

# Stage\_M2\_NB\_3\_FST

Nicolas Bettembourg

2024-04-22

## Contents

Chargement des packages R . . . . .	1
<b>Analyse de l'indice de différenciation génétique (FST) des SNPS</b>	<b>2</b>
SeqApiPop - 629 échantillons - SNPsBeeMuSe filtered . . . . .	2
No filter - No LD pruning - 10030 SNPs . . . . .	2
MAF > 0.01 - LD pruning = 0.3 (fenêtre de 1749 SNPS et pas de 175 bp) - 3848 SNPs . . .	6
MAF > 0.01 - LD pruning = 0.1 (fenêtre de 50 SNPS et pas de 10 bp) - 1055 SNPs . . . .	10
. . . . .	15

## Chargement des packages R

```
library(ggplot2)
library(qqman)
```

```
##
```

```
## For example usage please run: vignette('qqman')
```

```
##
```

```
## Citation appreciated but not required:
```

```
## Turner, (2018). qqman: an R package for visualizing GWAS results using Q-Q and manhattan plots. Jour
```

```
##
```

```
library(vioplot)
```

```
## Loading required package: sm
```

```
## Package 'sm', version 2.2-6.0: type help(sm) for summary information
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##      as.Date, as.Date.numeric
```

## Analyse de l'indice de différenciation génétique (FST) des SNPS

SeqApiPop - 629 échantillons - SNPsBeeMuSe filtered

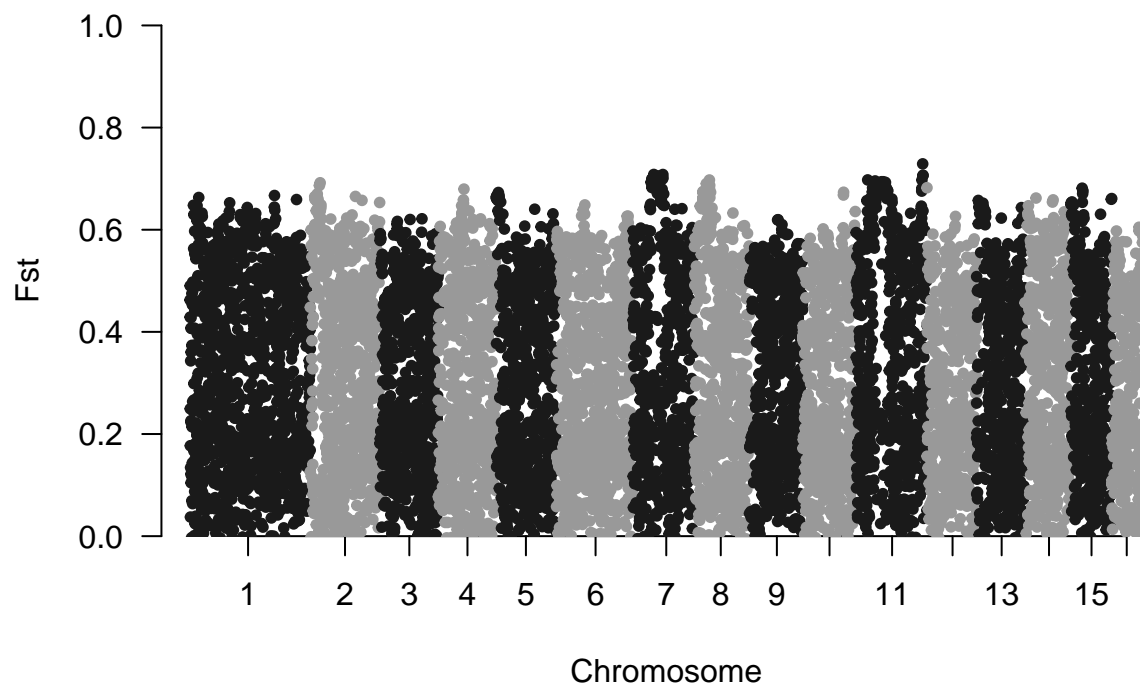
No filter - No LD pruning - 10030 SNPs

```
setwd("~/Documents/Stage_NB/data/SeqApiPop_629_SNPsBeeMuSe")

# Charger les données
fst_data_10030 <- read.table("SeqApiPop_629_SNPsBeeMuSe_filtered_fst.fst", header=TRUE)

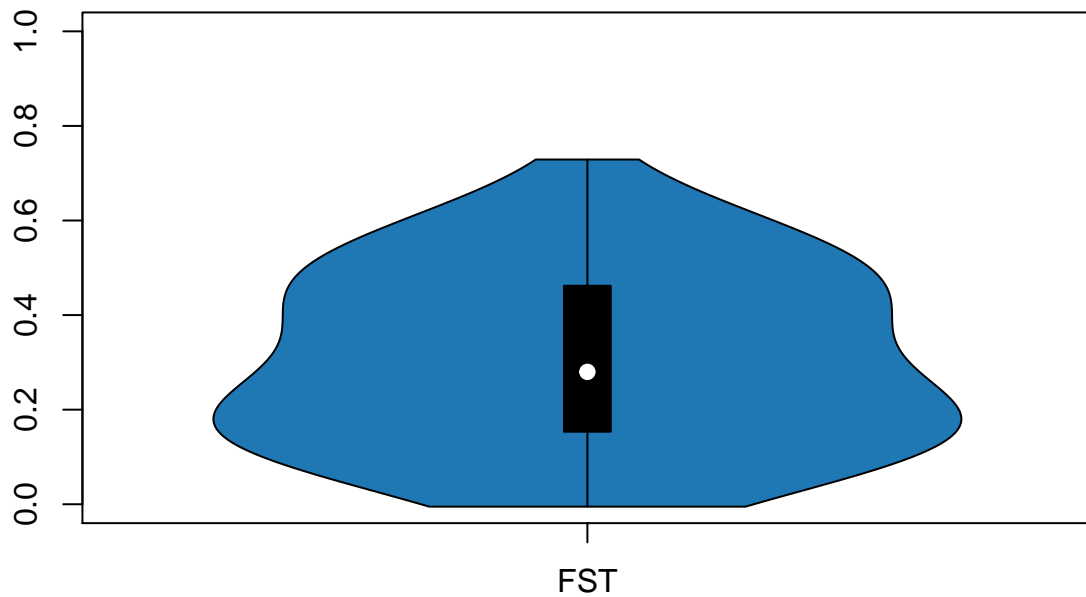
fstsubset <- fst_data_10030[complete.cases(fst_data_10030),]
SNP <- c(1:(nrow(fstsubset)))
mydf <- data.frame(SNP, fstsubset)

# Manhattan plot FST
manhattan(mydf, chr="CHR", bp="POS", p="FST"
, snp="SNP", logp=FALSE, ylab="Fst")
```



