544.bam'
MiSeq_ZIKA_2 <- 'home/justyna/projekt/BAM/SRR3191545.bam'
MiSeq_MOCK_1 <- 'home/justyna/projekt/BAM/SRR3191542.bam'
MiSeq_MOCK_2 <- 'home/justyna/projekt/BAM/SRR3191543.bam'
# wczytanie genomu referencyjnego
hg19 <- 'home/justyna/projekt/hg19.gtf'
```

```
# zliczanie odczytów dla obu urządzeń
zliczanie_NextSeq <- featureCounts(files=c(NextSeq_ZIKA_1, NextSeq_ZIKA_2, NextSeq_MOCK_1,
NextSeq_MOCK_2), annot.ext = hg19, isGTFAnnotationFile = T)
zliczanie_MiSeq <- featureCounts(files=c( MiSeq_ZIKA_1, MiSeq_ZIKA_2, MiSeq_MOCK_1,
MiSeq_MOCK_2), annot.ext = hg19, isGTFAnnotationFile = T)
```

Celem ułatwienia dalszej analizy utworzono pliki z rozszerzeniem csv. Dzięki temu dalszy odczyt będzie prawidłowy.

```
# stworzenie plików csv
write.csv (zliczanie_NextSeq $counts, '~/projekt/R/NextSeq.csv')

write.csv(zliczanie_MiSeq$counts, '~/projekt/R/MiSeq.csv')
```

Wczytanie plików csv.

```
NextSeq <- read.csv('~/projekt/R/NextSeq.csv', row.names = 1)

MiSeq <- read.csv('~/projekt/R/MiSeq.csv', row.names = 1)
```

5. Analiza różnicowa eksresji

W tym etapie posłużoną się biblioteką DESeq2.

```
library(DESeq2)

sampleNS <- names(NEXTSEQ)

sampleMiS <- names(MISEQ)

cond_1 = rep('CASE', 2)

cond_2 = rep('reference', 2)

condition = factor(c(cond_1, cond_2))

col_NextSeq = data.frame(samples=sampleNS, condition=condition)

col_MiSeq = data.frame(samples=sampleMiS, condition=condition)

dds_NextSeq = DESeqDataSetFromMatrix(countData=NextSeq, colData=col_NextSeq, design =
~condition)
dds_NextSeq = estimateSizeFactors(dds_NextSeq)
dds_MiSeq = DESeqDataSetFromMatrix(countData=MiSeq, colData=col_MiSeq, design =
~condition)
dds_MiSeq <- estimateSizeFactors(dds_MiSeq)

# Normalizacja danych

log_NextSeq <- rlog(dds_NextSeq)

norm_NextSeq <-assay(log_NextSeq)

norm_NextSeq <- as.data.frame(norm_NextSeq)
# zapis danych po normalizacji do pliku csv

write.csv(norm_NextSeq, '~/projekt/R/NextSeq_norm.csv')

log_MiSeq <- rlog(dds_MiSeq)

norm_MiSeq<-assay(log_MiSeq)

norm_MiSeq <- as.data.frame(norm_MiSeq)
```

```
write.csv(norm_MiSeq, '~/projekt/R/MiSeq_norm.csv')

# Analiza za pomocą funkcji DESeq

porownanie_NS <- DESeq(dds_NextSeq)

porownanie_MS <- DESeq(dds_MiSeq)

# Wyniki

wynik_NS <- results(porownanie_NS)

wynik_MS <- results(porownanie_MS)

# Filtrowanie wyników
wynik_NS <- wynik_NS[wynik_NS$baseMean != 0,]
wynik_MS <- wynik_MS[wynik_MS$baseMean != 0,]
różnica_NS <- wynik_NS[wynik_NS$pvalue < 0.05,]
różnica_MS <- wynik_MS[wynik_MS$pvalue < 0.05,]

