

# Stage\_M2\_NB\_4\_IBS\_IBD\_KINSHIP

Nicolas Bettembourg

2024-04-22

## Contents

Chargement des packages R . . . . .	1
<b>Analyse du degré d'apparentement IBD (Identity By Descent) et de similarité IBS (Identity By Similarity)</b>	2
SeqApiPop . . . . .	2
629 échantillons - SNPsBeeMuse - MAF > 0.01 - LD pruning = 0.3 (fenêtre de 1749 SNPs et pas de 175 bp) . . . . .	2
BeeMuSe . . . . .	17
748 échantillons - SNPsBeeMuSe filtered - 10256 SNPs . . . . .	17
“plink –genome –keep” . . . . .	19
Analyse IBD entre les 748 échantillons . . . . .	19
Analyse IBD - ID_2a - 29 groupes + Unknown (pedigree inconnue) . . . . .	27
Analyse IBD - BBS_6-19 . . . . .	36
Analyse IBD - KBJ_1-19 . . . . .	40
Analyse IBD - MM_31-20 . . . . .	44
“plink –genome” . . . . .	47
Analyse IBD entre les 748 échantillons . . . . .	48
Analyse IBD - ID_2a - 29 groupes + Unknown (pedigree inconnue) . . . . .	56
Analyse IBD - BBS_6-19 . . . . .	65
Analyse IBD - KBJ_1-19 . . . . .	69
Analyse IBD - MM_31-20 . . . . .	73
“plink2 –make-king-table” . . . . .	76
Analyse KINSHIP + IBS0 - 748 échantillons . . . . .	76
Analyse IBD intra-familles - ID_2a - 29 groupes + Unknown (pedigree inconnue) .	82

## Chargement des packages R

```

library(ggplot2)
library(reshape2)
library(pheatmap)
library(MASS)

```

## Analyse du degré d'apparentement IBD (Identity By Descent) et de similarité IBS (Identity By Similarity)

SeqApiPop

629 échantillons - SNPsBeeMuse - MAF > 0.01 - LD pruning = 0.3 (fenêtre de 1749 SNPs et pas de 175 bp)

```

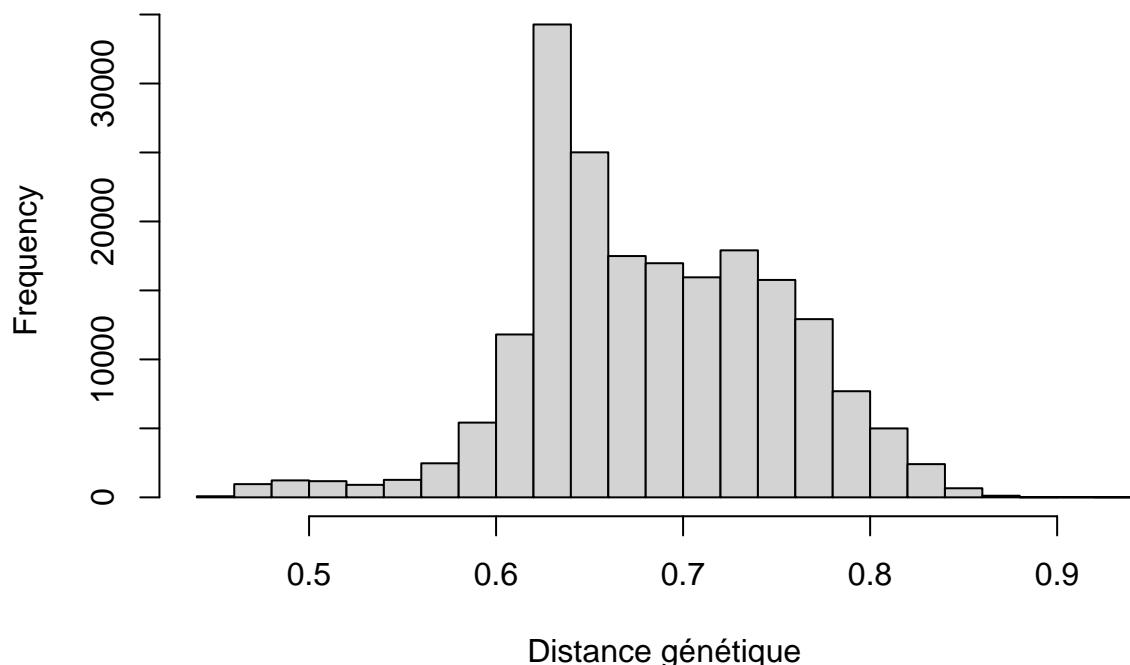
setwd("~/Documents/Stage_NB/data/IBD")

# Charger les données des coefficients de parenté
relatedness <- read.table("SeqApiPop_629_SNPsBeeMuSe_filtered_maf001_LD03_pruned_ibd.genome", header=TRUE)

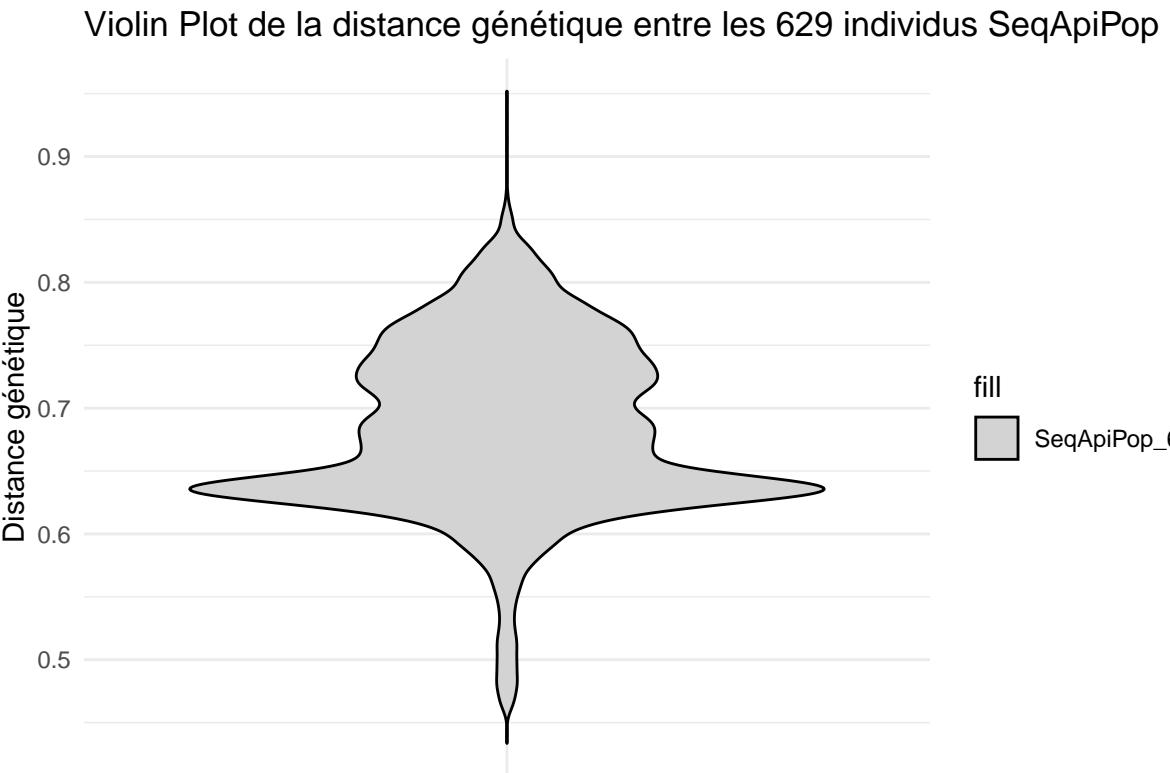
hist(relatedness$DST, breaks=20, col="lightgrey", main="Histogramme de la distance génétique entre les 629 individus SeqApiPop")

```

Histogramme de la distance génétique entre les 629 individus SeqApiPop

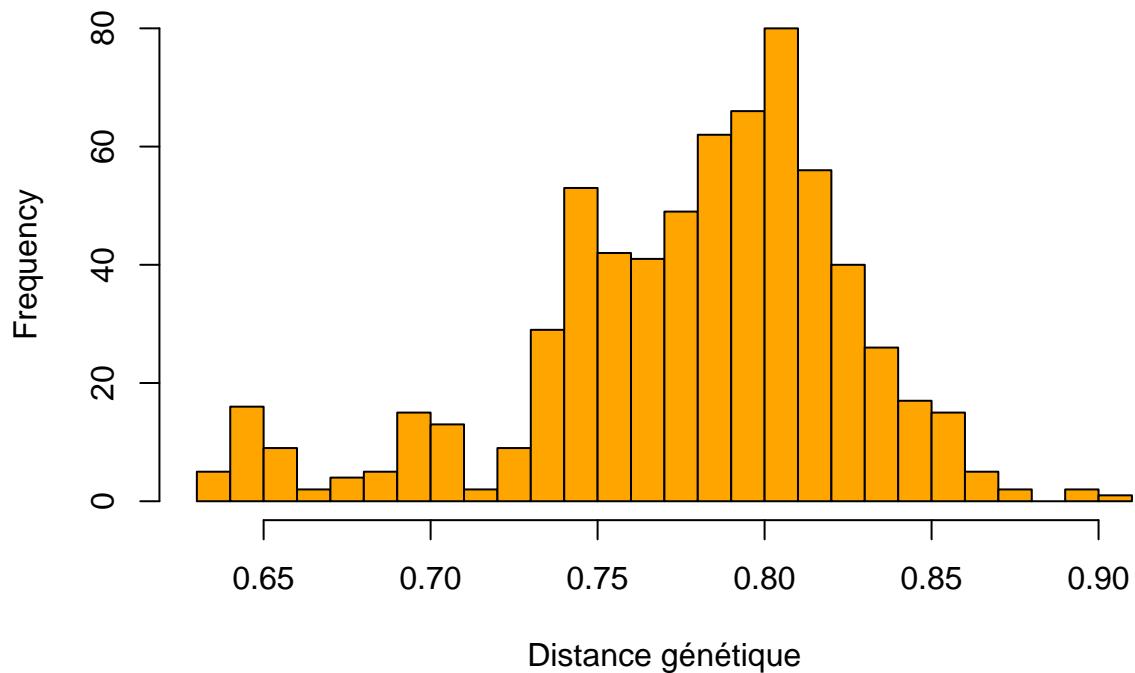


```
# Créer un violin plot pour la distance génétique entre les individus SeqApiPop
ggplot(relatedness, aes(x = "", y = DST, fill = "SeqApiPop_629")) +
  geom_violin(trim = FALSE, color = "black") +
  scale_fill_manual(values = c("SeqApiPop_629" = "lightgrey")) +
  labs(title = "Violin Plot de la distance génétique entre les 629 individus SeqApiPop",
       x = "", y = "Distance génétique") +
  theme_minimal()
```