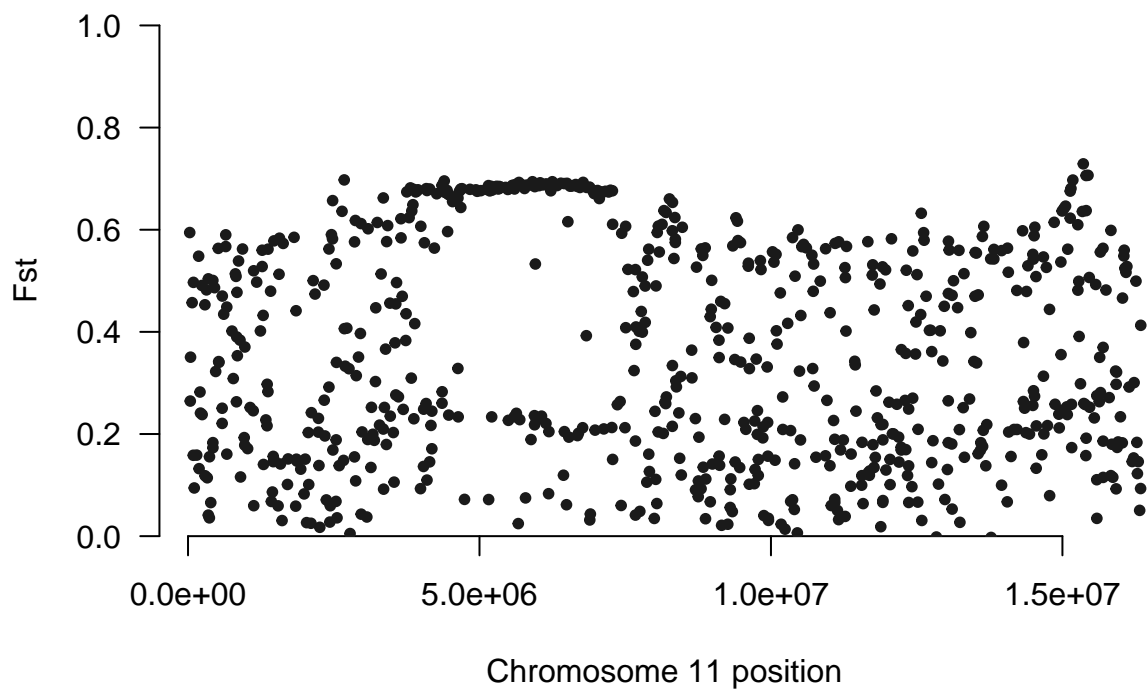
```
# Créer un violin plot des valeurs de FST
vioplot(fst_data_10030$FST, names="FST", col="#1f77b4",ylim=c(0, 1), horizontal=FALSE, main="Violin Plot")
```