# Usuwanie NA
różnica_NS <- na.omit(różnica_NS)
różnica_MS <- na.omit(różnica_MS)

# Filtrowanie danych na podstawie wartości padj (p-value z poprawką Benjamina-Hochberga na wielokrotne testowanie)
Jako istotne wyniki zostały przyjęte te, których p-value 0.05. Drugim kryterium była różnica ekspresji. Uwzględnione zostały tylko te geny, których wartości bezwzględne z logarytmu FoldChange były przynajmniej na 2 poziomie.

różnica_NS <- różnica_NS[różnica_NS$padj < 0.05,]
różnica_MS <- różnica_MS[różnica_MS$padj < 0.05,]
```

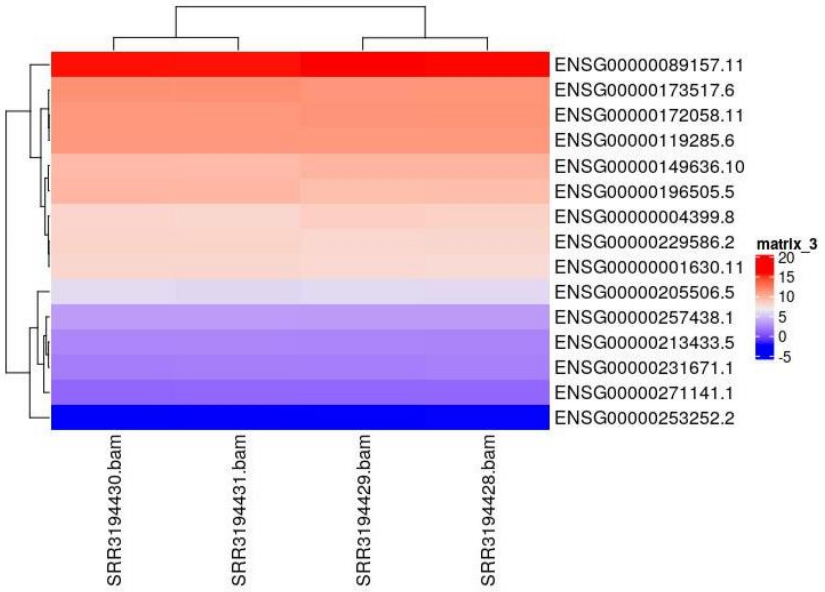
6. Porównanie wyników uzyskanych przez NextSeq oraz MiSeq (PCA oraz heatmapa)

Na tym etapie pracy w celu lepszej wizualizacji w porównaniu wyników sekwencjonowania uzyskanych za pomocą urządzeń NextSeq oraz MiSeq wykonano heatmapę oraz PCA.
Podczas tworzenia heatmapy wykorzystano bibliotekę ComplexHeatmap. Pracę rozpoczęto od wylosowania 15 genów.

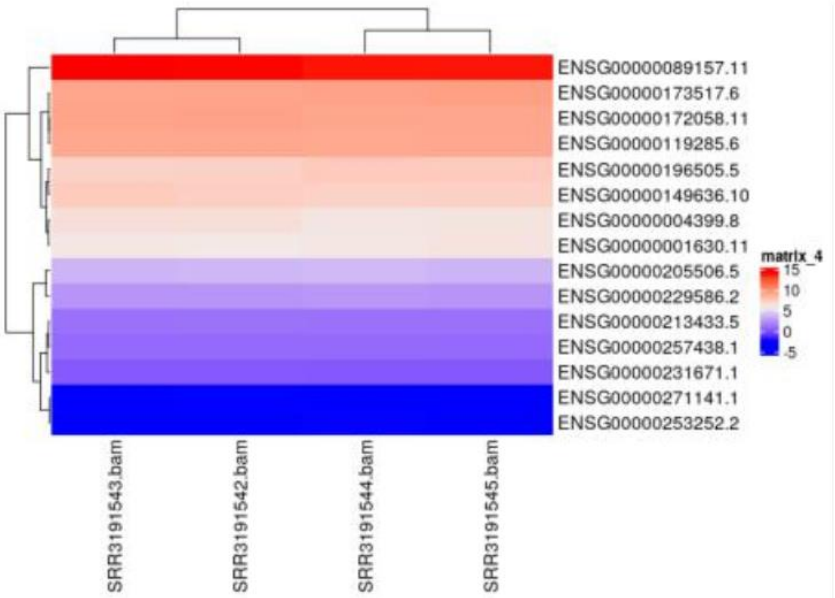
```
library (ComplexHeatmap)

