Genomics en Transcriptomics analyse muis B-cellen

Floris Menninga

2024-10-07

Table of Contents

# Libraries  
library(tidyverse)  
library(ggplot2)

### **Inleiding:**

Het gekozen artikel: (<https://pubmed.ncbi.nlm.nih.gov/38866970/>) heeft betrekking tot de verandering in 3d conformatie van chromatine die leiden tot verminderde expressie van het gen Ebf1. Uit het onderzoek bleek de reden voor deze vermindering migratie van genen naar het B compartiment te zijn, hier liggen stukken chromatine die niet op dat moment tot transcriptie hoeven te komen. Het onderzoek is uitgevoerd op voorloper- B-cellen van muizen. Deze cellen hebben het gen Ebf1 nodig om te differentiëren van hematopoetische stamcellen naar uitgerijpte B cellen. Als dit gen minder tot expressie komt kunnen deze B cellen meer eigenschappen van de voorloper cel hebben.

### **Workflow (Genomics):**

Ondanks dat DNA data onderdeel van de dataset was, staat er in de methode/artikel niet wat ze hiermee gedaan hebben. Daarom hebben we besloten een variant analyse hierop uit te voeren. Het referentiegenoom (muis) zal vergeleken worden met DNA seq data. In het onderzoek was een muis genoom versie uit 2012 gebruikt (mm10), deze gaan we vervangen door de nieuwste versie (GRCm39). De eerste stap is het downloaden van de .SRA archieven die de fastq bestanden bevatten, hier wordt prefetch voor gebruikt. Daarna worden deze .SRA’s uitgepakt met behulp van fasterq-dump. De nieuw verkregen fastq bestanden kunnen nu op kwaliteit gecontroleerd worden met fastqc, hierna worden ze getrimmed met trimmomatic en wordt de kwaliteit nogmaals bekeken. Duplicaten kunnen verwijderd worden.

### **Workflow (Transcriptomics):**

Net zoals in het originele artikel beschreven was wordt read mapping uitgevoerd met het FASTQ bestand van elk van de samples. Met het .SAM bestand dat verkregen is werd daarna met FeatureCounts en RSEM de gen expressie gekwantificeerd. Deze read mapping was met STAR uitgevoerd maar wij gaan hier HISAT2 voor gebruiken omdat STAR verouderd is volgens het github repo. FeatureCounts gebruikt namelijk het .SAM (of .BAM) bestand en telt hoeveel reads bij elke “feature” (gen/exon) horen. RSEM kan hier ook voor gebruikt worden maar is complexer om te gebruiken dan FeatureCounts. Dit kan daarna gevisualiseerd worden met R in een box-plot, viool-plot, heatmap, MA-plot etc. Ook was er in het artikel gebruik gemaakt van “Cufflinks”, deze gaan we vervangen door “StringTie”. Volgens de website van Cufflinks is StringTie accurater en veel efficiënter.

**De volgende R libraries en commandline tools zijn gebruikt voor de analyse:** Samtools: 1.16.1

### Log:

**02-09-2024**:  
Het gekozen artikel: <https://pubmed.ncbi.nlm.nih.gov/38866970> Gene expression omnibus: <https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE211975>

### SRA’s downloaden (Genomics):

De eerste stap om een variant calling analyse uit te voeren is het downloaden van de genetische data die verkregen was door het sequencen van de samples. Met behulp van de SRA run selector van NCBI is een selectie van SRA’s gemaakt die gedownload moeten worden. Een SRA (Sequence Read Archive) is een gecomprimeerd archief dat de sequencing reads bevat. Om deze bestanden te downloaden wordt gebruikt gemaakt van “prefetch”, deze commandline tool is onderdeel van de SRA toolkit.

In de volgende stap worden deze bestanden uitgepakt. Het onderstaande stuk code is op mijn computer uitgevoerd en de data is op een externe HDD opgeslagen. Dit zelfde is gedaan op de assemblix computer waar de andere groepsleden ook bij kunnen. export om de “prefetch” binairy (tijdelijk) toe te voegen aan het PATH.

export PATH="/home/floris/Documenten/Applicaties/sratoolkit.3.1.1-ubuntu64/bin/:$PATH"  
  
cd /run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/DNA/SRA/  
  
prefetch $(</run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/DNA/SRA/SraAccList.csv) \  
--output-directory "/run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/DNA/SRA/" --max-size 200G

**.SRA bestanden naar .FASTQ omzetten:**  
Met behulp van fasterq-dump worden de .SRA bestanden uitgepakt.

fasterq-dump maakt ook onderdeel uit van de SRA toolkit en haalt de data uit het .SRA archief naar het FASTQ format.