## Violin Plot FST – 10030 SNPs



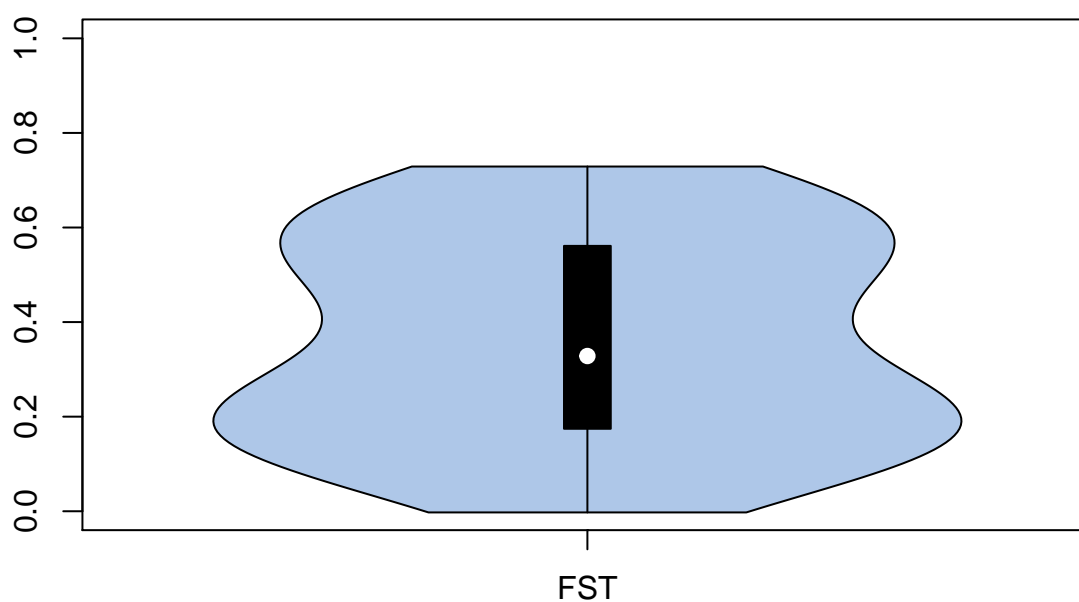
```
# Filtrer les données pour inclure uniquement les positions du chromosome 11
fst_data_10030_chr11 <- subset(fst_data_10030, CHR == "11")

SNP <- seq_len(nrow(fst_data_10030_chr11))
mydf_11 <- data.frame(SNP = SNP, fst_data_10030_chr11)
manhattan(mydf_11, chr = "CHR", bp = "POS", p = "FST", snp = "SNP", logp = FALSE, ylab = "Fst")
```



```
vioplot(fst_data_10030_chr11$FST, names="FST", col="#aec7e8", ylim=c(0, 1),horizontal=FALSE, main="Viol.
```

