EBAI2017 : TP Variants

Elodie GIRARD

2017-11-14

Contents

```
pwd

source activate python3

cp -r /data/home/mbernard/atelier_variant .

cd atelier_variant
ls -lh *
```

Contrôle qualité avec FASTQC

```
mkdir part1_qual_align
cd part1_qual_align

mkdir FASTQC_rawdata

fastqc -h # affiche l'aide

fastqc --threads 2 --outdir FASTQC_rawdata ../fastq/SRR1262731_extract_R1.fq
fastqc --threads 2 --outdir FASTQC_rawdata ../fastq/SRR1262731_extract_R2.fq

ls -lh *
```

Retrait des adaptateurs avec Cutadapt

```
cutadapt -h # affiche l'aide

cutadapt --trim-n --max-n 0.3 --error-rate 0.1 -q 30,30 --minimum-length 50 --pair-filter
    both --paired-output SRR1262731_extract_R2.trimmed.fq --output
    SRR1262731_extract_R1.trimmed.fq ../fastq/SRR1262731_extract_R1.fq
    ../fastq/SRR1262731_extract_R2.fq > SRR1262731_extract_trimming_stats.txt

ls -lh
cat SRR1262731_extract_trimming_stats.txt
```

Contrôle qualité des données trimmées

```
mkdir FASTQC_trimmed

fastqc --threads 2 --outdir FASTQC_trimmed SRR1262731_extract_R1.trimmed.fq
fastqc --threads 2 --outdir FASTQC_trimmed SRR1262731_extract_R2.trimmed.fq
```

Alignement avec BWA

```
bwa # affiche les différentes options

bwa index ../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa

samtools  # affiche les différentes options
```

```
samtools faidx ../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa

picard # affiche les différentes options
java -jar /data/softwares/anaconda3/envs/python3/libexec/picard-2.9.2/picard.jar

picard CreateSequenceDictionary REFERENCE=../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa
    OUTPUT=../genome/Bos_taurus.UMD3.1.dna.toplevel.6.dict

bwa mem # affiche l'aide

bwa mem -t 4 -R "@RG\tID:id1\tPL:illumina\tPU:HXXX\tLB:Solexa\tSM:sample1"
    ../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa SRR1262731_extract_R1.trimmed.fq
    SRR1262731_extract_R2.trimmed.fq  > SRR1262731_extract.trimmed.sam

less -S SRR1262731_extract.trimmed.sam

samtools view -@ 4 -Sh SRR1262731_extract.trimmed.sam -bo SRR1262731_extract.trimmed.bam
samtools sort -@ 4 SRR1262731_extract.trimmed.bam SRR1262731_extract.trimmed.sort
samtools index SRR1262731_extract.trimmed.sort.bam

samtools flagstat SRR1262731_extract.trimmed.sort.bam >
    SRR1262731_extract.trimmed.sort.flagstat

cat SRR1262731_extract.trimmed.sort.flagstat

rm SRR1262731_extract.trimmed.sam
rm SRR1262731_extract.trimmed.bam
```

Recherche de contaminants

```
fastq_screen -h

fastq_screen --outdir FastQScreenOutputDirR1 SRR1262731_extract_R1.trimmed.fq

# Missing configuration file

less /data/softwares/fastqscreen/0.11.3/FastQScreen.conf

fastq_screen --outdir FastQScreenOutputDirR1 --subset 1000000 --conf
    /data/softwares/fastqscreen/0.11.3/FastQScreen.conf SRR1262731_extract_R1.trimmed.fq
fastq_screen --outdir FastQScreenOutputDirR2 --subset 1000000 --conf
    /data/softwares/fastqscreen/0.11.3/FastQScreen.conf SRR1262731_extract_R2.trimmed.fq
```

InteRlude R N°1

```
# Quelle est la distribution des qualités de mapping (MAPQ) ? Et quel pourcentage de reads
    ont une MAPQ >=30 ?

## Extraction des qualités de mapping stockées dans la colonne n°5
```

```
samtools view SRR1262731_extract.trimmed.sort.bam | awk '{print $5}' >
    SRR1262731_extract.trimmed.sort.mapping_qualities.txt

## Extraction des reads qui ont une qualité de mapping >=30

samtools view -bh -@ 4 -q 30 > SRR1262731_extract.trimmed.sort.bam >
    SRR1262731_extract.trimmed.sort.q30.bam
samtools flagstat SRR1262731_extract.trimmed.sort.q30.bam

## Statistiques de mapping à MAPQ>=0 et MAPQ>=30

cat SRR1262731_extract.trimmed.sort.flagstat

cat SRR1262731_extract.trimmed.sort.q30.bam

R # ouvre R en interactif
```

```
## Utilisation d'une librairie permettant d'obtenir facilement des graphes de qualité :
    http://www.sthda.com/english/rpkgs/ggpubr/

library(ggpubr)

## Chargement des qualités de mapping sous forme de data frame

mapq <- data.frame(read.table("SRR1262731_extract.trimmed.sort.mapping_qualities.txt"))

head(mapq)
```

```
  V1
1  3
2  7
3  0
4  0
5  1
6  0
```

```
## Représentation simple des MAPQ sous forme d'histogramme (commande R de base)

### stockage des informations de l'histogramme dans une variable

#?hist

h <- hist(mapq[,"V1"])
```