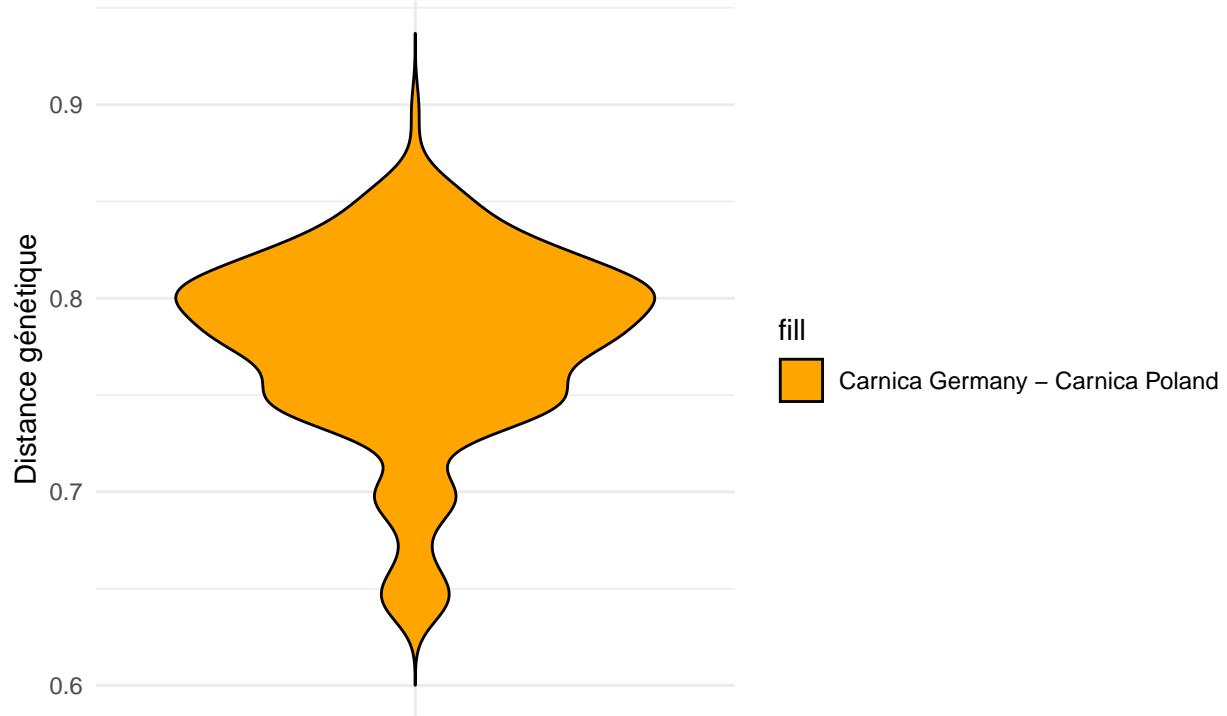
```
relatedness_car_ger_pol<- read.table("SeqApiPop_629_SNPsBeeMuSe_filtered_maf001_LD03_pruned_ibd_carnica"
hist(relatedness_car_ger_pol$DST, breaks=20, col="orange", main="Histogramme de la distance génétique e
```

### Histogramme de la distance génétique entre individus Carnica Germany – Carnica Poland



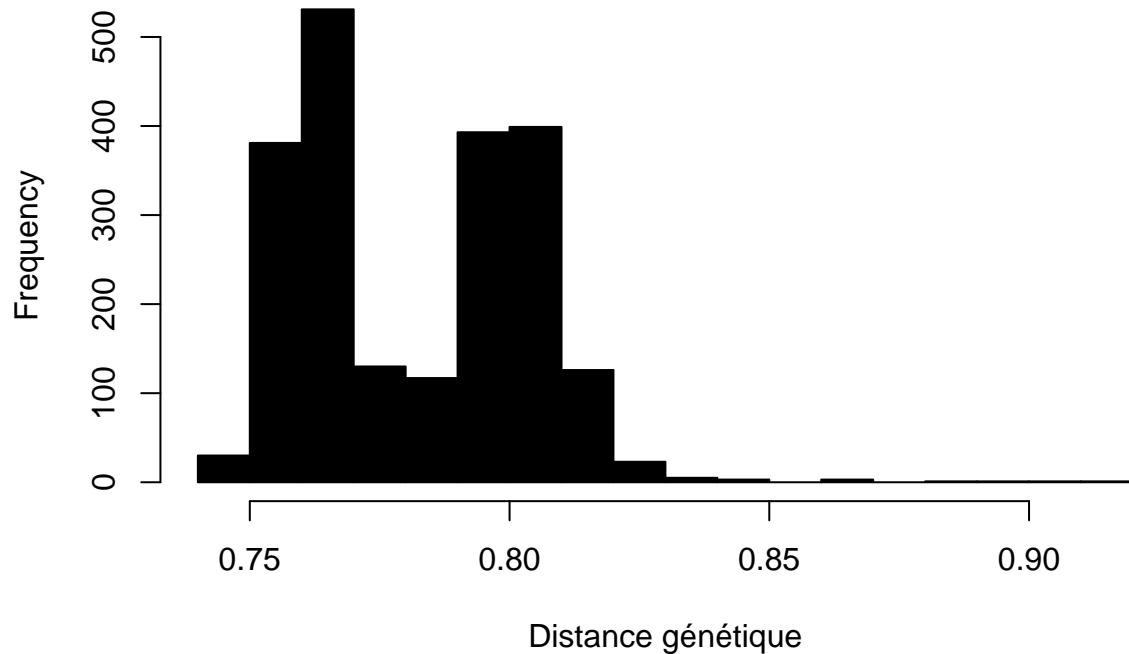
```
# Créer un violin plot pour la distance génétique entre individus Carnica Germany – Carnica Poland
ggplot(relatedness_car_ger_pol, aes(x = "", y = DST, fill = "Carnica Germany – Carnica Poland")) +
  geom_violin(trim = FALSE, color = "black") +
  scale_fill_manual(values = c("Carnica Germany – Carnica Poland" = "orange")) +
  labs(title = "Violin Plot de la distance génétique entre individus Carnica Germany – Poland",
       x = "", y = "Distance génétique") +
  theme_minimal()
```

Violin Plot de la distance génétique entre individus Carnica Germany – Poland



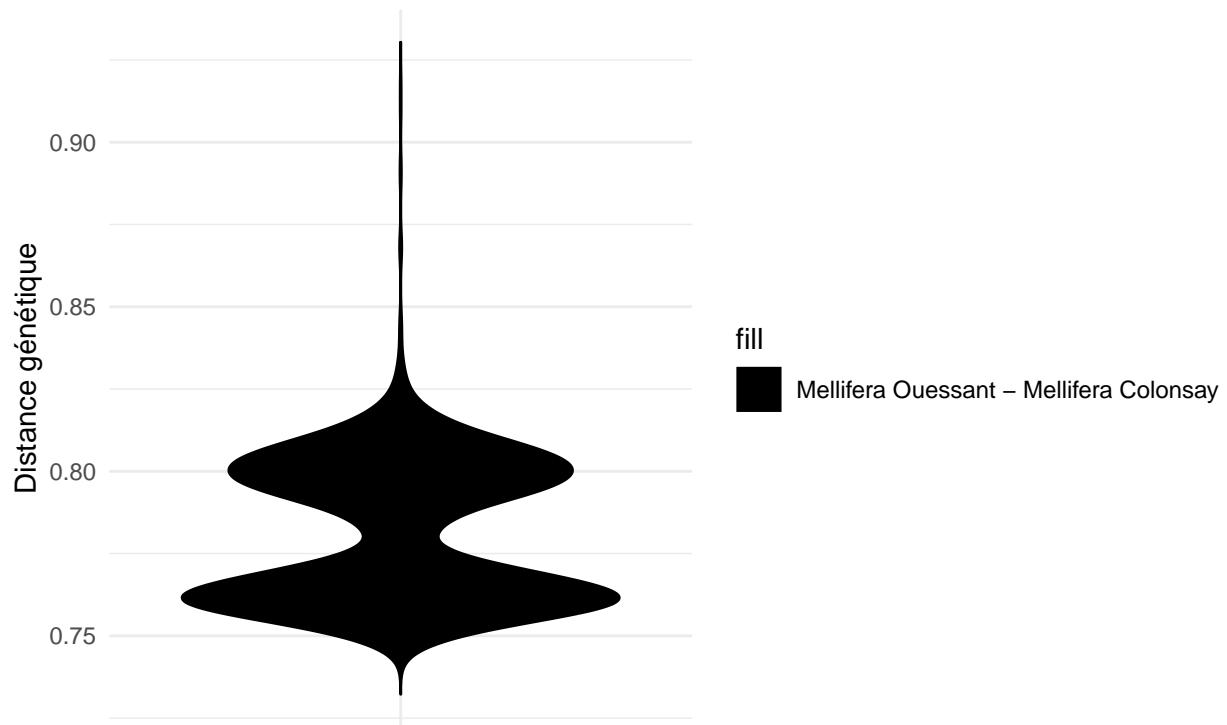
```
relatedness_mel_oue_col <- read.table("SeqApiPop_629_SNPsBeeMuSe_filtered_maf001_LD03_pruned_ibd_melliflhist(relatedness_mel_oue_col$DST, breaks=20, col="black", main="Histogramme de la distance génétique en
```

## Histogramme de la distance génétique entre individus Mellifera Ouessant – Mellifera Colonsay



```
# Créer un violin plot pour la distance génétique entre individus Mellifera Ouessant – Mellifera Colonsay
ggplot(relatedness_mel_oue_col, aes(x = "", y = DST, fill = "Mellifera Ouessant – Mellifera Colonsay"))
  geom_violin(trim = FALSE, color = "black") +
  scale_fill_manual(values = c("Mellifera Ouessant – Mellifera Colonsay" = "black")) +
  labs(title = "Violin Plot de la distance génétique entre individus Mellifera Ouessant – Colonsay",
       x = "", y = "Distance génétique") +
  theme_minimal()
```

## Violin Plot de la distance génétique entre individus Mellifera Ouessant – Colonsay



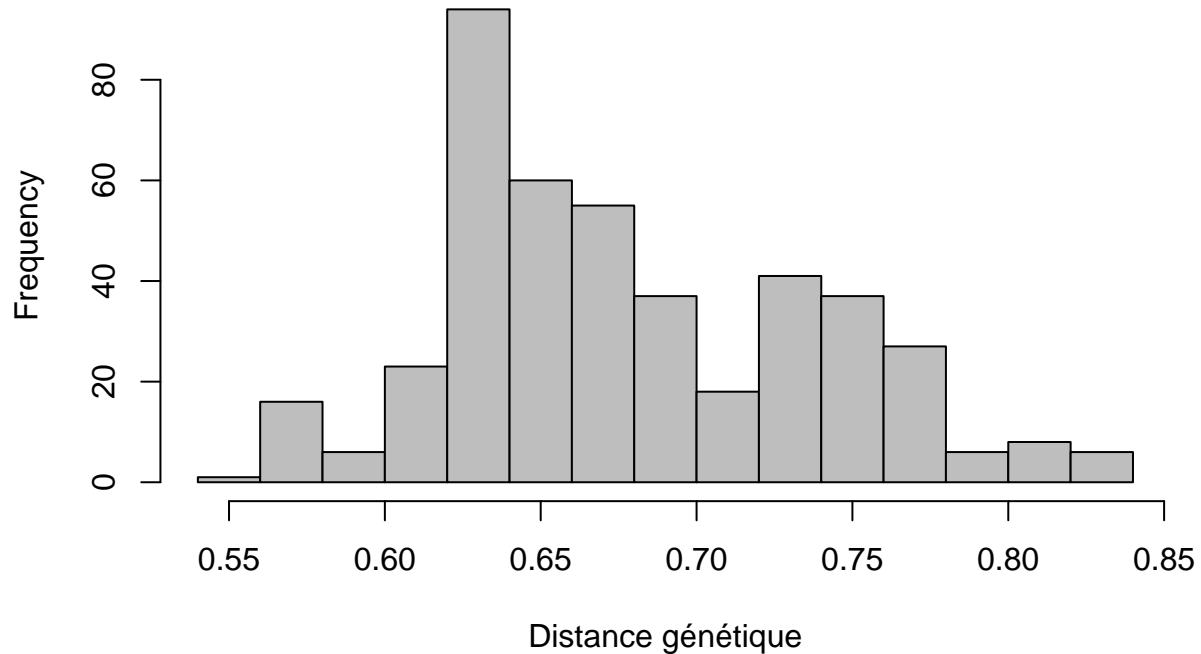
```
# Un individu représentant de chaque population de SeqApiPop
```

```
# Charger les données des coefficients de parenté
```

```
relatedness_rep <- read.table("SeqApiPop_629_SNPsBeeMuSe_filtered_maf001_LD03_pruned_ibd_representants.txt")
```