## Violin Plot FST – Chr 11 – 10030 SNPs



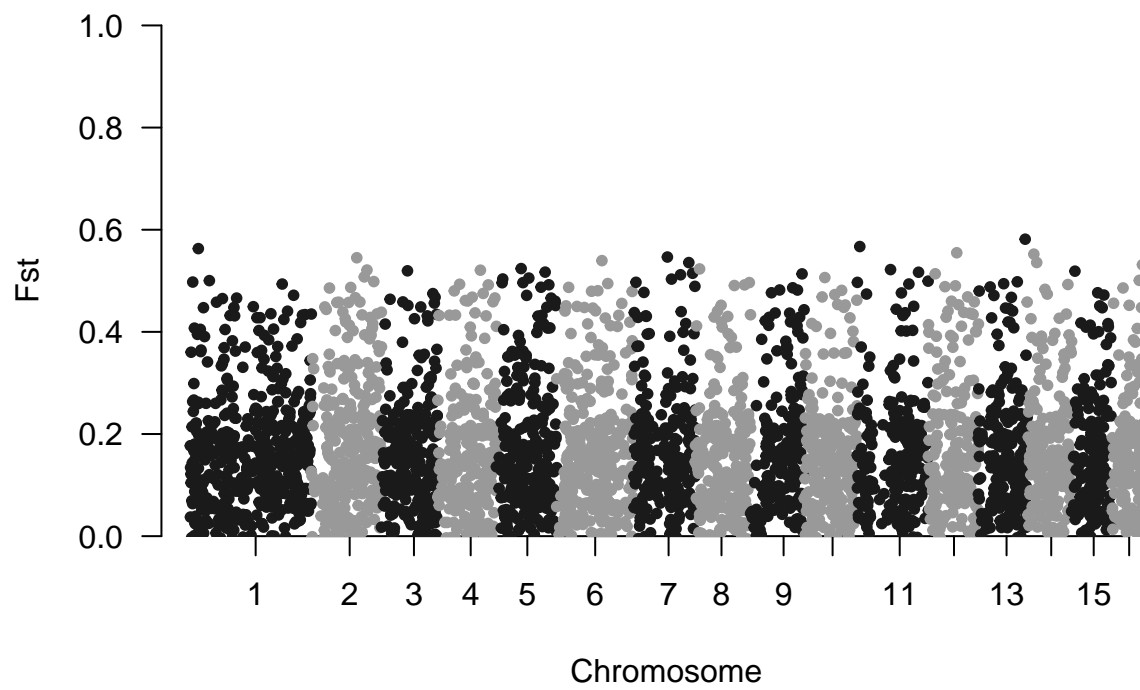
MAF > 0.01 - LD pruning = 0.3 (fenêtre de 1749 SNPS et pas de 175 bp) - 3848 SNPs

```
setwd("~/Documents/Stage_NB/data/SeqApiPop_629_SNPsBeeMuSe")

# Charger les données
fst_data_3848 <- read.table("SeqApiPop_629_SNPsBeeMuSe_filtered_maf001_LD03_pruned_fst.fst", header=TRUE)

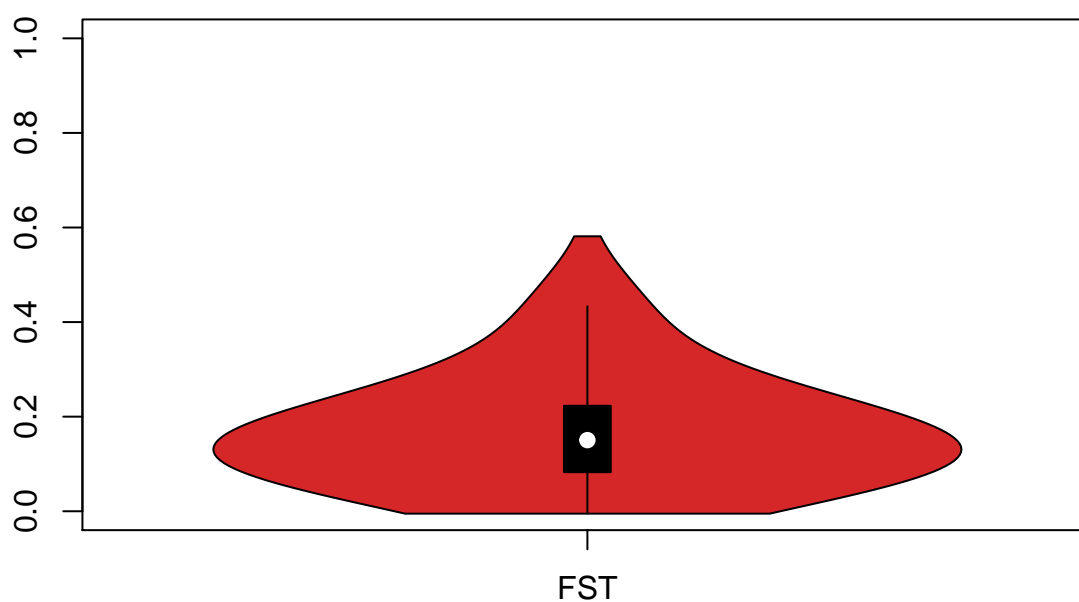
fstsubset <- fst_data_3848[complete.cases(fst_data_3848),]
SNP <- c(1:(nrow(fstsubset)))
mydf <- data.frame(SNP, fstsubset)

# Manhattan plot FST
manhattan(mydf, chr="CHR", bp="POS", p="FST",
           , snp="SNP", logp=FALSE, ylab="Fst")
```



```
vioplot(fst_data_3848$FST, names="FST", col="#d62728", ylim=c(0, 1),horizontal=FALSE, main="Violin Plot")
```