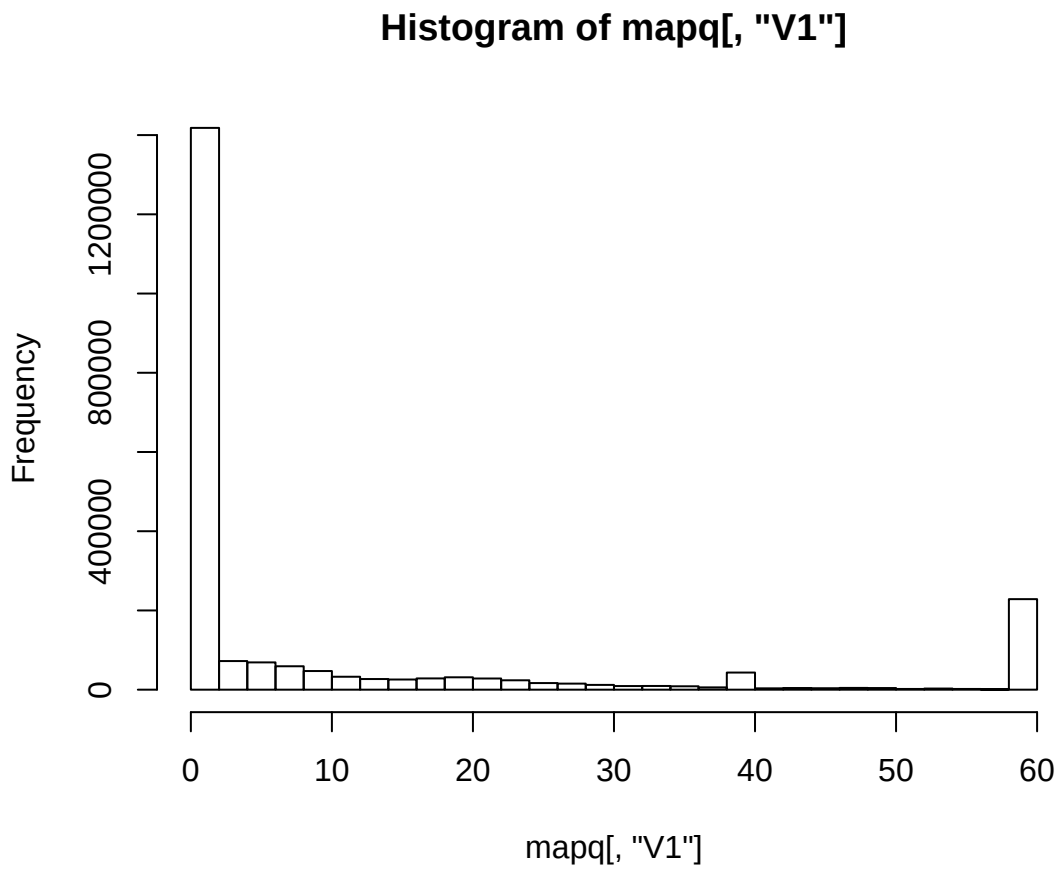

Figure 1: Histogram of mapping qualities

```
h
```

```
$breaks
 [1]  0  2  4  6  8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44
[24] 46 48 50 52 54 56 58 60

$counts
 [1] 1418149   72250   68841   59109   47023   32682   26851   25661
 [9]   28432   31265   28183   23702   16679   15286   12058    9154
[17]    9345    8541    5814   43243    3379    4173    3642    4420
[25]    4091    1812    2796    1822     882  228514

$density
 [1] 0.3168624617 0.0161430942 0.0153814082 0.0132069502 0.0105065290
 [6] 0.0073022644 0.0059994218 0.0057335355 0.0063526706 0.0069856587
[11] 0.0062970356 0.0052958286 0.0037266528 0.0034154095 0.0026941651
[16] 0.0020453133 0.0020879891 0.0019083483 0.0012990443 0.0096619491
[21] 0.0007549829 0.0009323894 0.0008137460 0.0009875775 0.0009140678
[26] 0.0004048621 0.0006247210 0.0004070964 0.0001970686 0.0510577581

$mids
```

```
 [1]  1  3  5  7  9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45
[24] 47 49 51 53 55 57 59

$xname
[1] "mapq[, \"V1\"]"

$equidist
[1] TRUE

attr(,"class")
[1] "histogram"
```

```
### affichage de l'histogramme avec des légendes, un titre et une couleur

plot(h,xlab="Mapping Qualities", main="SRR1262731" , col = "lightgray")
```

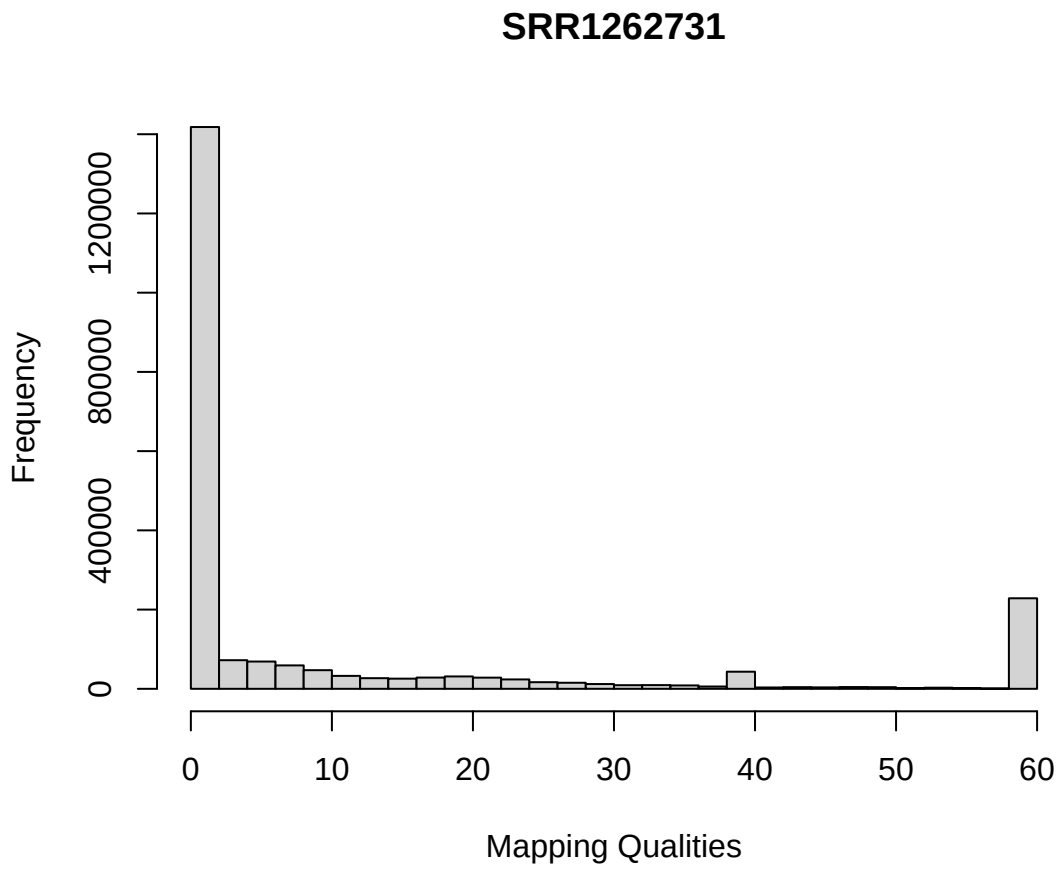

Figure 2: Histogram of mapping qualities with graphical parameters

```
## Représentation des MAPQ sous forme d'histogramme avec ggpubr

#?gghistogram
```

```
p <- gghistogram(mapq, x = "V1", fill = "lightgray", xlab="Mapping Qualities",
    title="SRR1262731")
```

```
Warning: Using `bins = 30` by default. Pick better value with the argument
`bins`.
```

```
print(p)
```