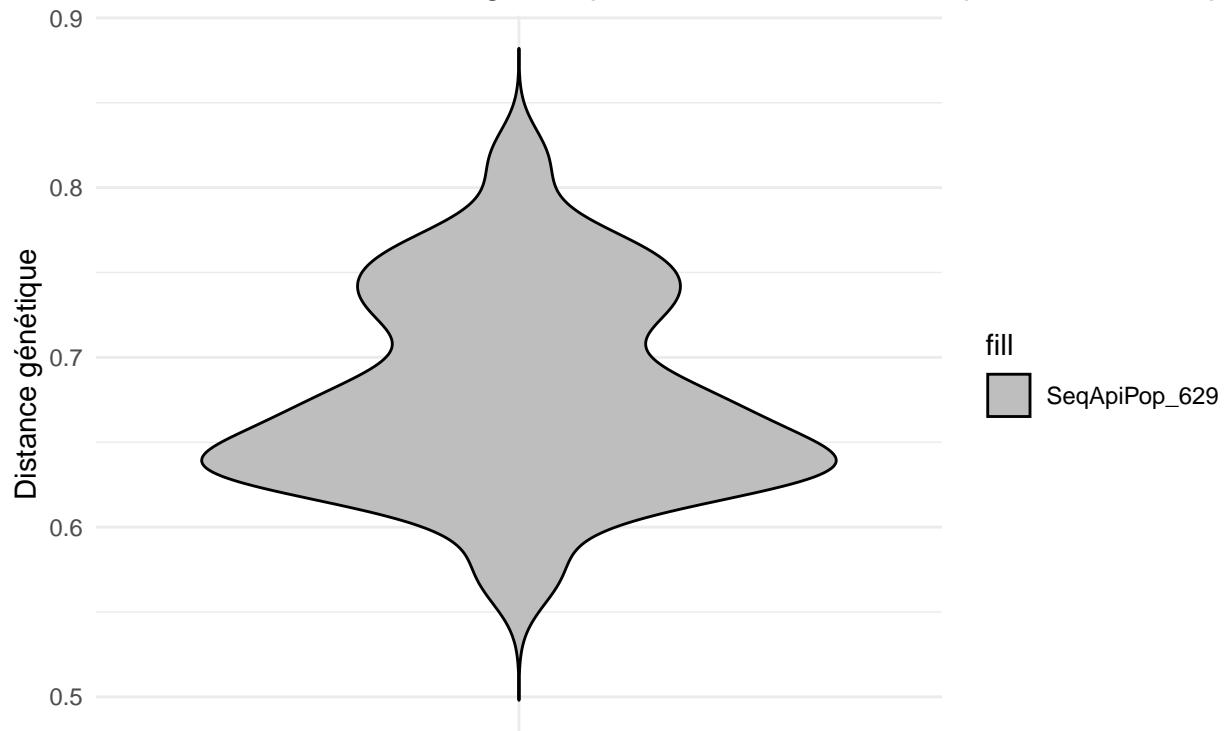
```
hist(relatedness_rep$DST, breaks=20, col="grey", main="Histogramme de la distance génétique entre les indi
```

### Histogramme de la distance génétique entre les individus représentants SeqApiPop



```
# Créer un violin plot pour la distance génétique entre les individus SeqApiPop
ggplot(relatedness_rep, aes(x = "", y = DST, fill = "SeqApiPop_629")) +
  geom_violin(trim = FALSE, color = "black") +
  scale_fill_manual(values = c("SeqApiPop_629" = "grey")) +
  labs(title = "Violin Plot de la distance génétique entre les individus représentants SeqApiPop",
       x = "", y = "Distance génétique") +
  theme_minimal()
```

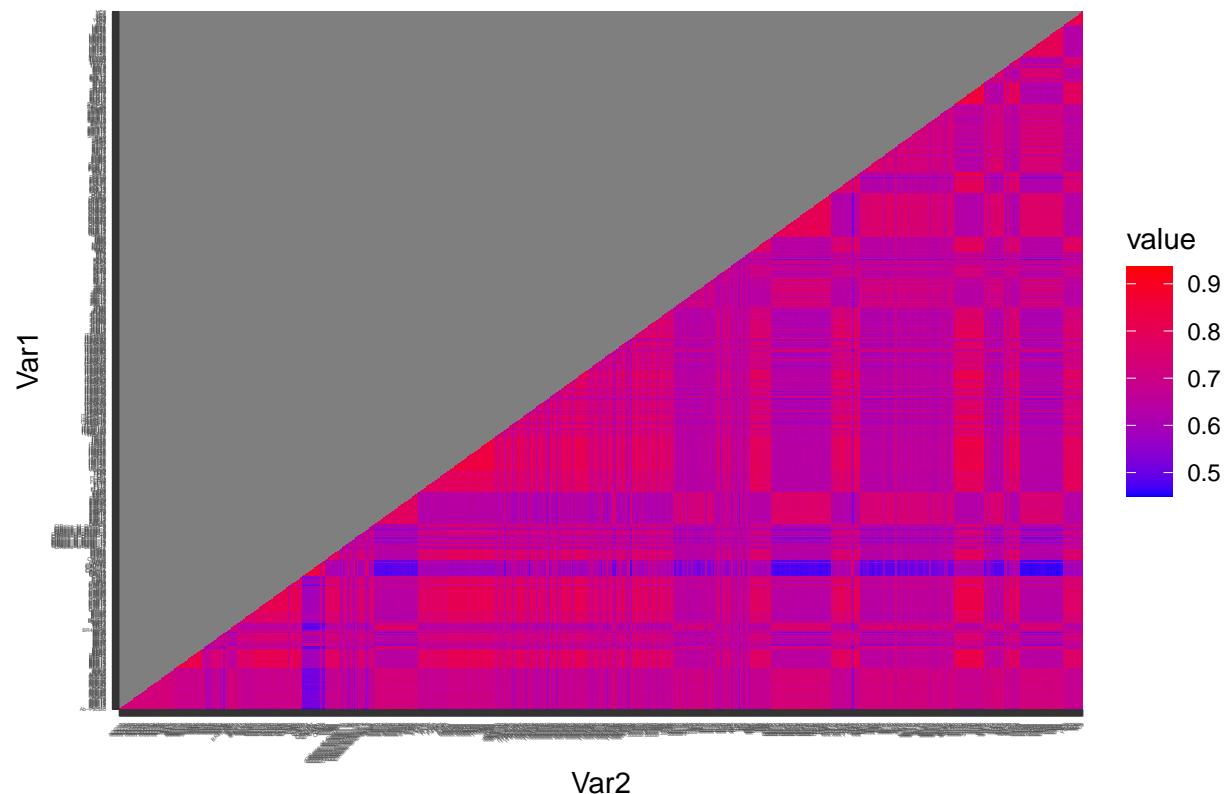
## Violin Plot de la distance génétique entre les individus représentants SeqAfr



```
matrice_dist <- acast(relatedness, IID1 ~ IID2, value.var = "DST")
matrice_dist2 <- acast(relatedness_car_ger_pol, IID1 ~ IID2, value.var = "DST")
matrice_dist3 <- acast(relatedness_mel_oue_col, IID1 ~ IID2, value.var = "DST")
matrice_dist4 <- acast(relatedness_rep, IID1 ~ IID2, value.var = "DST")

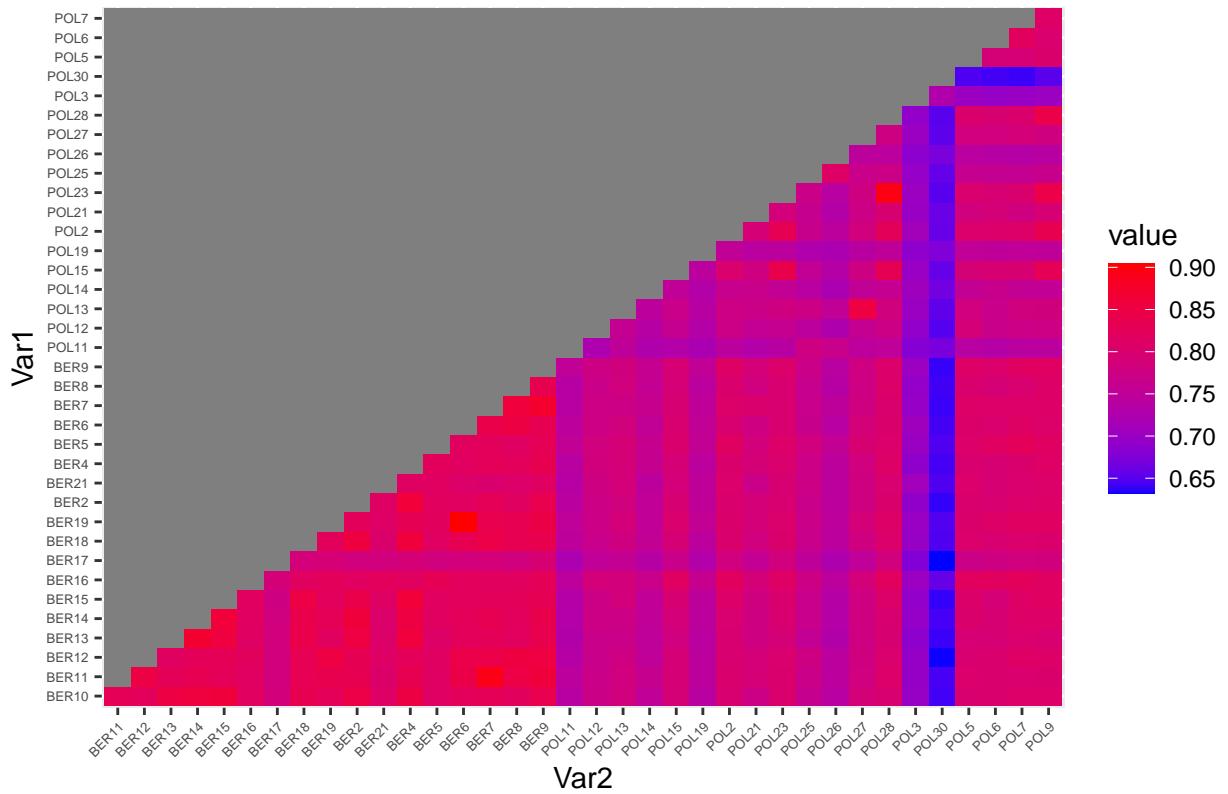
# Créer un graphique heatmap avec ggplot
ggplot(data = melt(matrice_dist), aes(x = Var2, y = Var1, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap de la matrice de distance génétique") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 2)) +
  theme(axis.text.y = element_text(hjust = 1, size = 2))
```