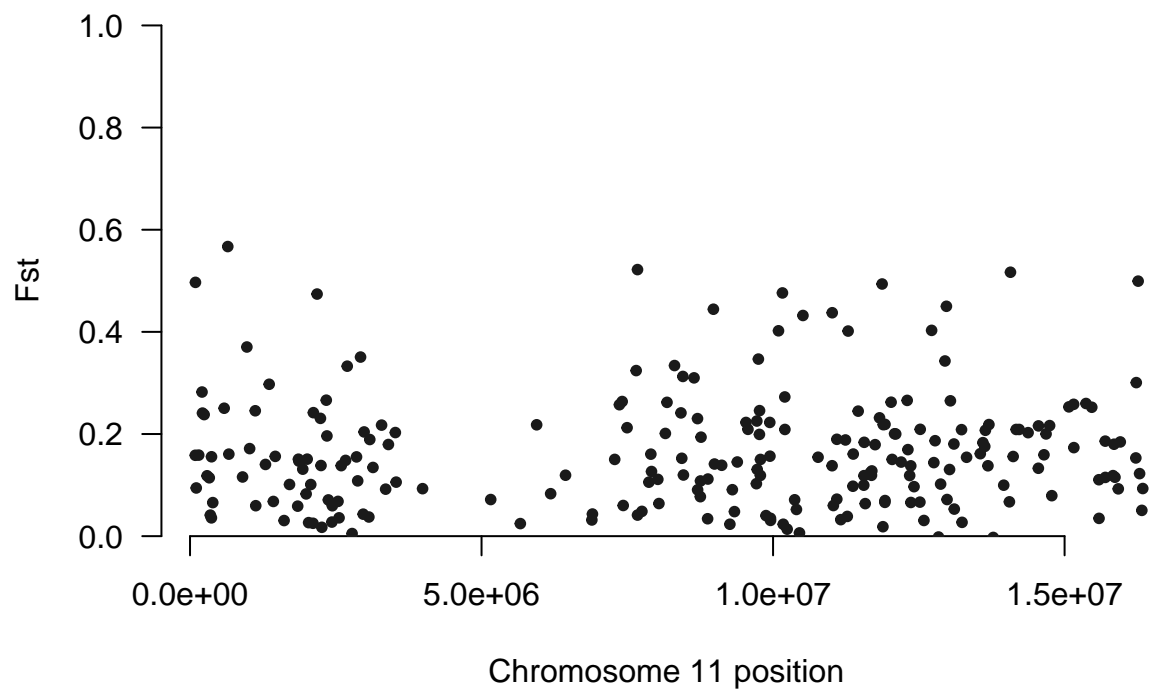
## Violin Plot FST – 3848 SNPs



```
# Filtrer les données pour inclure uniquement les positions du chromosome 11
fst_data_3848_chr11 <- subset(fst_data_3848, CHR == "11")

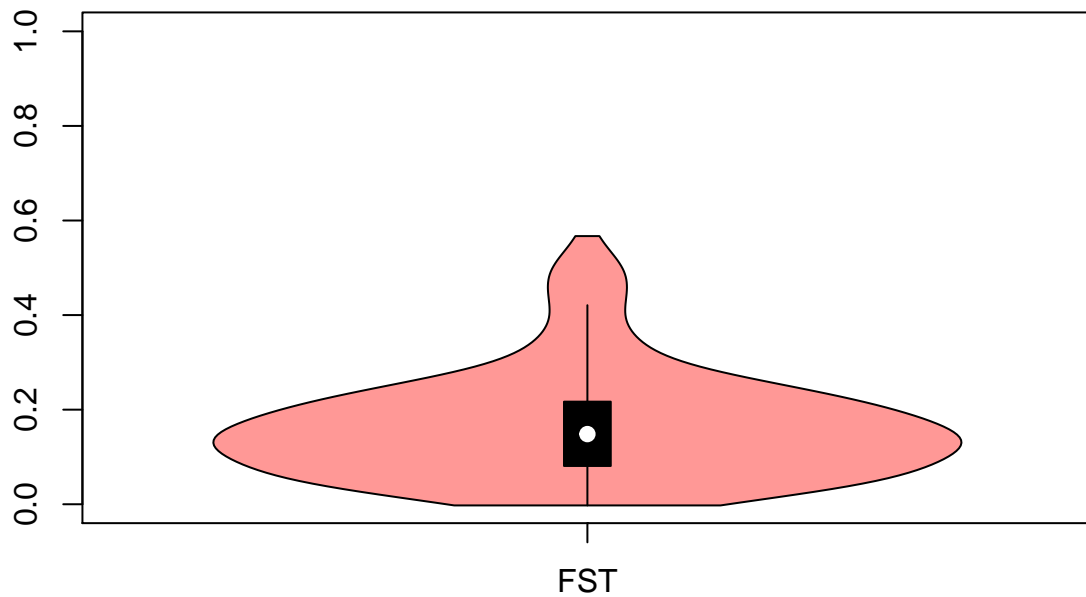
SNP <- seq_len(nrow(fst_data_3848_chr11))
mydf_11 <- data.frame(SNP = SNP, fst_data_3848_chr11)
manhattan(mydf_11, chr = "CHR", bp = "POS", p = "FST", snp = "SNP", logp = FALSE, ylab = "Fst")
```





```
vioplot(fst_data_3848_chr11$FST, names="FST", col="#ff9896",ylim=c(0, 1), horizontal=FALSE, main="Violin")
```