Een FASTQ bestand bevat tekst met de sequence data, de reads en ook een bijbehorende kwaliteitsscore. Deze score wordt aangegeven met een ASCII karakter.

Een FASTQ bestand heeft de volgende indeling:

Een header die met “@” begint gevolgd door een sequentie ID en optioneel een beschrijving wat het bestand bevat. Daar onder zitten de sequentie letters.

En daar onder een regel met een “+” karakter. Hier onder staat de kwaliteitsscore, deze gaat van ASCII 33 (“!” teken), laagste kwaliteit tot ASCII 126 (“~” teken).

De kwaliteitsscore wordt ook wel Phred score genoemd. Deze score is logaritmisch gerelateerd aan de waarschijnlijkheid dat de “base call” verkeerd is. Q = -log(E) waarbij Q de phred score is en E de waarschijnlijkheid van verkeerde base call.

Phred-33 is het meest gebruikt maar Phred-64 bestaat ook. Het verschil is dat Phred-33 ge-encodeerd is met met ASCII 33 (!) tot ASCII 126 (~) terwijl Phred-64 van ASCII 64 (@) tot ASCII 126 gaat.

Bron: <https://gatk.broadinstitute.org/hc/en-us/articles/360035531872-Phred-scaled-quality-scores>

# Eerst nieuwe map maken voor fastq  
# Locale test: (niet op de assemblix server)   
cd /run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/DNA/SRA  
  
  
find /run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/DNA/SRA -name "\*.sra" | \  
 parallel fasterq-dump -O /run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/DNA/SRA/FASTQ {}  
  
  
# parallel gebruiken: cat accessionlist.txt | parallel ls -lah SRA/{}/{}.sra

**Onderzoeksvraag Genomics**

De volgende onderzoeksvraag is geformuleerd voor het genomics onderzoek:

Zijn er varianten aanwezig van PAX5 en Ebf1 in het genoom van de rag2(-/-) muizen die gebruikt zijn?

Deze vraag is relevant omdat voor het transcriptomics gedeelte van het onderzoek gekeken wordt naar verschillen in expressie van deze genen. Als het blijkt dat er variaten van deze genen aanwezig zijn kan het verschil in expressie tussen de muizen die gebruikt zijn in het onderzoek niet alleen toegewezen worden aan de factoren waar naar getest wordt, de invloed van veroudering op de expressie. Dan zou het ook kunnen komen door mutaties van deze genen.

Er is DNA sequentie data beschikbaar van rag2(-/-) muizen. Daarom kan er niet gekeken worden naar mutaties in het rag2 gen omdat deze niet meer aanwezig is.

Rag2 knockout muizen zijn niet in staat om uitgerijpte T en B lymfocyten te maken. Bron: <https://pubmed.ncbi.nlm.nih.gov/1547487/>

Het gen Ebf1 (Early B-cell factor 1) is een gen dat bijdraagt aan de differentiatie van voorloper b cellen. bron: <https://www.nature.com/articles/ni.2641>

De reden dat gekozen is voor dit gen is dat het resultaat van een verminderde expressie zichtbaar gemaakt kan worden door functieverlies van B cellen. Samen met PAX5 werken deze genen aan de uitrijping van hematopoetische stam cellen. Functieverlies van PAX5 leidt tot accumulatie van snel proliferende lymfoblasten die niet meer normale differentiatie kunnen ondergaan. Ook is PAX5 een tumorsuppressor gen maar dat is niet van invloed op dit onderzoek.

bron: <https://www.sciencedirect.com/topics/neuroscience/pax5>

**Test data genereren** **16-09-2024**

Om te voorkomen dat een parallel commando na het uitvoeren van een lange bewerking een fout of onverwacht resultaat geeft moet er eerst een subset van de te gebruiken data gemaakt worden. Hiermee kunnen de volgende commando’s eerst uitgevoerd worden om te verifiëren dat alles goed werkt.

Om test data te genereren op basis van twee van de fastq bestanden is de volgende code gebruikt: Per sample bestand zitten er 1000000 reads in. Dit commando is ook uitgevoerd op de assemblix server maar dan met een ander path naar de data. Voor het maken van deze test data is seqkit (versie 2.3.0) gebruikt.