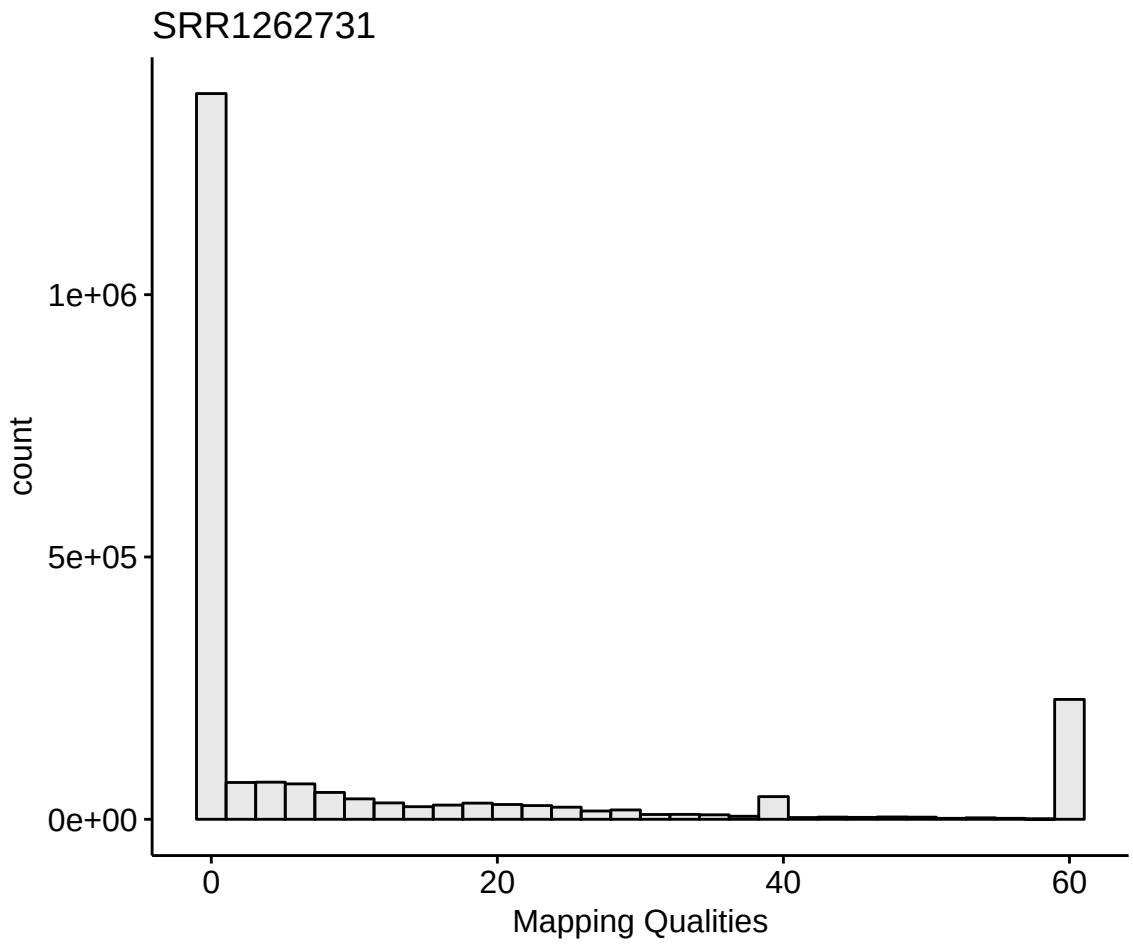

Figure 3: Histogram of mapping qualities with ggpubr

```
## Création d'une table de contingence résumant le nombre de reads présentant une MAPQ >=30

table(mapq[,1]>=30)
```

```
  FALSE    TRUE
1899352  338447
```

```
## Création du data frame contenant les pourcentages de reads non alignés, et alignés avec
    MAPQ>=30 ou MAPQ<30

total <- 2237799
```

```r
mapped <- 1653081

mapped_q30 <- 338447

pc_unmapped <- round((total-mapped)*100/total)
pc_mapq30 <- round(mapped_q30*100/total)
pc_below_mapq30 <- 100-pc_unmapped-pc_mapq30

df <- data.frame(Group = c("Unmapped", "Mapped mapq>=30", "Mapped mapq<30"),Value =
    c(pc_unmapped,pc_mapq30,pc_below_mapq30))

df
```

```
            Group Value
1        Unmapped    26
2 Mapped mapq>=30    15
3  Mapped mapq<30    59
```

```r
# Représentation de ces pourcentages dans un camember ou pie plot avec ggpubr

labs <- paste0(df$Value, "%")

#?ggpie

ggpie(df, "Value", label = labs, lab.pos = "in", lab.font = "white",  fill = "Group", color
    = "white", palette = c("#00AFBB", "#E7B800", "#FC4E07"))
```

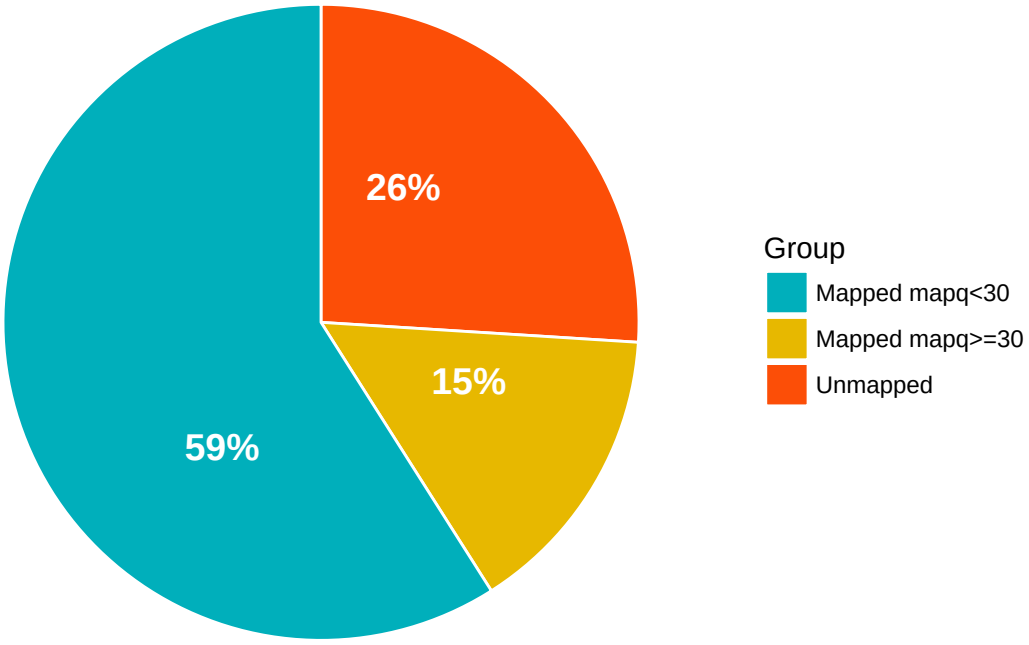

Figure 4: Pie-chart of the alignments summary

```
q("no")
```

Retrait des duplicats de PCR

```
cd ../
mkdir part2_var_calling
cd part2_var_calling

picard MarkDuplicates # affiche l'aide

picard MarkDuplicates INPUT=../part1_qual_align/SRR1262731_extract.trimmed.sort.bam
    OUTPUT=SRR1262731_extract.trimmed.sort.md.bam  METRICS_FILE=metrics_md.txt
    VALIDATION_STRINGENCY=SILENT

samtools flagstat SRR1262731_extract.trimmed.sort.md.bam
```

Intersection avec le fichier de capture

```
bedtools # affiche les différentes options
```

```
bedtools intersect  # affiche l'aide

bedtools intersect -abam SRR1262731_extract.trimmed.sort.md.bam -b ../QTL_BT6.bed >
    SRR1262731_extract.trimmed.sort.md.onTarget.bam

samtools index SRR1262731_extract.trimmed.sort.md.onTarget.bam
```

Calcul de la couverture

```
samtools depth -Q 30 -b ../QTL_BT6.bed  SRR1262731_extract.trimmed.sort.md.onTarget.bam >
    SRR1262731_extract.trimmed.sort.md.onTarget.dp.txt

head SRR1262731_extract.trimmed.sort.md.onTarget.dp.txt
```