Heatmap de la matrice de distance génétique



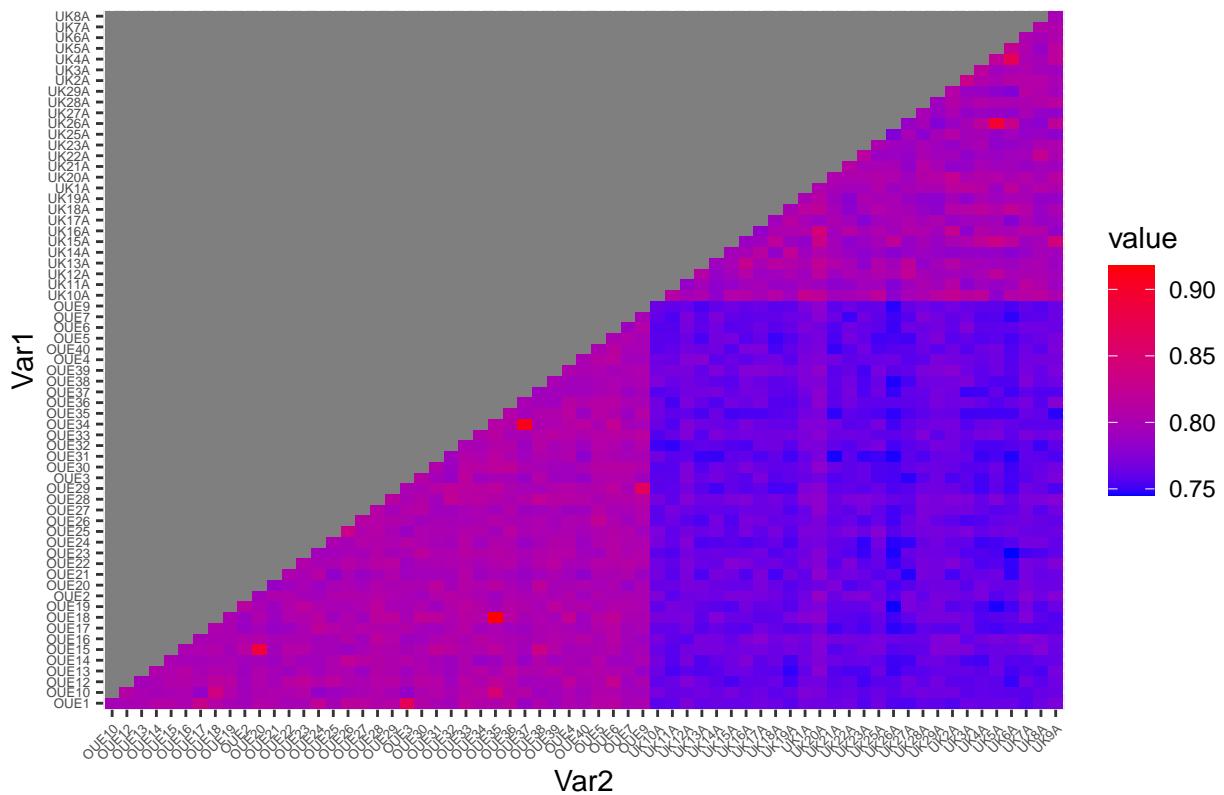
```
ggplot(data = melt(matrice_dist2), aes(x = Var2, y = Var1, fill = value)) +  
  geom_tile() +  
  scale_fill_gradient(low = "blue", high = "red") +  
  labs(title = "Heatmap de la matrice de distance génétique") +  
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 5)) +  
  theme(axis.text.y = element_text(hjust = 1, size = 5))
```

## Heatmap de la matrice de distance génétique



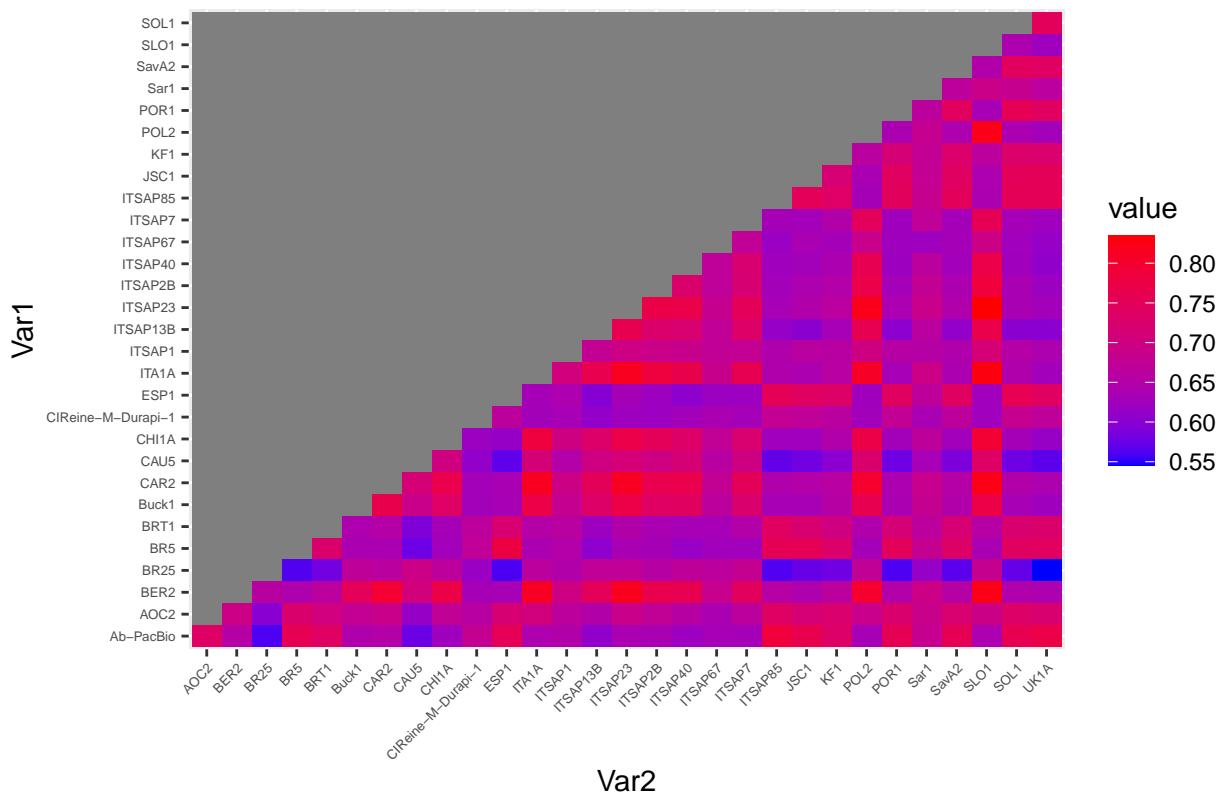
```
ggplot(data = melt(matrice_dist3), aes(x = Var2, y = Var1, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap de la matrice de distance génétique") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 5)) +
  theme(axis.text.y = element_text(hjust = 1, size = 5))
```

## Heatmap de la matrice de distance génétique



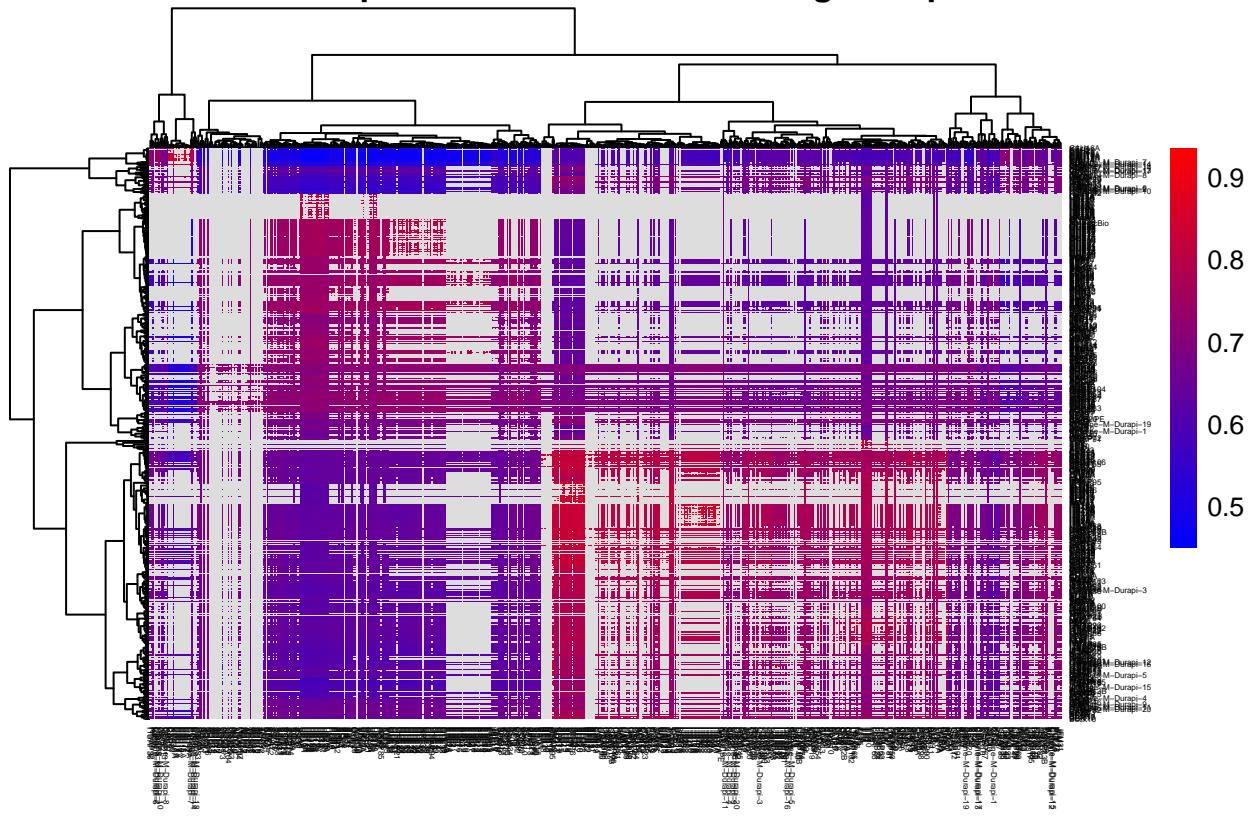
```
ggplot(data = melt(matrice_dist4), aes(x = Var2, y = Var1, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap de la matrice de distance génétique") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 5)) +
  theme(axis.text.y = element_text(hjust = 1, size = 5))
```

## Heatmap de la matrice de distance génétique



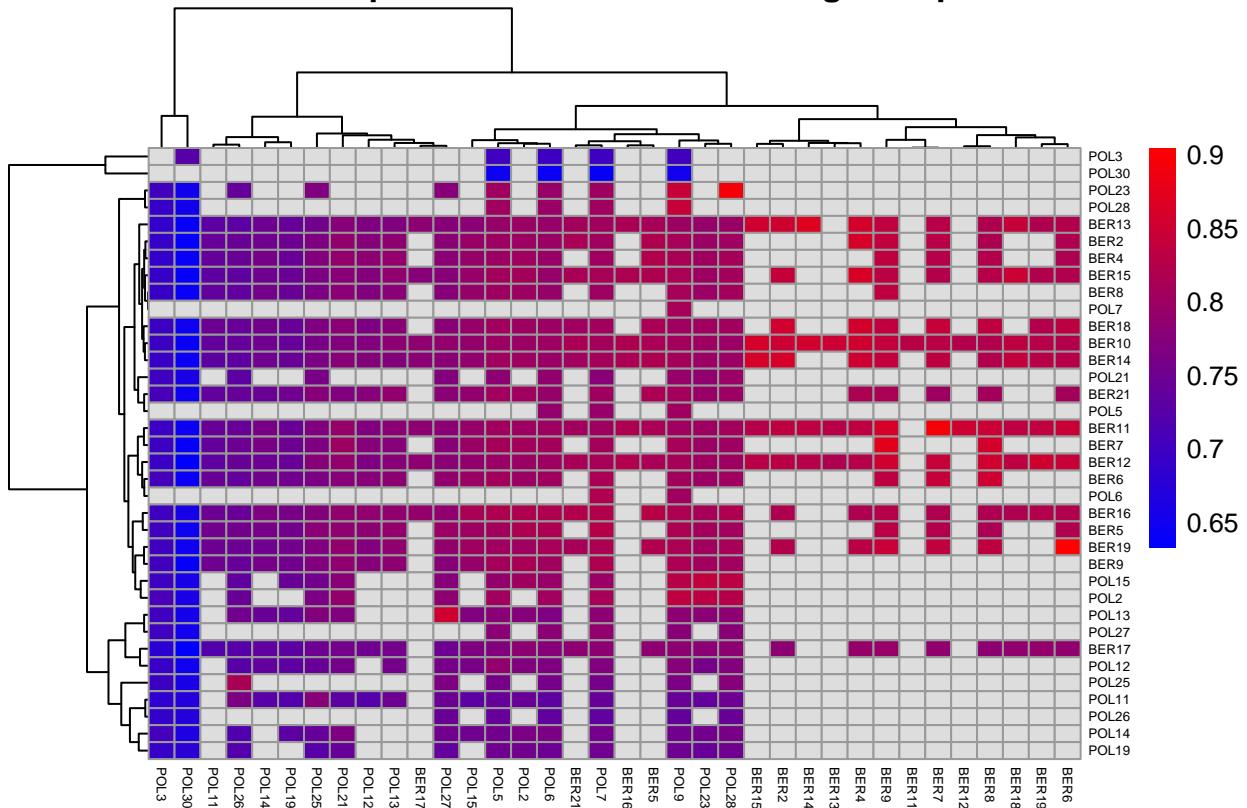
```
# Créer un graphique heatmap avec pheatmap
pheatmap(matrice_dist,
          color = colorRampPalette(c("blue", "red"))(256),
          fontsize_row = 3, fontsize_col = 3,
          main = "Heatmap de la matrice de distance génétique")
```