Seqkit kan gebruikt worden voor meerdere bewerkingen zoals het filteren van sequenties op lengte / kwaliteit, DNA/RNA translateren naar een aminozuur sequentie. bron: [https://bioinf.shenwei.me/seqkit](https://bioinf.shenwei.me/seqkit/)/

cd /run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/DNA/SRA  
  
seqkit head -n 1000000 SRR26980527\_1.fastq > subset\_SRR26980527\_1.fastq  
seqkit head -n 1000000 SRR26980527\_2.fastq > subset\_SRR26980527\_2.fastq

**Gebruikte samples** **17-09-2024**

Na het controle van de kwaliteit van de fastq bestanden bleek het dat de samples met read lengtes van 250 bp getrimd moeten worden. De volgende samples zijn wildtype: (WT C57BL/6J) SRR26980527, SRR26980528, SRR26980529, SRR26980530. En de volgende samples zijn Rag2(-/-) knockout: SRR26980549, SRR26980550, SRR26980551, SRR26980552 Al deze samples zijn paired end sequenced Primary pro-B cells geselecteerd op CD19+.

### Indexeren en alignen:

Met enkel een fastq bestand met reads is nog niet duidelijk waar in het genoom van het organisme deze reads kwamen. Met read mapping worden de reads vergeleken met een bekend genoom, in dit geval het mm39 muis referentiegenoom en worden de reads er tegen aan gelegd zodat hun locatie in het genoom bekend wordt. Aangezien er geannoteerde versies van het referentiegenoom zijn, met de namen van de genen kan dan ook achterhaald worden van welke genen de reads deel uitmaken.

<https://www.ebi.ac.uk/training/online/courses/functional-genomics-ii-common-technologies-and-data-analysis-methods/rna-sequencing/performing-a-rna-seq-experiment/data-analysis/read-mapping-or-alignment/>

bron: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3375638/>

Omdat het programma dat gebruikt gaat worden voor het alignen, BWA-mem2, een geïndexeerde versie van het referentiegenoom nodig heeft moet deze eerst gemaakt worden. Dit kan gedaan worden met bwa-mem2 index. indexeren maakt het align proces veel sneller.

De samples SRR26980528\_1 en \_2 zijn gebruikt na het referentiegenoom te indexeren met BWA\_MEM2. Het gebruikte referentie genoom is mm39 (GCF\_000001635.27).

In de onderstaande code chunk wordt het geïndexeerde muisgenoom “testnaam” genoemd. Daarna worden twee samples (forward read en reverse read) ge-aligned met het zojuist verkregen geïndexeerde referentiegenoom. Deze twee bewerkingen worden beiden met bwa-mem2 uitgevoerd.

cd /run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/DNA/SRA  
  
  
# Het indexeren van het referentiegenoom met bwa-mem2.   
bwa-mem2 index -p testnaam referentiegenoom  
  
# Voorbeeld aligning:  
bwa-mem2 mem ref.fa read1.fq read2.fq > aln-pe.sam  
  
# Align sample SRR26980549\_1.fastq (deel 1 en 2) met het referentiegenoom mm39 muis.   
./bwa\_mem2/bwa-mem2-2.2.1\_x64-linux/bwa-mem2 mem -t 40 testnaam fastq/SRR26980549\_1.fastq fastq/SRR26980549\_2.fastq > aligned\_SRR26980549.sam &  
  
# Met parallel:   
cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel './bwa\_mem2/bwa-mem2-2.2.1\_x64-linux/bwa-mem2 mem -t 40 testnaam fastq/{}\_1.fastq fastq/{}\_2.fastq > BWA\_aligned\_output/aligned\_{}.sam' &

## Trimmen van reads

cat data/GSE149995\_Sra\_RunInfo.csv | \  
 parallel 'TrimmomaticPE -threads 16 ' \  
 '/students/2024-2025/Thema05/3dconformatieChromatine/fastq/{}.fastq.gz ' \  
 '/students/2024-2025/Thema05/3dconformatieChromatine/fastq-trimmed/{}.trimmed.fastq.gz ' \  
 'ILLUMINACLIP:/students/2024-2025/Thema05/3dconformatieChromatineTrimmomatic/adapters/TruSeq3-SE.fa:2:30:10 ' \  
 'MINLEN:40 ' \  
 'SLIDINGWINDOW:4:20'