Appel des Variants avec HaplotypeCaller

Création de VCF

```
gatk -h  # affiche les différentes options
java -jar /data/softwares/anaconda3/envs/python3/opt/gatk-3.8/GenomeAnalysisTK.jar

gatk -T HaplotypeCaller -h  # affiche l'aide

gatk -T HaplotypeCaller -I SRR1262731_extract.trimmed.sort.md.onTarget.bam -L ../QTL_BT6.bed
    -o SRR1262731.vcf -R ../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa -mmq 30 -ERC "none"

less -S SRR1262731.vcf
```

Création de gVCF

```
gatk -T HaplotypeCaller -I SRR1262731_extract.trimmed.sort.md.onTarget.bam -L ../QTL_BT6.bed
    -o SRR1262731.g.vcf -R ../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa  -mmq 30 -ERC "GVCF"

less -S SRR1262731.g.vcf
```

Combiner les deux échantillons

```
gatk -T CombineGVCFs -L ../QTL_BT6.bed -R ../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa
    --variant SRR1262731.g.vcf --variant ../additional_data/SRR1205992.g.vcf -o
    SRR1205992_SRR1262731.g.vcf

gatk -T GenotypeGVCFs -L ../QTL_BT6.bed -R ../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa
    --variant SRR1205992_SRR1262731.g.vcf -o SRR1205992_SRR1262731.vcf
```

InteRlude R N°2

```
# Quelle est la moyenne de couverture à MAPQ>=30 des régions ciblées ?

R # ouvre R en interactif
```

```
## Chargement du fichier créé avec samtools depth

cov <- read.table("SRR1262731_extract.trimmed.sort.md.onTarget.dp.txt",header=FALSE,sep="\t")

head(cov)
```

```
  V1       V2 V3
1  6 37913111  3
2  6 37913112  3
3  6 37913113  3
4  6 37913114  3
5  6 37913115  3
6  6 37913116  3
```

```
## Changement du nom des colonnes du data frame

colnames(cov) <- c("chr","position","depth")

head(cov)
```

```
  chr position depth
1   6 37913111     3
2   6 37913112     3
3   6 37913113     3
4   6 37913114     3
5   6 37913115     3
6   6 37913116     3
```

```
## Calcul des différentes métriques (minimum, maximum, moyenne médiane...) de la colonne
    "depth"

summary(cov[,"depth"])
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   4.000   6.000   6.707   8.000 116.000
```

```
## Calcul de la moyenne de profondeur et stockage dans une variable

mean_cov <- mean(cov[,"depth"])

mean_cov
```

```
[1] 6.707015
```

```
## Création du fichier pdf qui va contenir le plot

pdf("SRR1262731_extract.trimmed.sort.md.onTarget.dp.pdf",width=10,height=6)

## Représentation sous forme de baton d'histogramme (type=h) de la profondeur par position
    des régions ciblées

plot(cov$position, cov$depth, type="h", col="steelblue", xlab="Position", ylab="Coverage at
    MAPQ>=30", main="SRR1262731 on QLT_BT6.bed")
```

```
## Ajout d'une ligne horizontale de couleur rouge représentant la moyenne de profondeur

abline(h=mean_cov,col="red")

## Ecriture et stockage du plot dans un fichier pdf
invisible(dev.off())
```

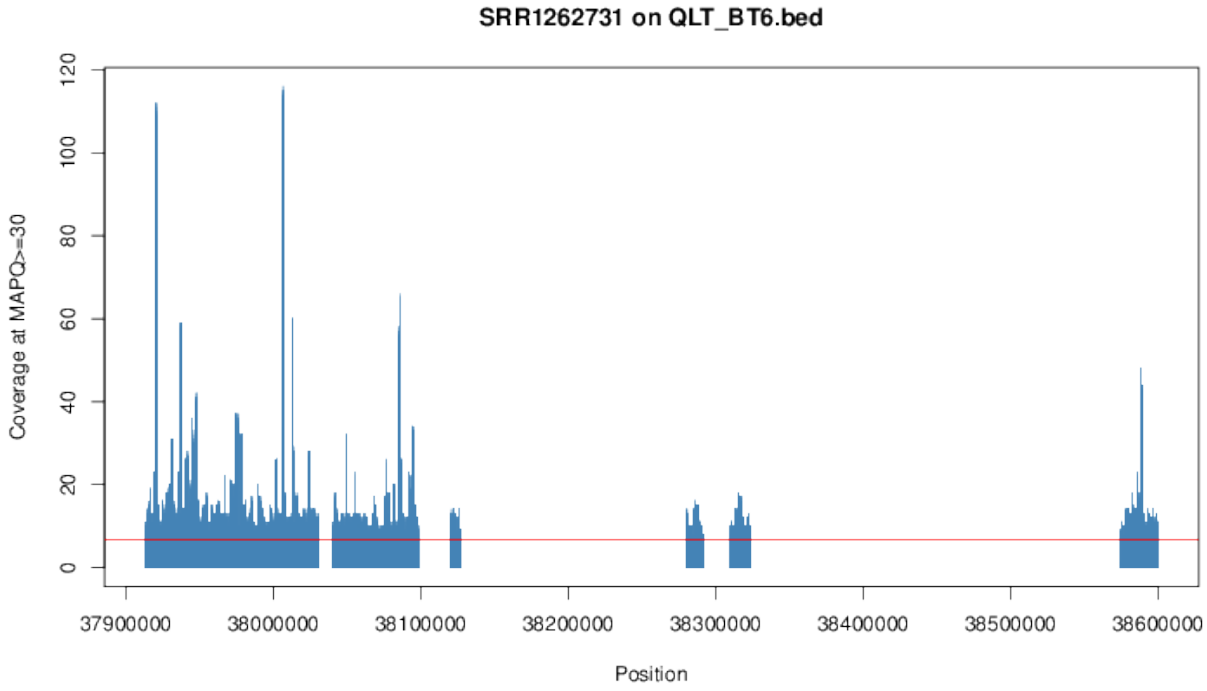

Figure 5: Representation of the per-depth coverage

```
q("no")
```

Filtrage et Annotation des variants

Filtres techniques

```
cd ../
mkdir part3_filt_anno
cd part3_filt_anno
```

```
mkdir Hard_Filtering
cd Hard_Filtering

gatk -T SelectVariants -R ../../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa -V
    ../../part2_var_calling/SRR1205992_SRR1262731.vcf -selectType SNP -o
    SRR1205992_SRR1262731_SNP.vcf
```

```
##Filtrer les SNP selon les paramètres 'dannotations recommandés par GATK:

gatk -T VariantFiltration -R ../../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa -V
    SRR1205992_SRR1262731_SNP.vcf --filterExpression "QD < 2.0 || FS > 60.0 || MQ < 40.0 ||
    MQRankSum < -12.5 || ReadPosRankSum < -8.0" --filterName "hard_filtering_snp" -o
    SRR1205992_SRR1262731_SNP_prefiltered.vcf

##Sélectionner uniquement les variants qui ont passé le filtre:
gatk -T SelectVariants -R ../../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa -V
    SRR1205992_SRR1262731_SNP_prefiltered.vcf -o SRR1205992_SRR1262731_SNP_filtered.vcf
    --excludeFiltered
```