## Heatmap de la matrice de distance génétique

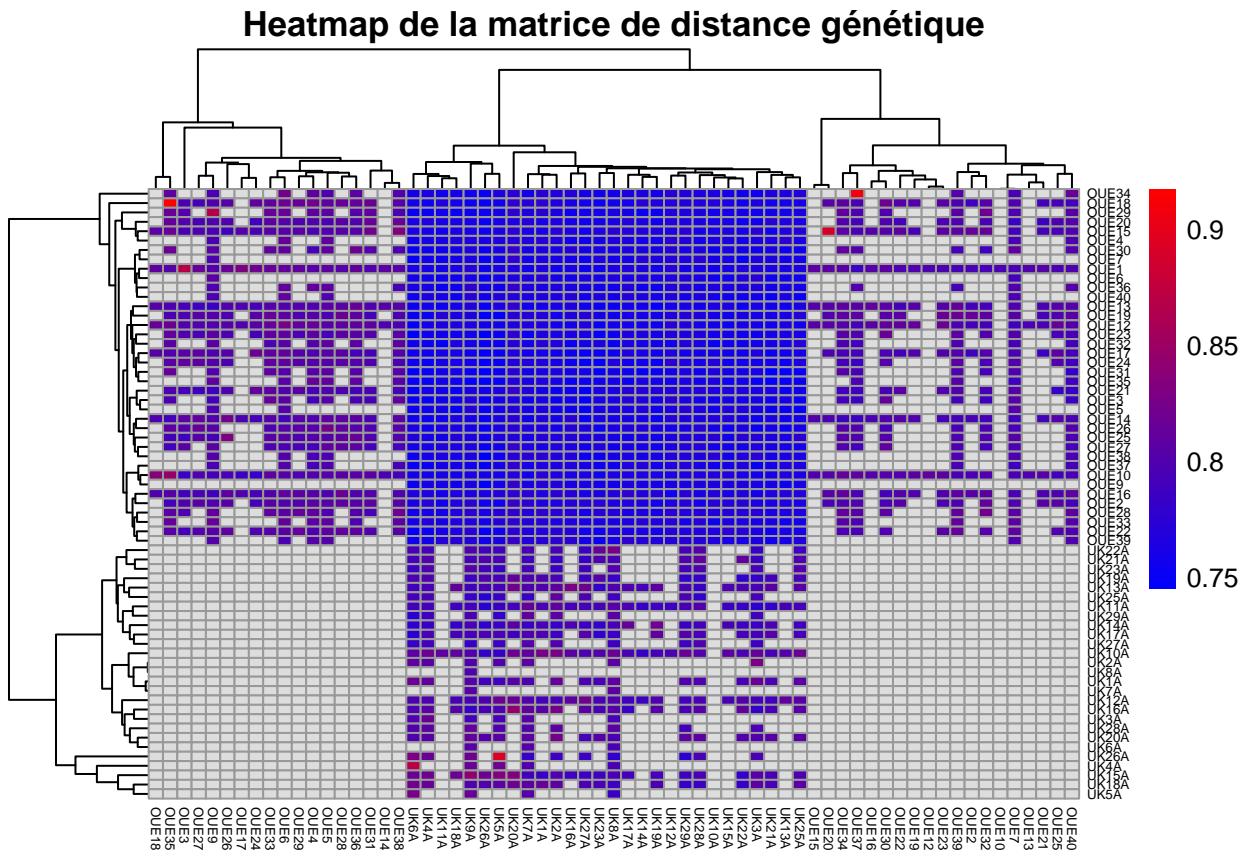


```
pheatmap(matrice_dist2,
          color = colorRampPalette(c("blue", "red"))(256),
          fontsize_row = 5, fontsize_col = 5,
          main = "Heatmap de la matrice de distance génétique")
```

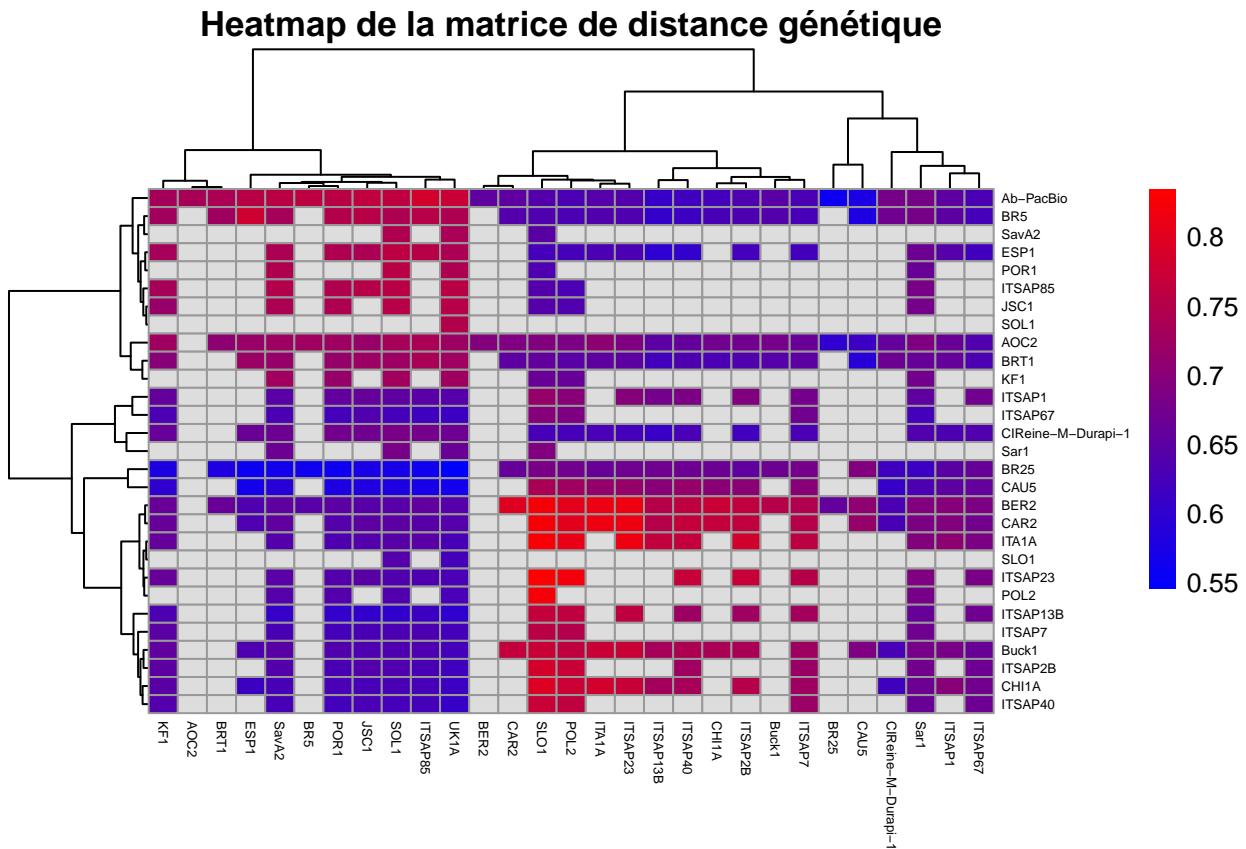
### Heatmap de la matrice de distance génétique



```
pheatmap(matrice_dist3,
          color = colorRampPalette(c("blue", "red"))(256),
          fontsize_row = 5, fontsize_col = 5,
          main = "Heatmap de la matrice de distance génétique")
```



```
pheatmap(matrice_dist4,
         color = colorRampPalette(c("blue", "red"))(256),
         fontsize_row = 5, fontsize_col = 5,
         main = "Heatmap de la matrice de distance génétique")
```



## BeeMuSe

748 échantillons - SNPsBeeMuSe filtered - 10256 SNPs

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

beemuse_genome <- read.table("BeeMuse_genome_full.genome", header=TRUE)
beemuse_BAH_20_19_genome <- read.table("BeeMuse_BAH_20-19_genome.genome", header=TRUE)
beemuse_BBS_6_19_genome <- read.table("BeeMuse_BBS_6-19_genome.genome", header=TRUE)
beemuse_BER_11_19_genome <- read.table("BeeMuse_BER_11-19_genome.genome", header=TRUE)
beemuse_BH_44_genome <- read.table("BeeMuse_BH_44_genome.genome", header=TRUE)
beemuse_BH_7_19_genome <- read.table("BeeMuse_BH_7-19_genome.genome", header=TRUE)
beemuse_BHA_2_20_genome <- read.table("BeeMuse_BHA_2-20_genome.genome", header=TRUE)
beemuse_BLS_53_19_genome <- read.table("BeeMuse_BLS_53-19_genome.genome", header=TRUE)
beemuse_ER_13_19_genome <- read.table("BeeMuse_ER_13-19_genome.genome", header=TRUE)
beemuse_KBJ_1_19_genome <- read.table("BeeMuse_KBJ_1-19_genome.genome", header=TRUE)
beemuse_KBru_6_20_genome <- read.table("BeeMuse_KBru_6-20_genome.genome", header=TRUE)
beemuse_KLoc_37_19_genome <- read.table("BeeMuse_KLoc_37-19_genome.genome", header=TRUE)
beemuse_KLSU_14_19_genome <- read.table("BeeMuse_KLSU_14-19_genome.genome", header=TRUE)
beemuse_MM_31_20_genome <- read.table("BeeMuse_MM_31-20_genome.genome", header=TRUE)
beemuse_MM_37_20_genome <- read.table("BeeMuse_MM_37-20_genome.genome", header=TRUE)
beemuse_MP_10_20_genome <- read.table("BeeMuse_MP_10-20_genome.genome", header=TRUE)
beemuse_PersoBC_2021_genome <- read.table("BeeMuse_PersoBC_2021_genome.genome", header=TRUE)
```

```

beemuse_PersoJLL_2021_genome <- read.table("BeeMuse_PersoJLL_2021_genome.genome", header=TRUE)
beemuse_PersoJLL_2022_genome <- read.table("BeeMuse_PersoJLL_2022_genome.genome", header=TRUE)
beemuse_PersoLD_2021_genome <- read.table("BeeMuse_PersoLD_2021_genome.genome", header=TRUE)
beemuse_PersoLD_2022_genome <- read.table("BeeMuse_PersoLD_2022_genome.genome", header=TRUE)
beemuse_PersoUB_2021_genome <- read.table("BeeMuse_PersoUB_2021_genome.genome", header=TRUE)
beemuse_PersoUB_2022_genome <- read.table("BeeMuse_PersoUB_2022_genome.genome", header=TRUE)
beemuse_S_GZ_2_19_genome <- read.table("BeeMuse_S_GZ_2-19_genome.genome", header=TRUE)
beemuse_SBJ_3_19_genome <- read.table("BeeMuse_SBJ_3-19_genome.genome", header=TRUE)
beemuse_SJ_16_20_genome <- read.table("BeeMuse_SJ_16-20_genome.genome", header=TRUE)
beemuse_SJ_24_20_genome <- read.table("BeeMuse_SJ_24-20_genome.genome", header=TRUE)
beemuse_SJ_30_20_genome <- read.table("BeeMuse_SJ_30-20_genome.genome", header=TRUE)
beemuse_TL_13_20_genome <- read.table("BeeMuse_TL_13-20_genome.genome", header=TRUE)
beemuse_TL_19_20_genome <- read.table("BeeMuse_TL_19-20_genome.genome", header=TRUE)
beemuse_unknown_genome <- read.table("BeeMuse_Unknown_genome.genome", header=TRUE)