**24-09-2024**  
Het mappen met behulp van BWA-mem2 is opnieuw uitgevoerd na het trimmen. Het referentiegenoom dat in de vorige stap geïndexeerd was wordt weer opnieuw gebruikt.

cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel '/students/2024-2025/Thema05/3dconformatieChromatine/bwa\_mem2/bwa-mem2-2.2.1\_x64-linux/bwa-mem2 mem -t 50 testnaam /students/2024-2025/Thema05/3dconformatieChromatine/Trimmomatic\_output/paired/{}\_forward\_paired.fastq /students/2024-2025/Thema05/3dconformatieChromatine/Trimmomatic\_output/paired/{}\_rev\_paired.fastq > /students/2024-2025/Thema05/3dconformatieChromatine/mapping/{}.sam' &

**14-09-2024**  
In tegenstelling tot het artikel gaan we gebruik maken van Muis (*Mus musculus*) referentiegenoom: GRCm39 (GCF\_000001635.27) gebruik maken. Deze is nieuwer dan versie mm10 die gebruikt was. Aangezien het onderzoek dit jaar gepubliceerd is en voltooid was in 2022 vraag ik mij af waarom ze voor een versie uit 2012 gekozen hebben terwijl er een nieuwere beschikbaar was.

**.sam bestanden sorteren:** **17-09-2024**

Na de vorige read mapping stap zijn er .SAM (Sequence alignment map) bestanden verkregen, dit is een tekst gebaseerde manier om sequenties die aligned zijn tegen een referentie sequentie op te slaan. Een .sam bestand bestaat uit een header en een alignment deel.

Omdat de .SAM bestanden niet georderd zijn op positie in het genoom moeten ze eerst met samtools sort omgezet worden naar een .BAM bestand, dit is nu wel gesorteerd door samtools. Daarnaast kan het opvragen van data sneller gemaakt worden als het geïndexeerd is.

Na de index stap uit gevoerd te hebben is er ook een .bai bestand, dit is een index voor het .bam bestand waardoor andere tools sneller met het .bam bestand kunnen werken.

bron: <https://gatk.broadinstitute.org/hc/en-us/articles/360035890791-SAM-or-BAM-or-CRAM-Mapped-sequence-data-formats>

Om al deze bewerkingen uit te voeren wordt gebruik gemaakt van samtools (versie 1.16.1). Dit is een collectie tools om met high-throughput sequence data te werken. Zo kan het bijvoorbeeld fastq omzetten naar .bam / .cram bestanden, WGS/WES mapping naar variant calls en het filteren van .vcf bestanden.

Met samtools sort -n worden de .sam bestanden gesorteerd op read naam (De QNAME kolom in het bestand). Ook wordt het bestand geconverteerd van .SAM naar .BAM.

Het format is aangegeven met “-O BAM” en multithreading wordt aangegeven met “-(**16?**)” (16 threads).

Bron: <http://www.htslib.org/doc/samtools-sort.html>

# Voor een enkel sample:  
samtools sort -@40 -n -O BAM -o aligned\_sorted\_SRR26980549.bam aligned\_SRR26980549.sam  
  
# Met parallel:  
cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'samtools sort -@40 -n -O BAM -o samtools\_output/aligned\_sorted\_{}.bam BWA\_aligned\_output/aligned\_{}.sam ' &

“fixmate -m” voegt mate score tags toe die gebruikt worden door markdup om de beste reads te selecteren om te houden.

Met “—Threads” wordt het aantal threads gespecificeerd dat de computer moet gebruiken voor dit commando.

Bron: <http://www.htslib.org/doc/samtools-markdup.html>

# Voor een enkel sample:  
samtools fixmate -m --threads 40 aligned\_sorted\_SRR26980549.bam \ fixed\_mates\_aligned\_sorted\_SRR26980549.bam  
  
# Met parallel:  
cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'samtools fixmate -m --threads 40 samtools\_output/aligned\_sorted\_{}.bam samtools\_output/fixed\_mates\_aligned\_sorted\_{}.bam' &

Daarna met “samtools sort” op coördinaten gesorteerd.