## Violin Plot FST – Chr 11 – 3848 SNPs



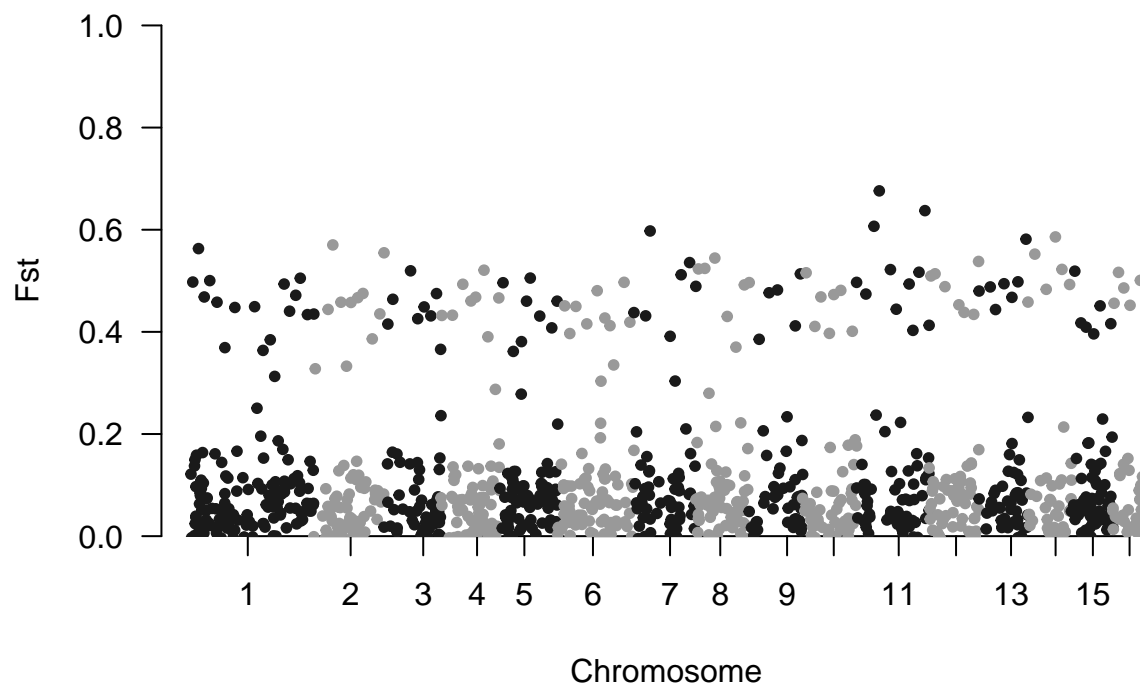
MAF > 0.01 - LD pruning = 0.1 (fenêtre de 50 SNPS et pas de 10 bp) - 1055 SNPs

```
setwd("~/Documents/Stage_NB/data/SeqApiPop_629_SNPsBeeMuSe")

# Charger les données
fst_data_1055 <- read.table("SeqApiPop_629_SNPsBeeMuSe_filtered_maf001_LD03_default_pruned_fst.fst", header=TRUE)

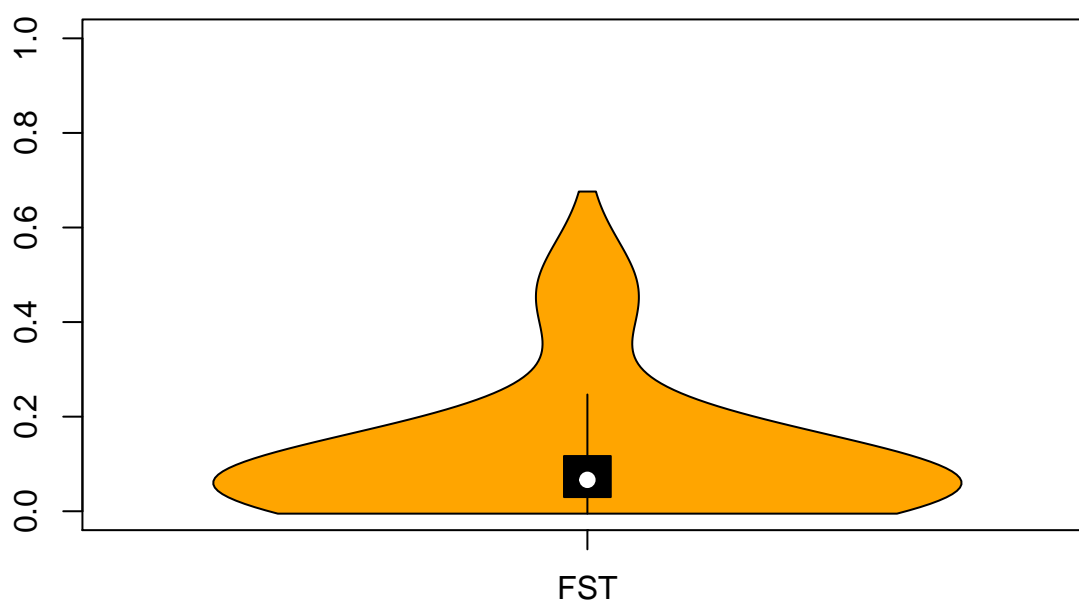
fstsubset <- fst_data_1055[complete.cases(fst_data_1055),]
SNP <- c(1:(nrow(fstsubset)))
mydf <- data.frame(SNP, fstsubset)

# Manhattan plot FST
manhattan(mydf, chr="CHR", bp="POS", p="FST",
           ,snp="SNP", logp=FALSE, ylab="Fst")
```