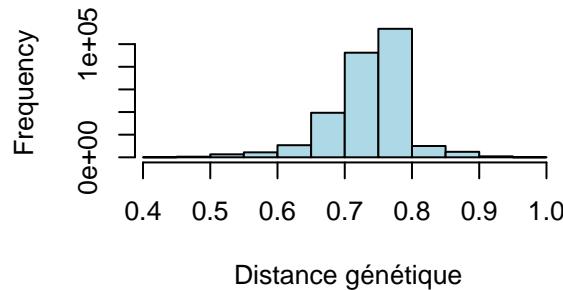
# 1
x_min <- min(c(beemuse_genome$DST, beemuse_BBS_6_19_genome$DST, beemuse_KBJ_1_19_genome$DST, beemuse_BAH_20_19_genome$DST))
x_max <- max(c(beemuse_genome$DST, beemuse_BBS_6_19_genome$DST, beemuse_KBJ_1_19_genome$DST, beemuse_BAH_20_19_genome$DST))

par(mfrow=c(2,2))

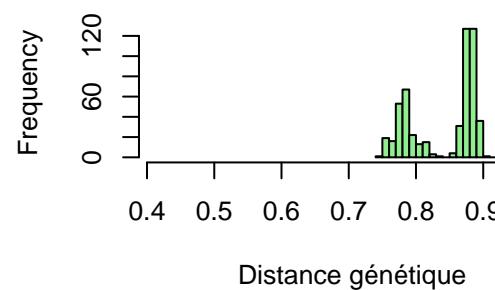
hist(beemuse_genome$DST, breaks=20, col="lightblue", main="Histogramme de la distance génétique (IBS) entre les individus")
hist(beemuse_BBS_6_19_genome$DST, breaks=20, col="lightgreen", main="Histogramme de la distance génétique entre les individus")
hist(beemuse_KBJ_1_19_genome$DST, breaks=20, col="lightpink", main="Histogramme de la distance génétique entre les individus")
hist(beemuse_BAH_20_19_genome$DST, breaks=20, col="lightyellow", main="Histogramme de la distance génétique entre les individus")

```

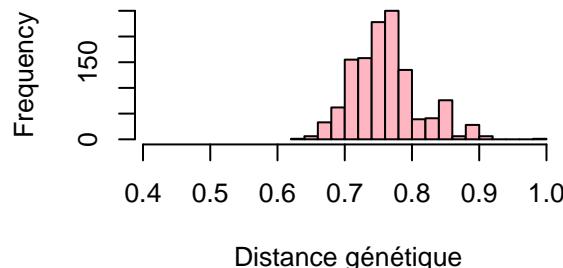
Histogramme de la distance génétique (IBS) entre les 748 échantillons



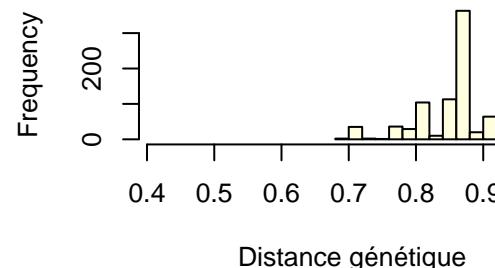
Histogramme de la distance génétique (IBS)



Histogramme de la distance génétique (IBS) – KBJ\_1-15



Histogramme de la distance génétique (IBS)



“plink --genome --keep”

```
similarity_matrix <- acast(beemuse_genome, IID1 ~ IID2, value.var = "PI_HAT")
```

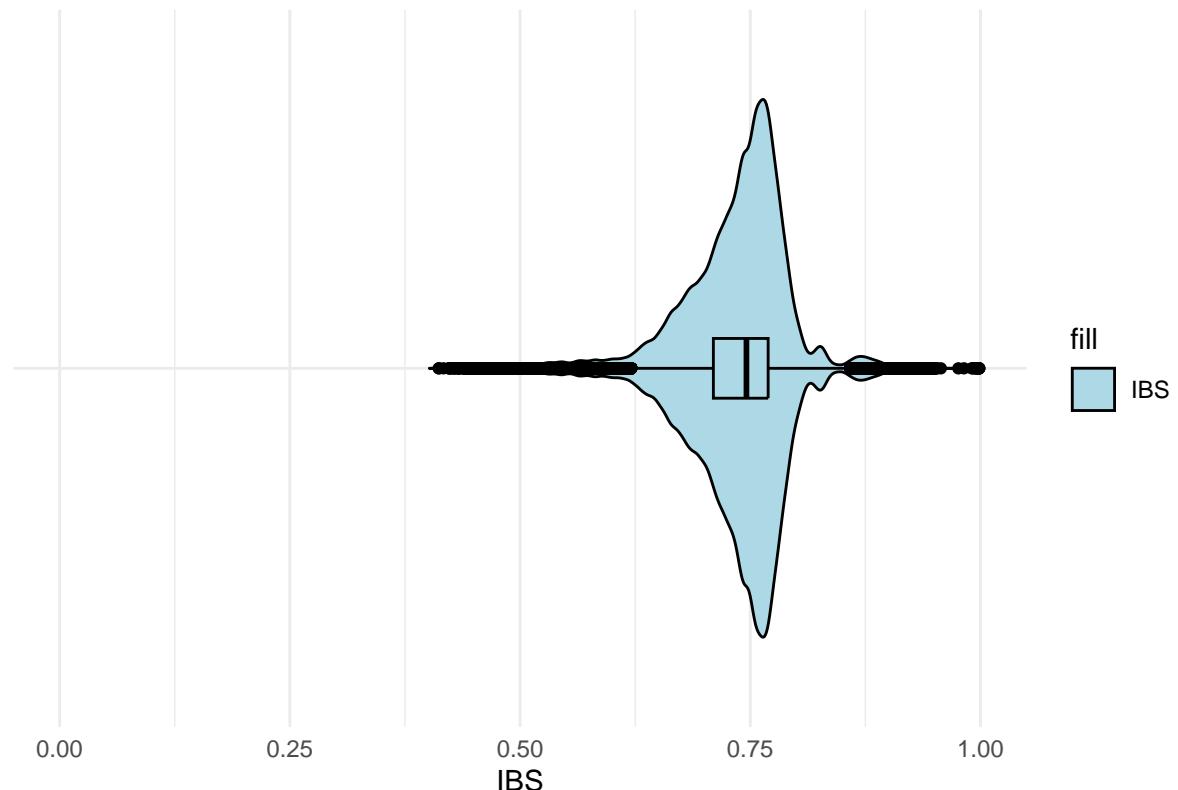
### Analyse IBD entre les 748 échantillons

```
## Aggregation function missing: defaulting to length
```

```
ggplot(beemuse_genome, aes(x = "", y = DST, fill = "IBS")) +  
  geom_violin(trim = FALSE, color = "black") +  
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width = 0.1)) +  
  scale_fill_manual(values = "lightblue") +  
  labs(title = "Violin Plot - IBS",  
       x = "", y = "IBS") +  
  theme_minimal() +  
  scale_y_continuous(limits = c(0, 1)) +  
  coord_flip()
```

```
## Warning: Removed 8 rows containing missing values ('geom_violin()'').
```

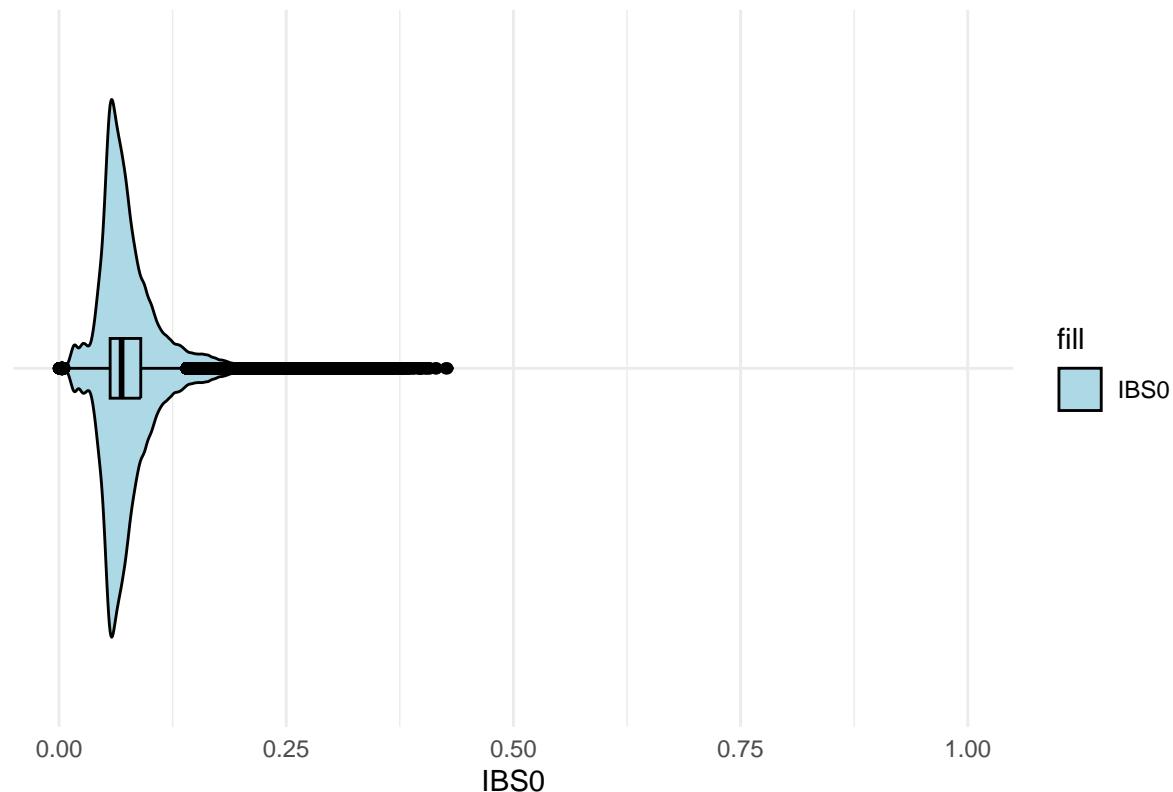
## Violin Plot – IBS



```
ggplot(beemuse_genome, aes(x = "", y = IBS0/(IBS0+IBS1+IBS2), fill = "IBS0")) +  
  geom_violin(trim = FALSE, color = "black") +  
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =  
    scale_fill_manual(values = "lightblue") +  
  labs(title = "Violin Plot - IBS0",  
    x = "", y = "IBS0") +  
  theme_minimal() +  
  scale_y_continuous(limits = c(0, 1)) +  
  coord_flip()
```