# Voor een enkel sample:  
samtools sort -@80 -o sorted\_coordinates\_SRR26980549.bam \ fixed\_mates\_aligned\_sorted\_SRR26980549.bam  
  
# Met parallel:  
cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'samtools sort -@80 -o samtools\_output/sorted\_coordinates\_{}.bam samtools\_output/fixed\_mates\_aligned\_sorted\_{}.bam' &

Met samtools markdup worden duplicaten gemarkeerd. Met het “-r” argument worden deze verwijderd en met “-s” worden statistieken over de data en uitgevoerde handelingen.

Bron: <http://www.htslib.org/doc/samtools-markdup.html>

# Voor een enkel sample:  
samtools markdup -r -s sorted\_coordinates\_SRR26980549.bam dedup\_SRR26980549.bam  
  
# Met parallel:  
cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'samtools markdup -@80 -r -s samtools\_output/sorted\_coordinates\_{}.bam samtools\_output/dedup\_{}.bam' &

Met samtools index wordt er een index bestand gemaakt.

# Voor een enkel sample:  
samtools index -@80 dedup\_SRR26980549.bam dedup\_SRR26980549.bai &  
  
# Met parallel:  
cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'samtools index -@80 samtools\_output/dedup\_{}.bam samtools\_output/dedup\_{}.bai' &

## Variant calling en filteren van varianten:

**Variant calling met Freebayes:**

Nu er gemapte reads zijn kan er variant calling op uitgevoerd worden. Hiermee kunnen mutaties / varianten gedetecteerd en mits ze statistisch significant zijn gerapporteerd worden. Dit zou ook grafisch met IGV (Genome browser) kunnen maar zou veel meer tijd kosten.

Voor variant calling is freebayes uitgekozen. Dit is een genetische variant detector gemaakt om kleine polymorfismen zoals: SNPs, inserties, deleties en MNPs (multi-nucleotide polymorfismes) te herkennen.

Bron: <https://github.com/freebayes/freebayes> Na het uitvoeren van het variant calling met freebayes worden de output bestanden, de .vcf’s gefilterd door ze in vcffilter te pipen. In dit onderstaande stuk code worden de resultaten van de freebayes variant calling direct gefilterd op kwaliteit met vcffilter. Met “QUAL = 20” worden enkel de variants geselecteerd die 99% kans hebben dat er een variant zit op die plaats. De reden dat op kwaliteit gefilterd moet worden is…. (nog bedenken.)

# Voor een bestand:  
/students/2024-2025/Thema05/3dconformatieChromatine/freebayes/freebayes-1.3.6-linux-amd64-static \  
 -f /students/2024-2025/Thema05/3dconformatieChromatine/Mapping\_ref/ncbi\_dataset/ncbi\_dataset/data/GCA\_000001635.9/GCA\_000001635.9\_GRCm39\_genomic.fna dedup\_SRR26980549.bam | /students/2024-2025/Thema05/3dconformatieChromatine/vcffilter/vcffilter-assembly-0.2.jar -f "QUAL > 20" > filtered\_SRR26980549.vcf  
   
   
# Met parallel:   
cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel '/students/2024-2025/Thema05/3dconformatieChromatine/freebayes/freebayes-1.3.6-linux-amd64-static \  
 -f /students/2024-2025/Thema05/3dconformatieChromatine/Mapping\_ref/ncbi\_dataset/ncbi\_dataset/data/GCA\_000001635.9/GCA\_000001635.9\_GRCm39\_genomic.fna samtools\_output/dedup\_{}.bam | /students/2024-2025/Thema05/3dconformatieChromatine/vcffilter/vcffilter-assembly-0.2.jar -f "QUAL > 20" > variant\_calling/filtered\_{}.vcf' &  
   
 # Test zonder filter stap:  
cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel '/students/2024-2025/Thema05/3dconformatieChromatine/freebayes/freebayes-1.3.6-linux-amd64-static \  
 -f /students/2024-2025/Thema05/3dconformatieChromatine/Mapping\_ref/ncbi\_dataset/ncbi\_dataset/data/GCA\_000001635.9/GCA\_000001635.9\_GRCm39\_genomic.fna samtools\_output/dedup\_{}.bam > variant\_calling/variants{}.vcf' &

**Filteren:** Met –minQualScore 30 wordt gefilterd op de waarschijnlijkheid dat de variant echt aanwezig is. Met de formule P = 10^(-Q/10) kan de kans uitgerekend worden dat de variant niet echt aanwezig is. Met een score van 30 is dit: P = 10^(-Q/10) = 0.001. In andere woorden: De kans is 100-0.001 = 99.999% dat het wel klopt.

cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'java -jar /students/2024-2025/Thema05/3dconformatieChromatine/vcffilter/vcffilter-assembly-0.2.jar -I variant\_calling/variants{}.vcf --minQualScore 30 --outputVcf variant\_calling/filterd\_variant\_{}.vcf'

De .vcf’s zijn nu gefilterd, voorbeeldbestandsnaam: filterd\_variant\_SRR26980549.vcf

**18-09-2024**

Om alle paired read samples te mappen is het onderstaande bash script gebruikt. Hier is weer het parallel commando voor gebruikt. De lijst met sample namen wordt aan bwa-mem2 gegeven zodat het programma naar input bestanden met deze namen kan kijken en de gegenereerde .sam bestanden ook deze naam te geven.

cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel '/students/2024-2025/Thema05/3dconformatieChromatine/bwa\_mem2/bwa-mem2-2.2.1\_x64-linux/bwa-mem2 mem -t 60 testnaam /students/2024-2025/Thema05/3dconformatieChromatine/fastq/{}\_1.fastq /students/2024-2025/Thema05/3dconformatieChromatine/fastq/{}\_2.fastq > /students/2024-2025/Thema05/3dconformatieChromatine/mapping/{}.sam' &

**23-09-2024**  
De fastq bestanden zijn met fastqc gecontroleerd en vervolgens getrimmed met Trimmomatic met het onderstaande stuk code. Voor Trimmomatic zijn de volgende instellingen gebruikt: MINLEN:40 en SLIDINGWINDOW:4:20.

**Ook zijn de volgende samples zijn niet gebruikt:**  
B220+CD43+IgM- sorted primary pro-B cells.  
Deze samples zijn verkregen met een Illumina MiSeq.

SRR26980527  
SRR26980528  
SRR26980529  
SRR26980530

De reden hiervoor is dat het samples van WT muizen zijn, hier gaan we het referentiegenoom voor gebruiken. Ook was de kwaliteit van deze sequenties volgens fastqc veel slechter dan die van de onderstaande samples. Mogelijk komt dit omdat de reads te lang waren en de kwaliteit daardoor te snel naar beneden ging.

**De volgende zijn wel gebruikt:**  
Primary pro-B cells by CD19+ selection (Rag2-/-) Deze samples zijn verkregen met een Illumina NovaSeq 6000. SRR26980549.sam  
SRR26980550.sam  
SRR26980551.sam  
SRR26980552.sam

De onnodige samples zijn ook verwijderd uit de lijst (SraAccList.csv) omdat deze met de pipe operator aan de parallel opdracht gegeven wordt. Als de namen in de accession

Feedback: Veel meer uitleg toevoegen, redenen waarom keuzes gemaakt zijn. Versies toevoegen van alle gebruikte software. Overal nog bronnen toevoegen. redenen waarom voor specifieke software gekozen is toevoegen.

Voordat freebayes uitgevoerd kan worden moet het referentiegenoom weer geïndexeerd worden, net als met BWA-mem2 index maar dan met samtools faidx zoals in de onderstaande code chunk beschreven is.

samtools faidx GCA\_000001635.9\_GRCm39\_genomic.fna

Met het geïndexeerde referentiegenoom (GCA\_000001635.9\_GRCm39) was vervolgens de freebayes variant calling uitgevoerd.

/students/2024-2025/Thema05/3dconformatieChromatine/freebayes/freebayes-1.3.6-linux-amd64-static -f GCA\_000001635.9\_GRCm39\_genomic.fna \ /students/2024-2025/Thema05/3dconformatieChromatine/aligned\_sorted\_SRR26980549.bam > \ /students/2024-2025/Thema05/3dconformatieChromatine/variant\_calling/SRR26980549\_test.vcf

**Gen annotatie** Eerst is er een .bed bestand van het niet door mensen leesbare .bam bestand gemaakt, daarna zijn de genen geannoteerd zodat ze geen namen hebben zoals: SRR26980549.46263152/2 maar …. Met behulp van bamToBed wordt het .bam bestand omgezet naar een .bed (Browser extensible data). De eerste kolom van een .bed bevat de naam van het chromosoom, de tweede kolom het start coördinaat van de feature, de derde kolom het eind coördinaat van de feature, 4e kolom de naam, daarna optioneel een score en dan de streng (-/+). bron: <https://genome.ucsc.edu/FAQ/FAQformat.html>

cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'bamToBed -i dedup\_{}.bam > dedeup\_{}.bed'