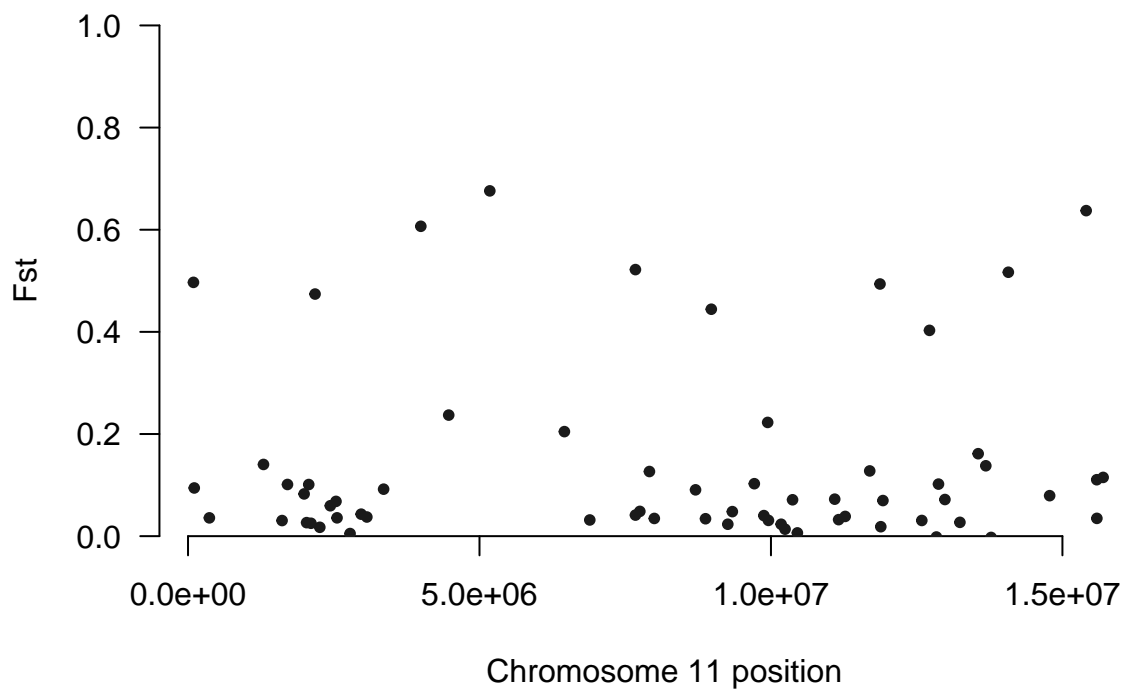
```
vioplot(fst_data_1055$FST, names="FST", col="#FFA500", ylim=c(0, 1),horizontal=FALSE, main="Violin Plot
```

## Violin Plot FST – 1055 SNPs



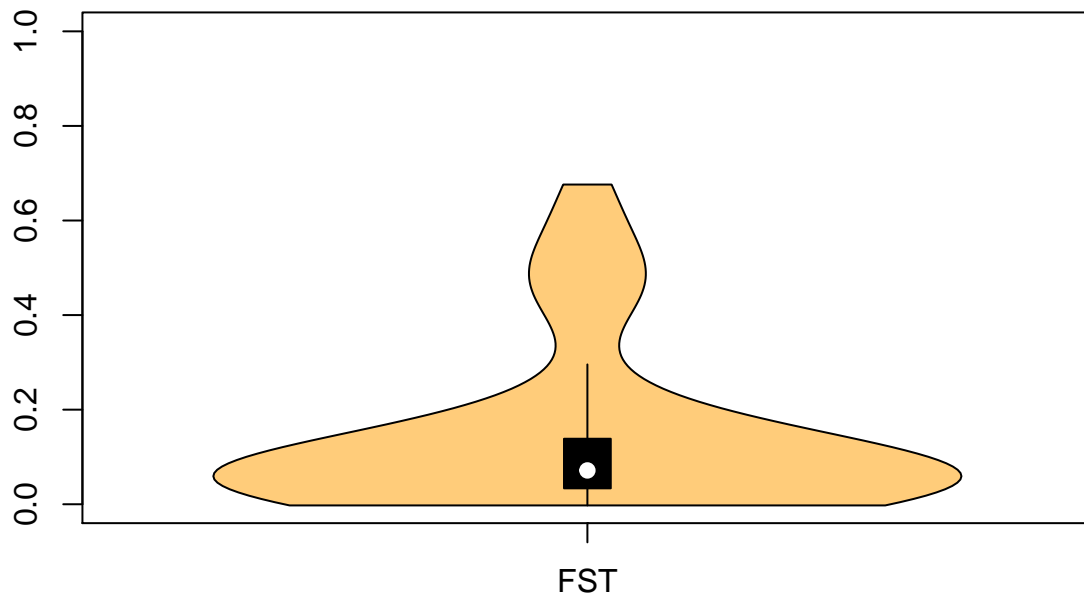
```
# Filtrer les données pour inclure uniquement les positions du chromosome 11
fst_data_1055_chr11 <- subset(fst_data_1055, CHR == "11")

SNP <- seq_len(nrow(fst_data_1055_chr11))
mydf_11 <- data.frame(SNP = SNP, fst_data_1055_chr11)
manhattan(mydf_11, chr = "CHR", bp = "POS", p = "FST", snp = "SNP", logp = FALSE, ylab = "Fst")
```



```
vioplot(fst_data_1055_chr11$FST, names="FST", col="#ffcc7a",ylim=c(0, 1), horizontal=FALSE, main="Violin")
```