```
## Warning: Removed 7 rows containing missing values ('geom_violin()').
```

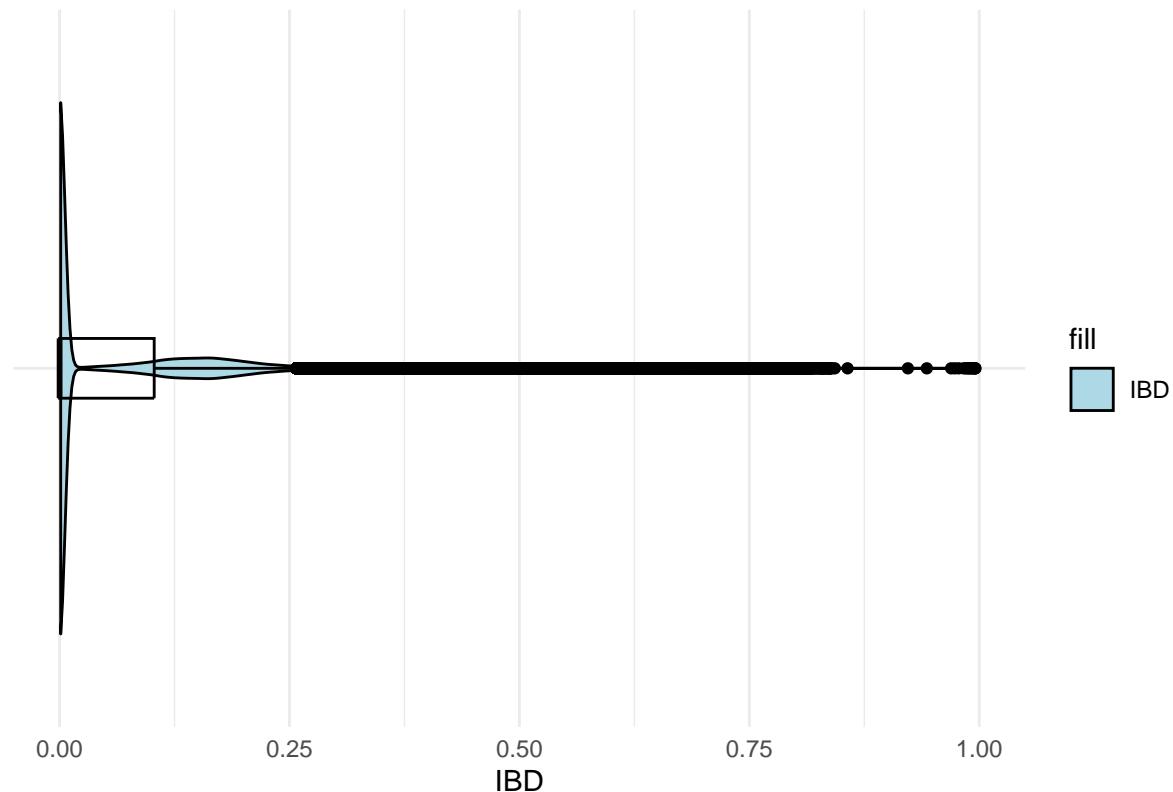
## Violin Plot – IBS0



```
ggplot(beemuse_genome, aes(x = "", y = PI_HAT, fill = "IBD")) +  
  geom_violin(trim = FALSE, color = "black") +  
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =  
    scale_fill_manual(values = "lightblue") +  
  labs(title = "Violin Plot - IBD",  
    x = "", y = "IBD") +  
  theme_minimal() +  
  scale_y_continuous(limits = c(0, 1)) +  
  coord_flip()
```

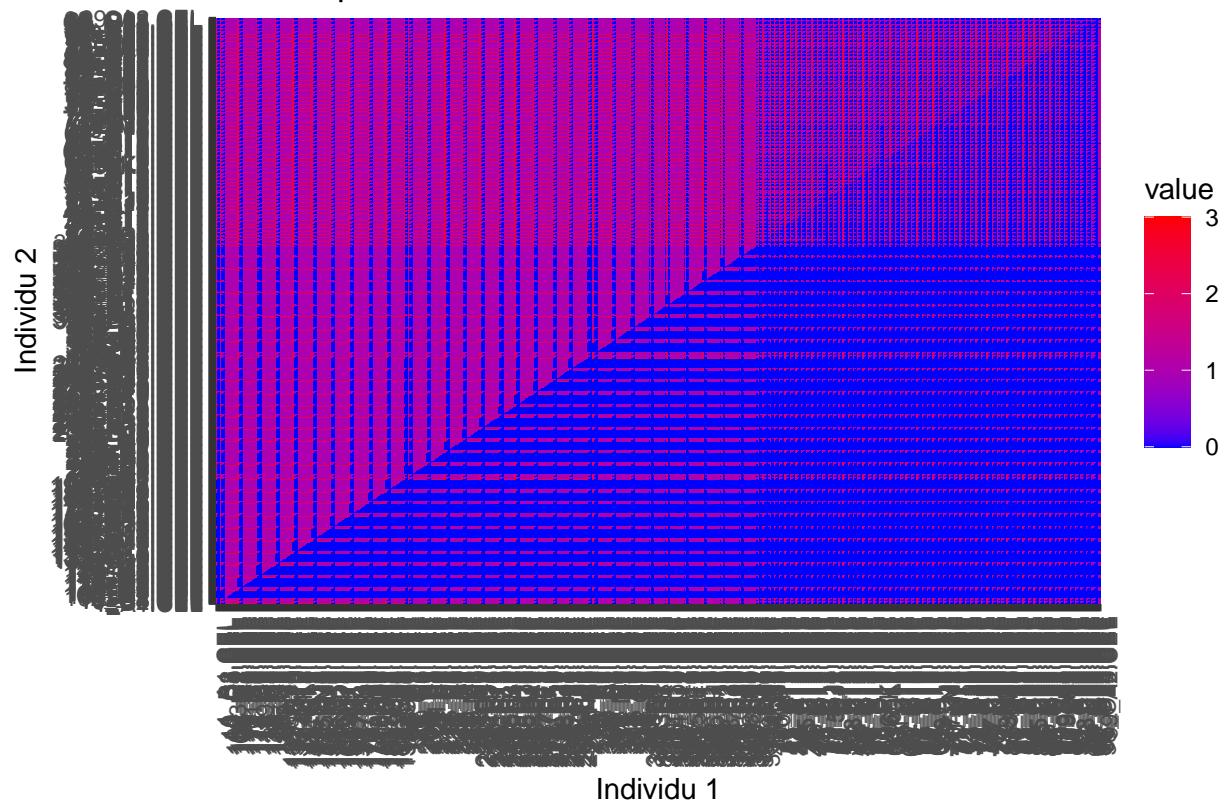
## Warning: Removed 16 rows containing missing values ('geom\_violin()').

## Violin Plot – IBD



```
# Tracer la heatmap
ggplot(melt(similarity_matrix), aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap - IBD - BeeMuSe",
       x = "Individu 1",
       y = "Individu 2") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Heatmap – IBD – BeeMuSe

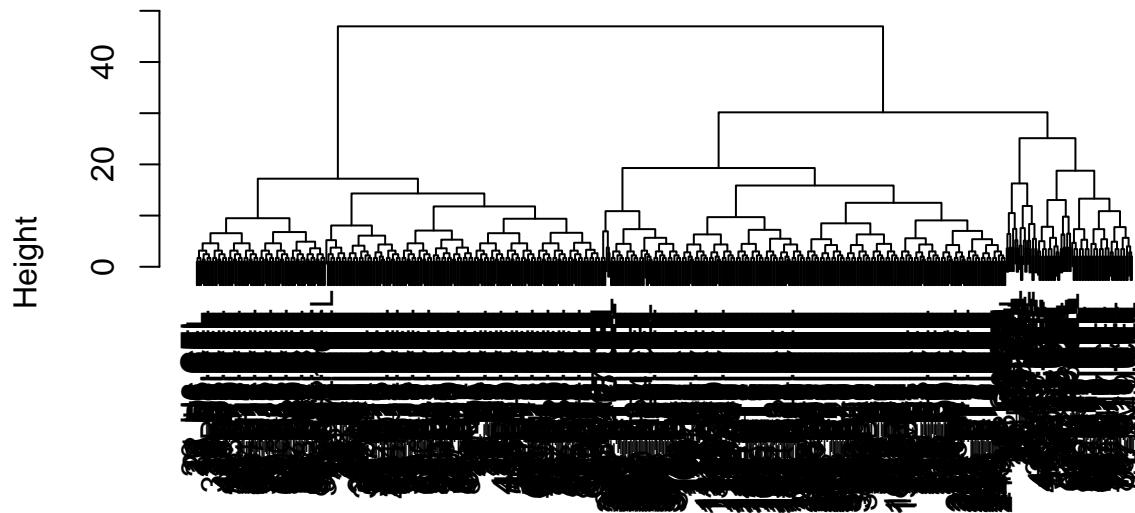


```
# Calcul de la matrice de similarité
similarity_matrix <- as.matrix(similarity_matrix)

# Calcul de la distance euclidienne
distance <- dist(1 - similarity_matrix)

# Création du dendrogramme
dendrogram <- hclust(distance)
plot(dendrogram, main = "Cluster Dendrogram - IBD Similarity")
```

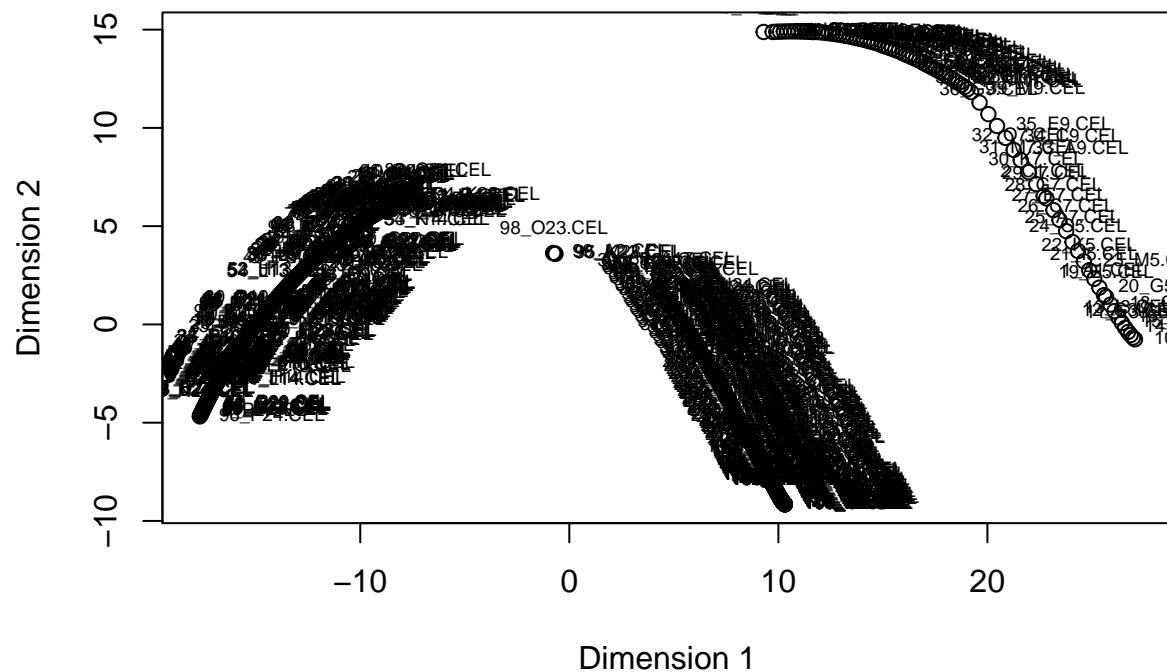
## Cluster Dendrogram – IBD Similarity



```
distance  
hclust (*, "complete")
```