g1 <- row.names(MiSeq_norm[MiSeq_norm$SRR3191544.bam!=0,])
g2 <- row.names(NextSeq_norm[NextSeq_norm$SRR3194430.bam!=0,])


geny_1i2 <- Reduce(intersect, list(g1,g2))

losowe_geny <- sample(geny_1i2, 15, replace=FALSE, prob=NULL)
```

geny_NextSeq <-
NextSeq_norm[row.names(NextSeq_norm)%in%losowe_geny,] geny_MiSeq <-
MiSeq_norm[row.names(MiSeq_norm)%in%losowe_geny,]

Heatmap(geny_NextSeq)



Heatmap(geny_MiSeq)

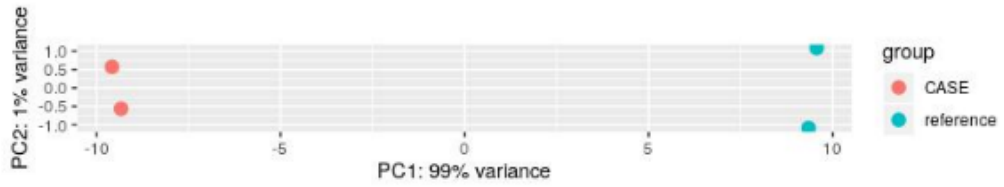


Heatmapy pokazują, że pomiędzy próbkami (NextSeq oraz MiSeq) występują podobieństwa. Wykonano również PCA, czyli Analizę Składowych Głównych. Posłużyła ona również do porównania obu metod sekwencjonowania.

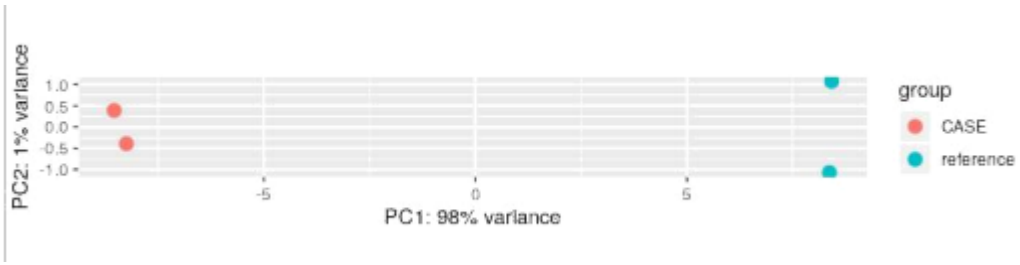
PCA_NextSeq <- rlogTransformation(NextSeq_porownanie)

PCA_MiSeq <- rlogTransformation(MiSeq_porownanie)

DESeq2::plotPCA(PCA_NextSeq)



DESeq2::plotPCA(PCA_MiSeq)



Widzimy, że powstały dwa osobne skupiska. Świadczy to o różnicy pomiędzy single oraz paired end. Dzięki temu możemy również zaobserwować podobieństwa pomiędzy próbkami pochodzącymi z tego samego urządzenia, zarówno dla MiSeq oraz NextSeq.

Wnioski: wyniki uzyskane obiema metodami różnią się od siebie. Niektóre geny się pokrywają, a niektóre są zupełnie różne. Są obserwowane różnice w metodzie sekwencjonowania oraz rodzajowi próbki (przypadek, kontrola). Są to różnice dosyć niewielkie. Dowodzą tego powyższe ryciny – heatmapy oraz PCA.