## Violin Plot FST – Chr 11 – 1055 SNPs



```
# Création d'un facteur pour distinguer les différentes données
fst_data_10030$Dataset <- "fst_data_10030"
fst_data_10030_chr11$Dataset <- "fst_data_10030_chr11"
fst_data_3848$Dataset <- "fst_data_3848"
fst_data_3848_chr11$Dataset <- "fst_data_3848_chr11"
fst_data_1055$Dataset <- "fst_data_1055"
fst_data_1055_chr11$Dataset <- "fst_data_1055_chr11"

fst_dataset <- rbind(fst_data_10030, fst_data_10030_chr11, fst_data_3848, fst_data_3848_chr11, fst_data_1055,
                    fst_data_1055_chr11)

group_order <- c("fst_data_10030", "fst_data_10030_chr11",
                  "fst_data_3848", "fst_data_3848_chr11",
                  "fst_data_1055", "fst_data_1055_chr11")
fst_dataset$Dataset <- factor(fst_dataset$Dataset, levels = group_order)

ggplot(fst_dataset, aes(x = Dataset, y = FST, fill = Dataset)) +
  geom_violin(trim = FALSE) +
  geom_boxplot(width = 0.1) +
  scale_fill_manual(values = c("fst_data_10030" = "#1f77b4",
                                "fst_data_10030_chr11" = "#aec7e8",
                                "fst_data_3848" = "#d62728",
                                "fst_data_3848_chr11" = "#ff9896",
                                "fst_data_1055" = "#FFA500",
                                "fst_data_1055_chr11" = "#ffcc7a")) +
  labs(title = "Violin Plot of FST values",
       x = "Group",
```

```

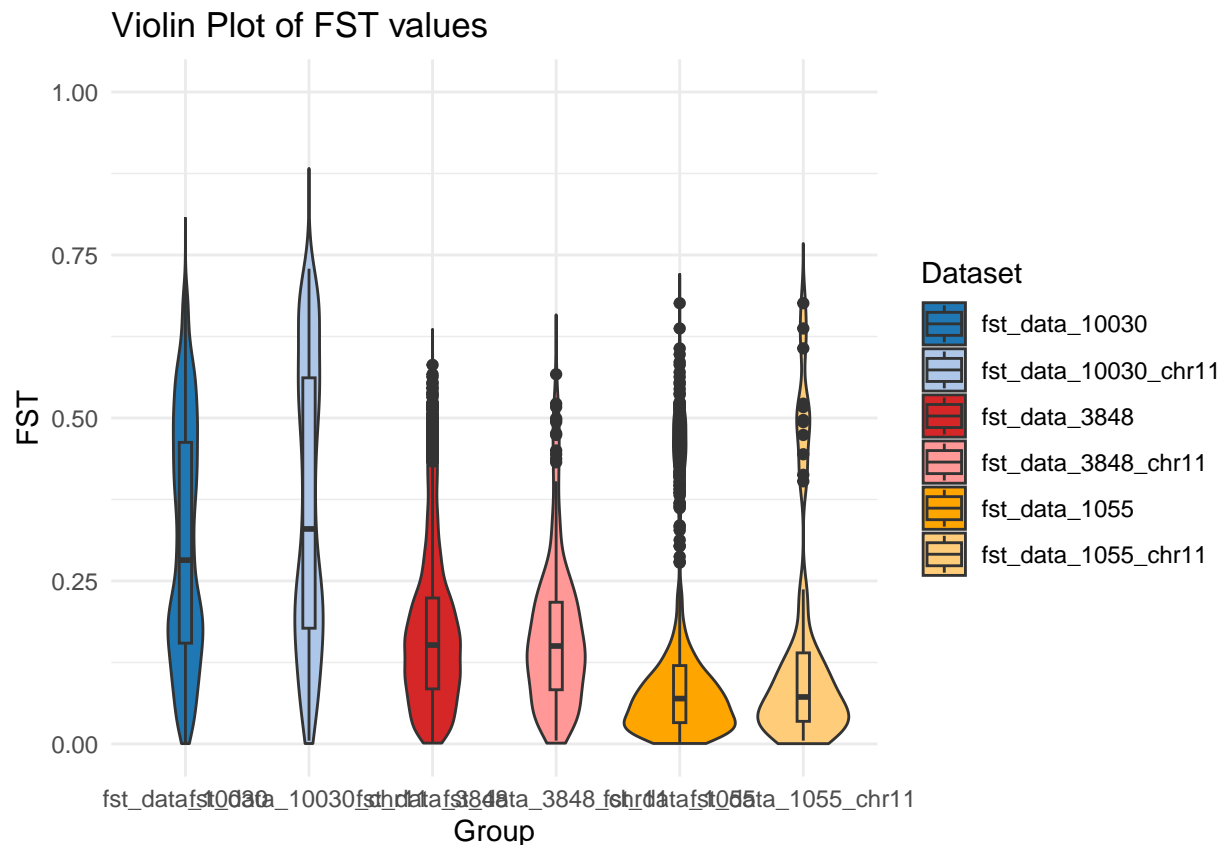
y = "FST") +
theme_minimal() +
ylim(0, 1)

```

```
## Warning: Removed 116 rows containing non-finite values ('stat_ydensity()').
```

```
## Warning: Removed 116 rows containing non-finite values ('stat_boxplot()').
```

```
## Warning: Removed 302 rows containing missing values ('geom_violin()').
```



```

mean_values <- aggregate(FST ~ Dataset, data = fst_dataset, FUN = mean)
print(mean_values)

```

```

##           Dataset      FST
## 1  fst_data_10030 0.3050629
## 2 fst_data_10030_chr11 0.3594720
## 3   fst_data_3848 0.1691465
## 4 fst_data_3848_chr11 0.1649481
## 5   fst_data_1055 0.1148031
## 6 fst_data_1055_chr11 0.1421054

```