- Selon la profondeur et la qualité

```
cd ..
mkdir Filtre_snpsift
cd Filtre_snpsift


java -jar /data/softwares/snpEff/4.3/SnpSift.jar # affiche l'aide

cat ../Hard_Filtering/SRR1205992_SRR1262731_SNP_filtered.vcf  | java -jar
    /data/softwares/snpEff/4.3/SnpSift.jar filter "( DP >= 20 ) & (QUAL >= 30)" >
    SRR1205992_SRR1262731_DP20_QUAL30.vcf

less -S SRR1205992_SRR1262731_DP20_QUAL30.vcf
```

- En fonction du status hétérozygote des variants

```
cat SRR1205992_SRR1262731_DP20_QUAL30.vcf | java -jar /data/softwares/snpEff/4.3/SnpSift.jar
    filter "isHet(GEN[1]) & isVariant(GEN[1])" > SRR1205992_SRR1262731_DP20_QUAL30_HET.vcf

wc -l *.vcf
```

Comparaison de variants

```
cd ..
mkdir Comparaison_variants
cd Comparaison_variants

grep -vc "#" ../../additional_data/SRR1262731_mpileup.vcf
grep -vc "#" ../../part2_var_calling/SRR1262731.vcf

grep -v "#" ../../additional_data/SRR1262731_mpileup.vcf | cut -f1,2,4,5 >
    SRR1262731_mpileup_cut.tab
grep -v "#" ../../part2_var_calling/SRR1262731.vcf | cut -f1,2,4,5 > SRR1262731_gatk_cut.tab

# Télécharger avec Cyberduck le fichier SRR1262731_mpileup_cut.tab SRR1262731_gatk_cut.tab

bgzip -h
tabix -h

bgzip -c  ../../additional_data/SRR1262731_mpileup.vcf >
    ../../additional_data/SRR1262731_mpileup.vcf.gz
```

```
tabix -p vcf ../../additional_data/SRR1262731_mpileup.vcf.gz

bgzip -c ../../part2_var_calling/SRR1262731.vcf > ../../part2_var_calling/SRR1262731.vcf.gz
tabix -p vcf ../../part2_var_calling/SRR1262731.vcf.gz

vcf-isec -h

vcf-isec  ../../additional_data/SRR1262731_mpileup.vcf.gz
    ../../part2_var_calling/SRR1262731.vcf.gz > common_variants.vcf
```

Annotation des variants

- Création de la base de données snpEff

```
cd ../
mkdir Annotation_variants
cd Annotation_variants

java -jar /data/softwares/snpEff/4.3/snpEff.jar  # affiche l'aide

cp /data/softwares/snpEff/4.3/snpEff.config mon_fichier_snpeff.config

echo "UMD3.1.genome" >> mon_fichier_snpeff.config

mkdir -p UMD3.1
cd UMD3.1

ln -s ../../../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa UMD3.1/sequences.fa
ln -s ../../../genome/Bos_taurus.UMD3.1.89_6.gtf UMD3.1/genes.gtf

cd ..

echo -e "UMD3.1\nSnpEff4.1" > UMD3.1.db

snpEff build -c mon_fichier_snpeff.config -gtf22 -v UMD3.1 -dataDir .

ls -l UMD3.1
```

- Annotation

```
java -jar /data/softwares/snpEff/4.3/snpEff.jar  eff # affiche l'aide

snpEff eff -c mon_fichier_snpeff.config -dataDir . UMD3.1 -s fichier_snpeff_resulat.html
    ../Filtre_snpsift/SRR1205992_SRR1262731_DP20_QUAL30_HET.vcf -csvStats
    fichier_snpeff_resulat.csv  > SRR1205992_SRR1262731.snpEff.vcf

less -S SRR1205992_SRR1262731.snpEff.vcf
```

Filtres fonctionnels

- Retrait des variants synonymes et intergéniques

```
cat SRR1205992_SRR1262731.snpEff.vcf | java -jar /data/softwares/snpEff/4.3/SnpSift.jar
    filter  "(( EFF[*].EFFECT !='synonymous_variant') | ( EFF[*].EFFECT !=
    'intergenic_variant'))" > SRR1205992_SRR1262731.snpEff.NO_SYN.NO_INT.vcf
```

- Sélection des variants situés dans les régions codantes

```
cat SRR1205992_SRR1262731.snpEff.NO_SYN.NO_INT.vcf | java -jar
    /data/softwares/snpEff/4.3/SnpSift.jar filter "EFF[*].BIOTYPE = 'protein_coding' "  >
    SRR1205992_SRR1262731.snpEff.NO_SYN.NO_INT.coding.vcf
```

- Sélection des variants dont l'effet est faux sens

```
cat SRR1205992_SRR1262731.snpEff.NO_SYN.NO_INT.coding.vcf | java -jar
    /data/softwares/snpEff/4.3/SnpSift.jar filter "( EFF[*].EFFECT ='missense_variant')"  >
    SRR1205992_SRR1262731.snpEff.NO_SYN.NO_INT.coding.missense.vcf
```