Voordat bcftools isec uitgevoerd kan worden moeten de vcf bestanden gecomprimeerd worden met bgzip.

cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'bgzip filterd\_variant\_{}.vcf'

Daarna, om de vervolg processen sneller te maken (en omdat isec anders een foutmelding geeft) moet een index bestand gemaakt worden. Hier wordt “tabix” voor gebruikt. Het onderstaande commando gebruikt de net gemaakte bgzip gecomprimeerde bestanden van de .vcf als input.

cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'tabix -p vcf filterd\_variant\_{}.vcf.gz'

## cat: /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv: Bestand of map bestaat niet

## Isec

Er zijn vier biologische replicaten, met behulp van “bcftools isec -c all” worden enkel de varianten die in alle vier de muizen aanwezig zijn geselecteerd. Hierdoor worden veel “willekeurige” mutaties er uit gefilterd die niet in alle vier de samples aanwezig waren. Bron: <https://samtools.github.io/bcftools/bcftools.html> In het onderstaande code chunk wordt bcftools isec uitgevoerd met argumenten: “-c all” dit betekend dat alle varianten die overeen komen tussen de vier samples in het output bestand opgeslagen moeten worden.

bcftools isec -c all filterd\_variant\_SRR26980549.vcf.gz filterd\_variant\_SRR26980550.vcf.gz filterd\_variant\_SRR26980551.vcf.gz filterd\_variant\_SRR26980552.vcf.gz > isec.vcf

## **Annoteren van de .vcf bestanden:**

Het onderstaande script was eerst uitgevoerd met mm39 als referentiegenoom voor de annotatie maar dit gaf een foutmelding (Op de github pagina van het programma snpEff waren er meer meldingen van dit gedrag. <https://github.com/pcingola/SnpEff/issues/536>) dus is het opnieuw uitgevoerd met het oude mm10 referentiegenoom. Dit is het zelfde referentiegenoom dat ook het artikel gebruikt was.

# Test:  
/students/2024-2025/Thema05/3dconformatieChromatine/snpEff/snpEff.jar mm39 filtered\_variants.vcf > annotated\_variants.vcf  
  
# Voor alle .vcf bestanden:  
cat /students/2024-2025/Thema05/3dconformatieChromatine/SRA/SraAccList.csv | parallel 'java -jar /students/2024-2025/Thema05/3dconformatieChromatine/snpEff/snpEff.jar mm39 /students/2024-2025/Thema05/3dconformatieChromatine/variant\_calling/filterd\_variant\_{}.vcf > /students/2024-2025/Thema05/3dconformatieChromatine/annotated\_variants\_{}.vcf'  
  
# Alleen het isec .vcf bestand:  
java -jar /students/2024-2025/Thema05/3dconformatieChromatine/snpEff/snpEff/snpEff.jar mm39 /students/2024-2025/Thema05/3dconformatieChromatine/variant\_calling/isec.vcf > /students/2024-2025/Thema05/3dconformatieChromatine/annotated\_isec.vcf

## **Kwaliteitscontrole vcf bestanden:**

## **Visualisatie van de verkregen data:**

## **Verschillen tussen onze resultaten en die van het originele onderzoek**

## **referenties**

Artikel 3d conformatie chromatine: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10329473/> Samtools: <http://www.htslib.org/> BWA-MEM2 <https://github.com/bwa-mem2/bwa-mem2> Trimmomatic: <http://www.usadellab.org/cms/?page=trimmomatic> FastQC: <https://www.bioinformatics.babraham.ac.uk/projects/fastqc/>

# Transcriptomics:

### SRA’s downloaden (Transcriptomics):

De SRA’s van de RNA seq worden gedownload met het onderstaande stuk code. De gebruikte accention list staat in de github repository.

export PATH="/home/floris/Documenten/Applicaties/sratoolkit.3.1.1-ubuntu64/bin/:$PATH"  
  
  
  
cd /run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/Transcriptomics/  
  
prefetch $(</run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/Transcriptomics/SRR\_Acc\_List.txt) \  
--output-directory "/run/media/floris/FLORIS\_3/DATA\_SETS/3D\_Chromatine\_Conformatie/Transcriptomics/" --max-size 200G