InteRlude R N°3

```
#source activate python3

# Représentation des variants détectés sur un chromosome

## Chargement de la librairie vcfR qui aide à ouvrir et gérer les fichiers au format vcf
    https://cran.r-project.org/web/packages/vcfR/vignettes/intro_to_vcfR.html

library(ape)
```

```
Warning: package 'ape' was built under R version 3.4.2
```

```
library(vcfR)
library(ggpubr)

## Chargement du fichier vcf avec la commande spéciale read.vcfR

vcf <- read.vcfR("../../part2_var_calling/SRR1262731.vcf",verbose=FALSE)

## Chargement du fichier fasta avec la commande spéciale read.dna du package ape

dna <- read.dna("../../genome/Bos_taurus.UMD3.1.dna.toplevel.6.fa", format = "fasta")

## Chargement du fichier gff contenant les informations sur les gênes du chr6

gff <- read.table("../../RefSeq_UMD3.1.1_protein_coding.chr6.gff3",quote="",sep="\t")

## Chargement du fichier bed contenant les régions cibles

bed <-read.table("../../QTL_BT6.bed")

## Extraction des positions start et end de l'ensemble des régions ciblées

start_region <- bed[1,2]
end_region <- bed[nrow(bed),3]

## Création de l'object chromR contenant les informations du vcf, du fasta et du gff,
    restreint au chromosome 6

chrom <- create.chromR(name='chr6', vcf=vcf, seq=dna, ann=gff)
```

```
head(chrom)
```

```
*****   Class chromR, method head   *****
Name:  chr6
Length:  119,458,736

*****     Sample names (chromR)     *****
[1] "SRR1262731"

*****    VCF fixed data (chromR)    *****
     CHROM POS        ID REF  ALT QUAL     FILTER
[1,] "6"   "37913396" NA "T"  "A" "49.77"  NA
[2,] "6"   "37914164" NA "A"  "G" "120.03" NA
[3,] "6"   "37916445" NA "GT" "G" "28.74"  NA
[4,] "6"   "37917208" NA "G"  "A" "35.77"  NA
[5,] "6"   "37917333" NA "T"  "C" "269.77" NA
[6,] "6"   "37917821" NA "G"  "A" "148.77" NA

INFO column has been suppressed, first INFO record:
 [1] "AC=1"                  "AF=0.500"
 [3] "AN=2"                  "BaseQRankSum=0.967"
 [5] "ClippingRankSum=0.000" "DP=4"
 [7] "ExcessHet=3.0103"      "FS=0.000"
 [9] "MLEAC=1"               "MLEAF=0.500"
[11] "MQ=60.00"              "MQRankSum=0.000"
[13] "QD=12.44"              "ReadPosRankSum=0.674"
[15] "SOR=0.693"

*****  VCF genotype data (chromR)  *****
     FORMAT           SRR1262731
[1,] "GT:AD:DP:GQ:PL" "0/1:2,2:4:71:78,0,71"
[2,] "GT:AD:DP:GQ:PL" "1/1:0,4:4:12:148,12,0"
[3,] "GT:AD:DP:GQ:PL" "0/1:1,2:3:28:66,0,28"
[4,] "GT:AD:DP:GQ:PL" "0/1:5,2:7:64:64,0,194"
[5,] "GT:AD:DP:GQ:PL" "0/1:3,8:11:99:298,0,100"
[6,] "GT:AD:DP:GQ:PL" "0/1:5,5:10:99:177,0,176"

*****      Var info (chromR)       *****
  CHROM      POS    MQ DP mask
1     6 37913396 60.00  4 TRUE
2     6 37914164 60.00  4 TRUE
3     6 37916445 60.00  3 TRUE
4     6 37917208 55.67  7 TRUE
5     6 37917333 60.00 11 TRUE
6     6 37917821 60.00 10 TRUE

*****      VCF mask (chromR)        *****
Percent unmasked: 100

*****      End head (chromR)        *****
```

```
## Extraction de la profondeur (dp) et du nombre de reads supportant chaque variant (ad)
```

```
tab1 <- data.frame(gt=extract.gt(chrom, element="GT", return.alleles=TRUE)[,1])
tab1$ad <- extract.gt(chrom, element="AD", as.numeric=TRUE)[,1]
tab1$dp <- extract.gt(chrom, element="DP", as.numeric=TRUE)[,1]

## Calcul du ratio allélique de chaque variant

tab1$all_ratio <- round(tab1$ad*100/tab1$dp,2)

## Détermination du type de variant
tab1$type <- "SNV"
tab1$len <-nchar(as.character(tab1$gt))-3
tab1[which(tab1$len>0),"type"] <- "INDEL"

head(tab1)
```

```
             gt ad dp all_ratio  type len
6_37913396  T/A  2  4     50.00   SNV   0
6_37914164  G/G  0  4      0.00   SNV   0
6_37916445 GT/G  1  3     33.33 INDEL   1
6_37917208  G/A  5  7     71.43   SNV   0
6_37917333  T/C  3 11     27.27   SNV   0
6_37917821  G/A  5 10     50.00   SNV   0
```

```
table(tab1$type)
```

```
INDEL   SNV
   56   694
```

```
# Pour le deuxième échantillon ...

vcf2 <- read.vcfR("../../additional_data/SRR1205992.g.vcf",verbose=FALSE)
chrom2 <- create.chromR(name='chr6', vcf=vcf2, seq=dna, ann=gff)
tab2 <- data.frame(gt=extract.gt(chrom2, element="GT", return.alleles=TRUE)[,1])
tab2$ad <- extract.gt(chrom2, element="AD", as.numeric=TRUE)[,1]
tab2$dp <- extract.gt(chrom2, element="DP", as.numeric=TRUE)[,1]

## Retrait des positions n'ayant pas de variants (cas du gVCF)

nrow(tab2)
```

```
[1] 81102
```

```
table(is.na(tab2[,"ad"]))
```

```
FALSE  TRUE
 4606 76496
```

```
head(is.na(tab2[,"ad"]))
```

```
[1] TRUE TRUE TRUE TRUE TRUE TRUE
```

```
head(!is.na(tab2[,"ad"]))
```

```
[1] FALSE FALSE FALSE FALSE FALSE FALSE
```

```
head(which(!is.na(tab2[,"ad"])))
```

```
[1]  123  417  490  782  791 1094
```

```
tab2 <- tab2[which(!is.na(tab2[,"ad"])),]

nrow(tab2)
```

```
[1] 4606
```

```
tab2$all_ratio <- round(tab2$ad*100/tab2$dp,2)

tab2$type <- "SNV"
tab2$len <- nchar(as.character(tab2$gt))-3
tab2[which(tab2$len>0),"type"] <- "INDEL"

head(tab2)
```

```
                                                                gt ad dp
6_37913396                                                     A/A  0 13
6_37914164                                                     A/G 11 20
6_37914431                                          T/TATTAAGGTGAAA 12 13
6_37915127                                                     G/A 14 18
6_37915171 A/AACATGTTTAGGAACCAGTTCAAATTCATTTTATTATGAGAATACCTCT 18 20
6_37915853                                                     T/T 15 15
           all_ratio  type len
6_37913396      0.00   SNV   0
6_37914164     55.00   SNV   0
6_37914431     92.31 INDEL  12
6_37915127     77.78   SNV   0
6_37915171     90.00 INDEL  48
6_37915853    100.00   SNV   0
```

```
table(tab2$type)
```

```
INDEL   SNV
  665  3941
```

```
## Représentation de la distribution des ratio alléliques pour les deux échantillons sous
    forme de boxplot avec ggpubr

ar <- data.frame(ratio=c(tab1$all_ratio,tab2$all_ratio) ,
    vcf=rep(c("SRR1262731","SRR1205992") , c(nrow(tab1),nrow(tab2))))

head(ar)
```

```
  ratio        vcf
1 50.00 SRR1262731
2  0.00 SRR1262731
3 33.33 SRR1262731
4 71.43 SRR1262731
5 27.27 SRR1262731
6 50.00 SRR1262731
```

```
g <- ggboxplot(ar, "vcf", "ratio", color="vcf", palette=c("#00AFBB","#E7B800"))
g
```

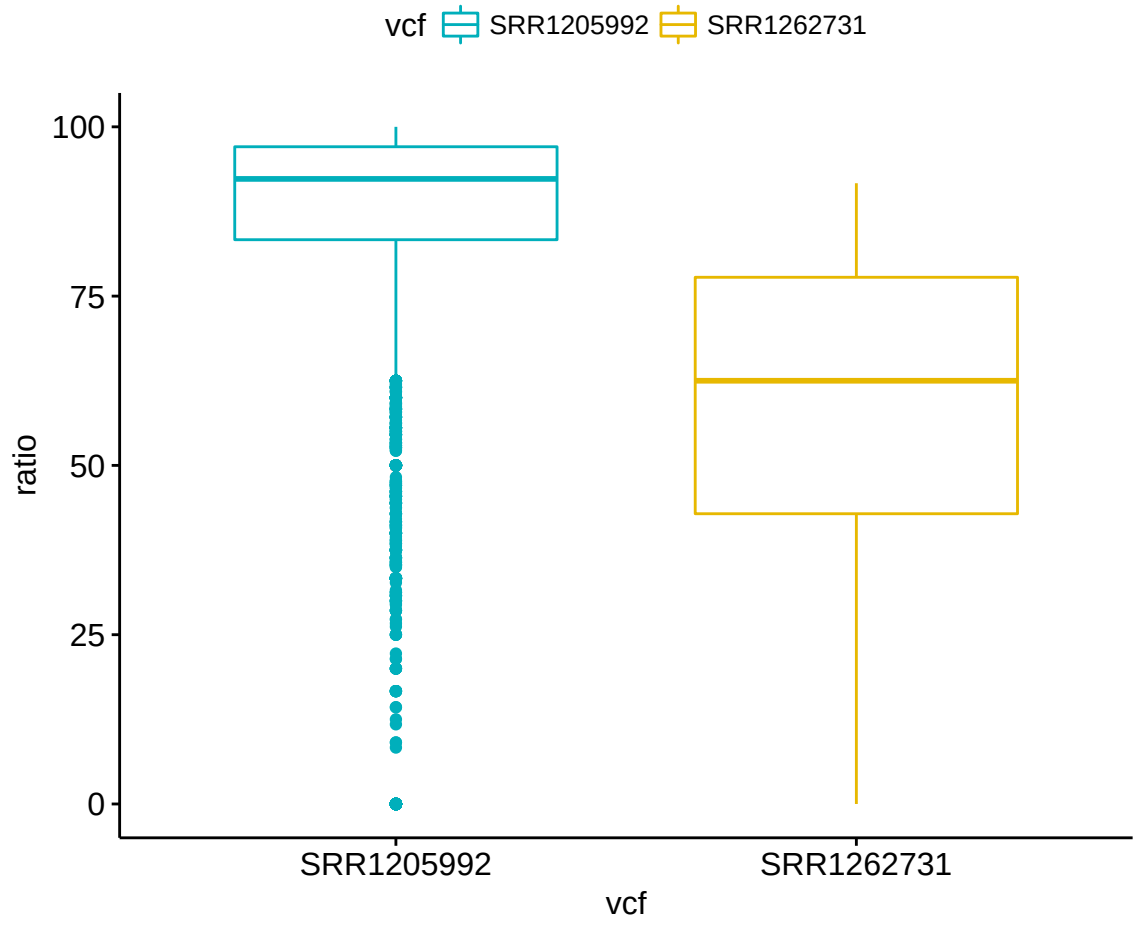

Figure 6: Allelic-ratio distribution as boxplot

```
## Calculation du nombre de variants par fenêtre

chrom <- proc.chromR(chrom, win.size=1000, verbose=TRUE)

## Représentation graphique entre les régions start et end

chromoqc(chrom, xlim=c(start_region, end_region))
```

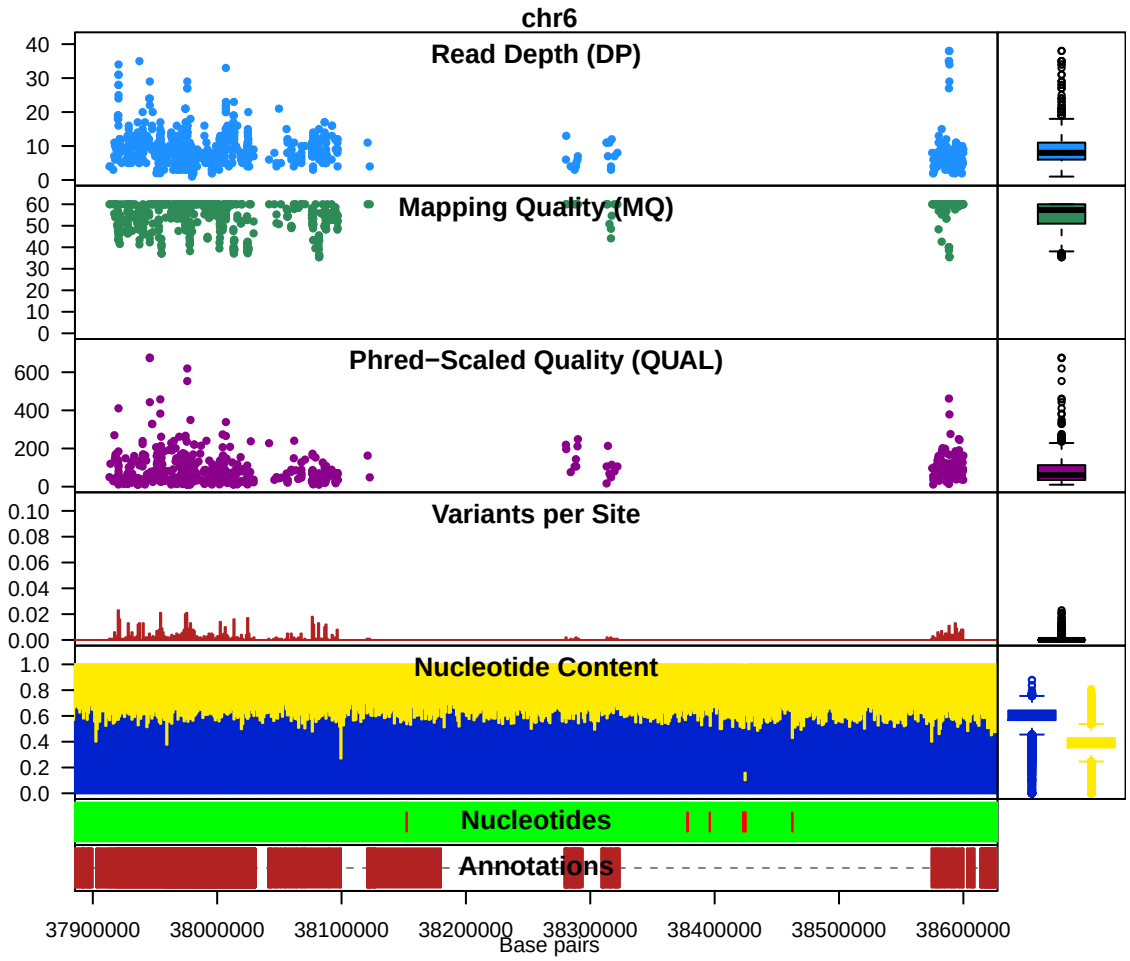

Figure 7: Variants information using vcfR

```
q("no")
```

Recherche de variant structuraux

Avec Pindel

- Détection de la taille d'insert moyenne

```
cd ../../
mkdir part4_sv
cd part4_sv

picard CollectInsertSizeMetrics # affiche l'aide

picard CollectInsertSizeMetrics I=../sv/Zt_chr_10.bam O=insert_size_metrics.txt
    H=insert_size_histogram.pd M=0.5
head -8 insert_size_metrics.txt
```

- Ecriture du fichier de configuration

```
readlink -m ../sv/Zt_chr_10.bam

echo -e $(readlink -m ../sv/Zt_chr_10.bam)'\t'250'\t'Zt_chr_10 > config.txt
cat config.txt
```

- Lancement de Pindel

```
pindel # affiche l'aide

pindel --number_of_threads 10 --report_inversions FALSE --report_duplications FALSE
    --report_long_insertions FALSE --report_breakpoints FALSE --fasta
    ../sv/Zymoseptoria_tritici.MG2.31.dna.toplevel.fa --config-file config.txt
    --output-prefix output_pindel

wc -l output_pindel_D
less output_pindel_D

cat output_pindel_D | grep ChrID > output_pindel_D_no_reads
tr ' ' \\t < output_pindel_D_no_reads > temp
mv temp output_pindel_D_no_reads
```

- Filtrage des résultats

Selon la taille de l'évenement

```
awk '{ if ($3 >= 100) print $0 }' output_pindel_D_no_reads > output_pindel_D_no_reads_ml100
wc -l output_pindel_D_no_reads_ml100
```

Selon le nombre de reads qui supportent l'évenement

```
awk '{ if ($16 >= 3) print $0 }' output_pindel_D_no_reads_ml100 >
    output_pindel_D_no_reads_ml100_rs3
wc -l output_pindel_D_no_reads_ml100_rs3
```

Selon les positions

```
cat output_pindel_D_no_reads_ml100_rs3 | cut -f 8,10,11
```

Avec Delly

- Lancement de Delly

```
#source activate python3

delly

delly --type DEL --indelsize 100 --noindels --genome
    ../sv/Zymoseptoria_tritici.MG2.31.dna.toplevel.fa ../sv/Zt_chr_10.bam -o
    output_DEL_delly.vcf

source deactivate

less output_DEL_delly.vcf

cat output_DEL_delly.vcf | grep -v '#' | wc -l
```

- Filtrage des résultats Récupération des variants flaggés PASS : supportés par 3 reads pairés avec dont la moyenne de qualité de mapping est > 20

```
cat output_DEL_delly.vcf | grep -v LowQual > output_DEL_delly_PASS.vcf
cat output_DEL_delly_PASS.vcf | grep -v '#' | wc -l
```

- Extraction des positions des délétions

```
cat output_DEL_delly_PASS.vcf | grep -v '#' | cut -f 2
cat output_DEL_delly_PASS.vcf | grep -v '#' | cut -d ";" -f 7 | cut -d "=" -f 2
```

InteRlude R N°4

```
# Il y a t-il une différence de taille significative entre les délétions appelées par Delly
    et par Pindel ?

library(vcfR)

## Chargement du vcf de delly

delly=read.vcfR("output_DEL_delly_PASS.vcf")
```

```
Meta line 34 read in.
All meta lines processed.
Character matrix gt created.
Character matrix gt rows: 12
Character matrix gt cols: 10
skip: 0
nrows: 12
row_num: 0


Processed variant: 12
All variants processed
```

```
## Récupération des positions start et end des délétions

start=as.numeric(as.character(getFIX(delly)[,"POS"]))
end=as.numeric(as.character(extract.info(delly,element="END")))

## Calcul de la taille
delly_size=end-start

## Chargeement du fichier de pindel

pindel=read.table("output_pindel_D_no_reads_ml100_rs3")

head(pindel)
```

```
   V1 V2   V3 V4 V5 V6    V7     V8 V9     V10     V11      V12     V13
1  13  D  521 NT  0    ChrID chr_10 BP   32795   33317 BP_range   32795
2  18  D  472 NT  0    ChrID chr_10 BP   57127   57600 BP_range   57127
```

```
3 150  D  221 NT  0    ChrID chr_10 BP  465858  466080 BP_range  465858
4 188  D  330 NT  0    ChrID chr_10 BP  612624  612955 BP_range  612624
5 347  D 9390 NT  0    ChrID chr_10 BP 1175379 1184770 BP_range 1175379
6 366  D 4032 NT  0    ChrID chr_10 BP 1241507 1245540 BP_range 1241507
      V14      V15 V16 V17 V18 V19 V20 V21 V22 V23 V24 V25    V26 V27 V28
1   33319 Supports   6   6   +   3   3   -   3   3  S1  16 SUM_MS 360   1
2   57600 Supports  10  10   +   6   6   -   4   4  S1  35 SUM_MS 600   1
3  466086 Supports  14  14   +  10  10   -   4   4  S1  55 SUM_MS 799   1
4  612955 Supports   8   8   +   5   5   -   3   3  S1  24 SUM_MS 480   1
5 1184779 Supports   8   8   +   4   4   -   4   4  S1  25 SUM_MS 474   1
6 1245548 Supports  11  11   +   9   9   -   2   2  S1  30 SUM_MS 660   1
            V29 V30 V31       V32 V33 V34 V35 V36 V37 V38
1 NumSupSamples   1   1 Zt_chr_10   0   1   3   3   3   3
2 NumSupSamples   1   1 Zt_chr_10   0   0   6   6   4   4
3 NumSupSamples   1   1 Zt_chr_10   2   0  10  10   4   4
4 NumSupSamples   1   1 Zt_chr_10   0   0   5   5   3   3
5 NumSupSamples   1   1 Zt_chr_10   2   0   4   4   4   4
6 NumSupSamples   1   1 Zt_chr_10   1   0   9   9   2   2
```

```
## Calcul de la taille des délétions

pindel_size=pindel[,11]-pindel[,10]

## test de Student dont l'hypothèse nulle est que les moyennes de tailles des délétions sont
    égales

t.test(pindel_size,delly_size)
```

```
    Welch Two Sample t-test

data:  pindel_size and delly_size
t = -1.5815, df = 20.107, p-value = 0.1294
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -12670.416   1740.583
sample estimates:
mean of x mean of y
  5303.75  10768.67
```

```
sessionInfo()
```

```
R version 3.4.1 (2017-06-30)
Platform: x86_64-apple-darwin15.6.0 (64-bit)
Running under: OS X El Capitan 10.11.6

Matrix products: default
BLAS: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib
LAPACK: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib

locale:
[1] fr_FR.UTF-8/fr_FR.UTF-8/fr_FR.UTF-8/C/fr_FR.UTF-8/fr_FR.UTF-8

attached base packages:
```

```
[1] stats     graphics  grDevices utils     datasets  base

other attached packages:
[1] vcfR_1.5.0    ggpubr_0.1.5  magrittr_1.5  ggplot2_2.2.1

loaded via a namespace (and not attached):
 [1] pinfsc50_1.1.0    Rcpp_0.12.13      highr_0.6
 [4] compiler_3.4.1    plyr_1.8.4        bindr_0.1
 [7] methods_3.4.1     tools_3.4.1       digest_0.6.12
[10] viridisLite_0.2.0 evaluate_0.10.1   tibble_1.3.4
[13] gtable_0.2.0      nlme_3.1-131      lattice_0.20-35
[16] mgcv_1.8-22       pkgconfig_2.0.1   rlang_0.1.2
[19] Matrix_1.2-11     yaml_2.1.14       parallel_3.4.1
[22] bindrcpp_0.2      dplyr_0.7.4       stringr_1.2.0
[25] knitr_1.17        cluster_2.0.6     rprojroot_1.2
[28] grid_3.4.1        glue_1.1.1        R6_2.2.2
[31] rmarkdown_1.6     purrr_0.2.3       codetools_0.2-15
[34] backports_1.1.1   scales_0.5.0      htmltools_0.3.6
[37] MASS_7.3-47       assertthat_0.2.0  memuse_3.0-1
[40] permute_0.9-4     colorspace_1.3-2  ape_5.0
[43] labeling_0.3      stringi_1.1.5     lazyeval_0.2.0
[46] munsell_0.4.3     vegan_2.4-4
```

```
q("no")
```