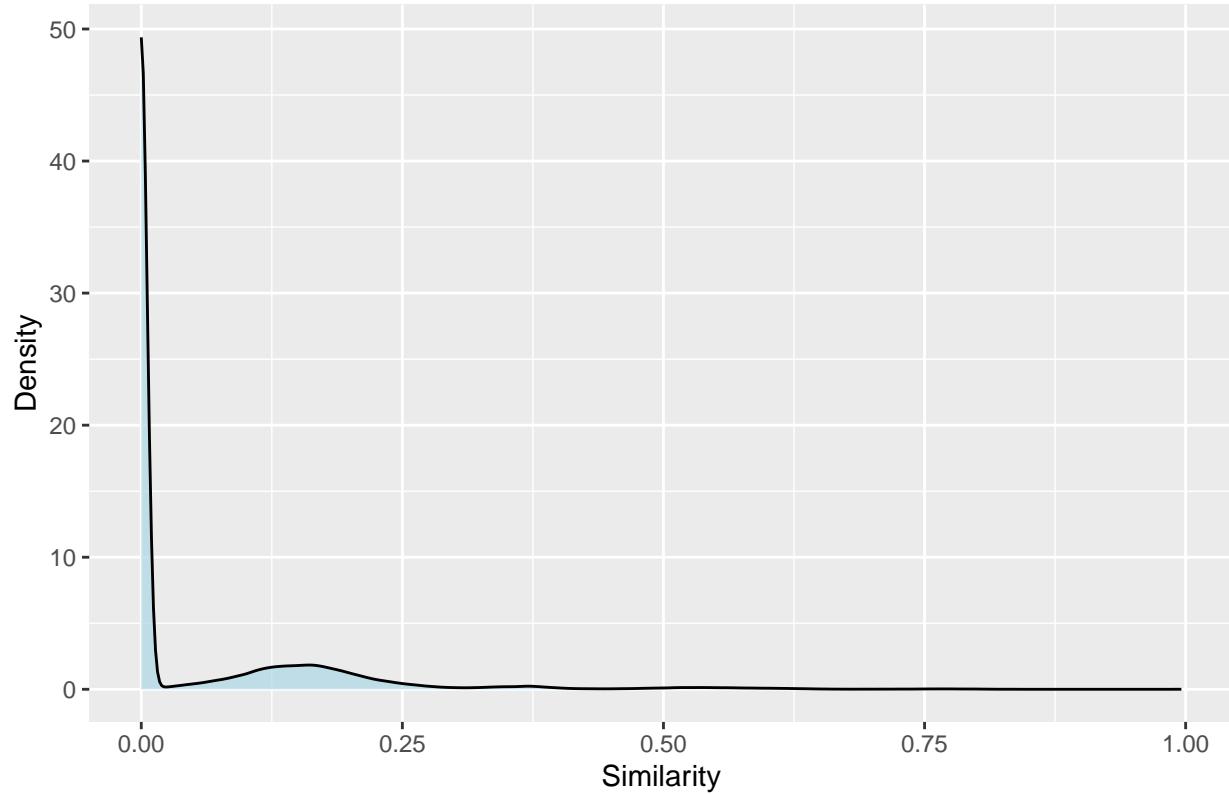
```
#library(MASS)  
# MDS  
mds <- cmdscale(distance)  
  
# Cr ation du plot MDS avec noms d'individus  
plot(mds, xlab = "Dimension 1", ylab = "Dimension 2", main = "MDS Plot - IBD Similarity")  
text(mds, labels = rownames(similarity_matrix), pos = c(3, 4), col = "black", cex = 0.6)
```

## MDS Plot – IBD Similarity



```
# Distribution de la similarité
ggplot(beemuse_genome, aes(x = PI_HAT)) +
  geom_density(fill = "lightblue", alpha = 0.7) +
  labs(title = "Distribution Plot – IBD Similarity", x = "Similarity", y = "Density")
```

## Distribution Plot – IBD Similarity



```
# Définir la palette de couleurs pastel
pastel_colors <- c(
  "#FF0000", "#FF3300", "#FF6600", "#FF9900", "#FFCC00",
  "#FFFF00", "#CCFF00", "#99FF00", "#66FF00", "#33FF00",
  "#00FF00", "#00FF33", "#00FF66", "#00FF99", "#00FFCC",
  "#00FFF", "#00CCFF", "#0099FF", "#0066FF", "#0033FF",
  "#0000FF", "#3300FF", "#6600FF", "#9900FF", "#CC00FF",
  "#FF00FF", "#FF00CC", "#FF0099", "#FF0066", "#FF0033"
)

# Création d'un facteur pour distinguer les différentes données
beemuse_BAH_20_19_genome$Dataset <- "BAH_20-19"
beemuse_BBS_6_19_genome$Dataset <- "BBS_6-19"
beemuse_BER_11_19_genome$Dataset <- "BER_11-19"
beemuse_BH_44_genome$Dataset <- "BH_44"
beemuse_BH_7_19_genome$Dataset <- "BH_7-19"
beemuse_BHA_2_20_genome$Dataset <- "BHA_2-20"
beemuse_BLS_53_19_genome$Dataset <- "BLS_53-19"
beemuse_ER_13_19_genome$Dataset <- "ER_13-19"
beemuse_KBJ_1_19_genome$Dataset <- "KBJ_1-19"
beemuse_KBru_6_20_genome$Dataset <- "KBru_6-20"
beemuse_KLoc_37_19_genome$Dataset <- "KLoc_37-19"
beemuse_KLSU_14_19_genome$Dataset <- "KLSU_14-19"
```

```

beemuse_MM_31_20_genome$Dataset <- "MM_31-20"
beemuse_MM_37_20_genome$Dataset <- "MM_37-20"
beemuse_MP_10_20_genome$Dataset <- "MP_10-20"
beemuse_PersoBC_2021_genome$Dataset <- "PersoBC_2021"
beemuse_PersoJLL_2021_genome$Dataset <- "PersoJLL_2021"
beemuse_PersoJLL_2022_genome$Dataset <- "PersoJLL_2022"
beemuse_PersoLD_2021_genome$Dataset <- "PersoLD_2021"
beemuse_PersoLD_2022_genome$Dataset <- "PersoLD_2022"
beemuse_PersoUB_2021_genome$Dataset <- "PersoUB_2021"
beemuse_PersoUB_2022_genome$Dataset <- "PersoUB_2022"
beemuse_S_GZ_2_19_genome$Dataset <- "S_GZ_2-19"
beemuse_SBJ_3_19_genome$Dataset <- "SBJ_3-19"
beemuse_SJ_16_20_genome$Dataset <- "SJ_16-20"
beemuse_SJ_24_20_genome$Dataset <- "SJ_24-20"
beemuse_SJ_30_20_genome$Dataset <- "SJ_30-20"
beemuse_TL_13_20_genome$Dataset <- "TL_13-20"
beemuse_TL_19_20_genome$Dataset <- "TL_19-20"
beemuse_unknown_genome$Dataset <- "Unknown"

# Combiner les ensembles de données
all_data <- rbind(beemuse_BAH_20_19_genome, beemuse_BBS_6_19_genome, beemuse_BER_11_19_genome, beemuse_BLA_53_19_genome, beemuse_ER_13_19_genome, beemuse_KBJ_1_19_genome, beemuse_KBru_6_20_genome, beemuse_KLoc_37_19_genome, beemuse_KLSU_14_19_genome, beemuse_MM_31_20_genome, beemuse_MM_37_20_genome, beemuse_MP_10_20_genome, beemuse_PersoBC_2021_genome, beemuse_PersoJLL_2021_genome, beemuse_PersoJLL_2022_genome, beemuse_PersoLD_2021_genome, beemuse_PersoLD_2022_genome, beemuse_PersoUB_2021_genome, beemuse_PersoUB_2022_genome, beemuse_S_GZ_2_19_genome, beemuse_SBJ_3_19_genome, beemuse_SJ_16_20_genome, beemuse_SJ_24_20_genome, beemuse_SJ_30_20_genome, beemuse_TL_13_20_genome, beemuse_TL_19_20_genome, beemuse_unknown_genome)

# Define the order of the groups and reverse it
group_order <- c(
  "BAH_20-19", "BBS_6-19", "BER_11-19", "BH_44", "BH_7-19", "BHA_2-20",
  "BLS_53-19", "ER_13-19", "KBJ_1-19", "KBru_6-20", "KLoc_37-19",
  "KLSU_14-19", "MM_31-20", "MM_37-20", "MP_10-20", "PersoBC_2021",
  "PersoJLL_2021", "PersoJLL_2022", "PersoLD_2021", "PersoLD_2022",
  "PersoUB_2021", "PersoUB_2022", "S_GZ_2-19", "SBJ_3-19", "SJ_16-20",
  "SJ_24-20", "SJ_30-20", "TL_13-20", "TL_19-20", "Unknown"
)
group_order <- rev(group_order)

# Convert the Dataset variable to a factor with the reversed order
all_data$Dataset <- factor(all_data$Dataset, levels = group_order)

# Créer le graphique de violon pour IBD
ggplot(all_data, aes(x = PI_HAT, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.3) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width = 0.5)) +
  scale_fill_manual(values = pastel.colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBD",
       x = "IBD", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

```

## Analyse IBD - ID\_2a - 29 groupes + Unknown (pedigree inconnue)

```

## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

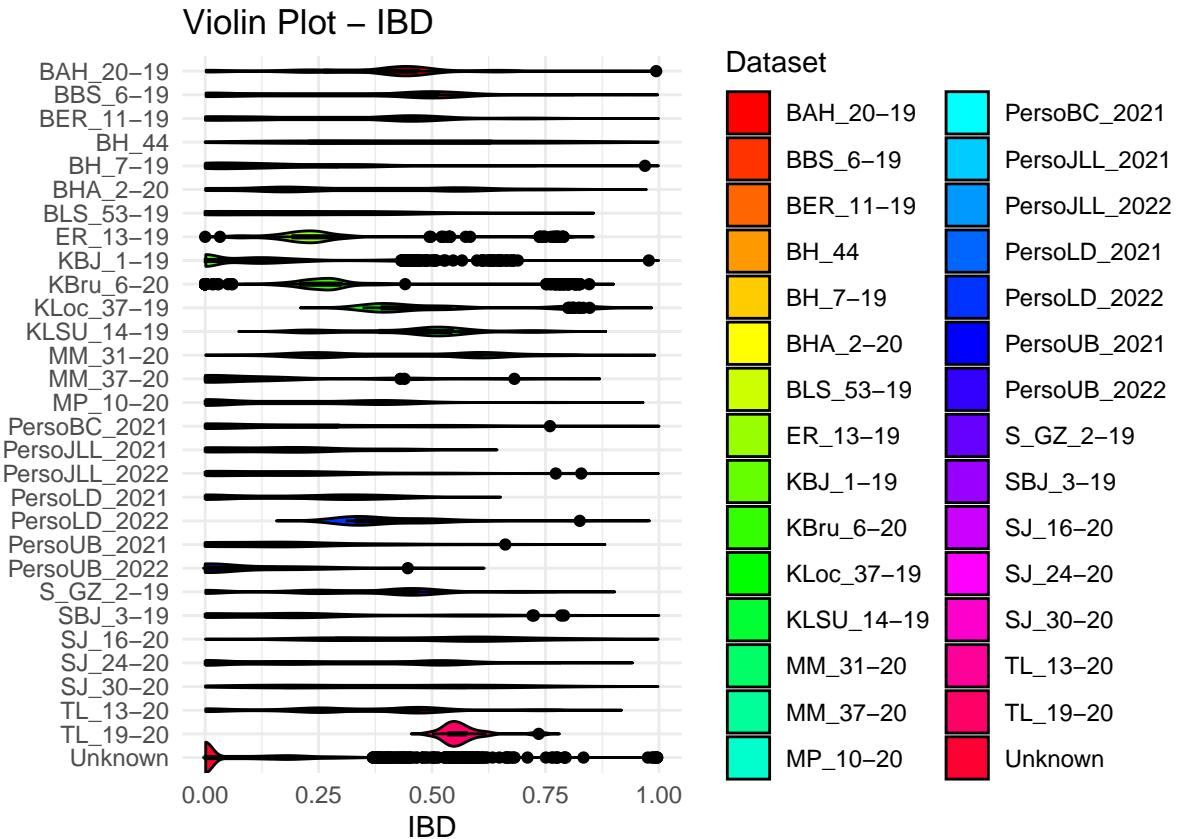
```

```

## Warning: 'position_dodge()' requires non-overlapping x intervals

## Warning: Removed 2358 rows containing missing values ('geom_violin()').

```



```

# Create the violin plot for IBS
ggplot(all_data, aes(x = DST, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.2) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBS",
      x = "IBS", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

```

```

## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

```

```

## Warning: 'position_dodge()' requires non-overlapping x intervals

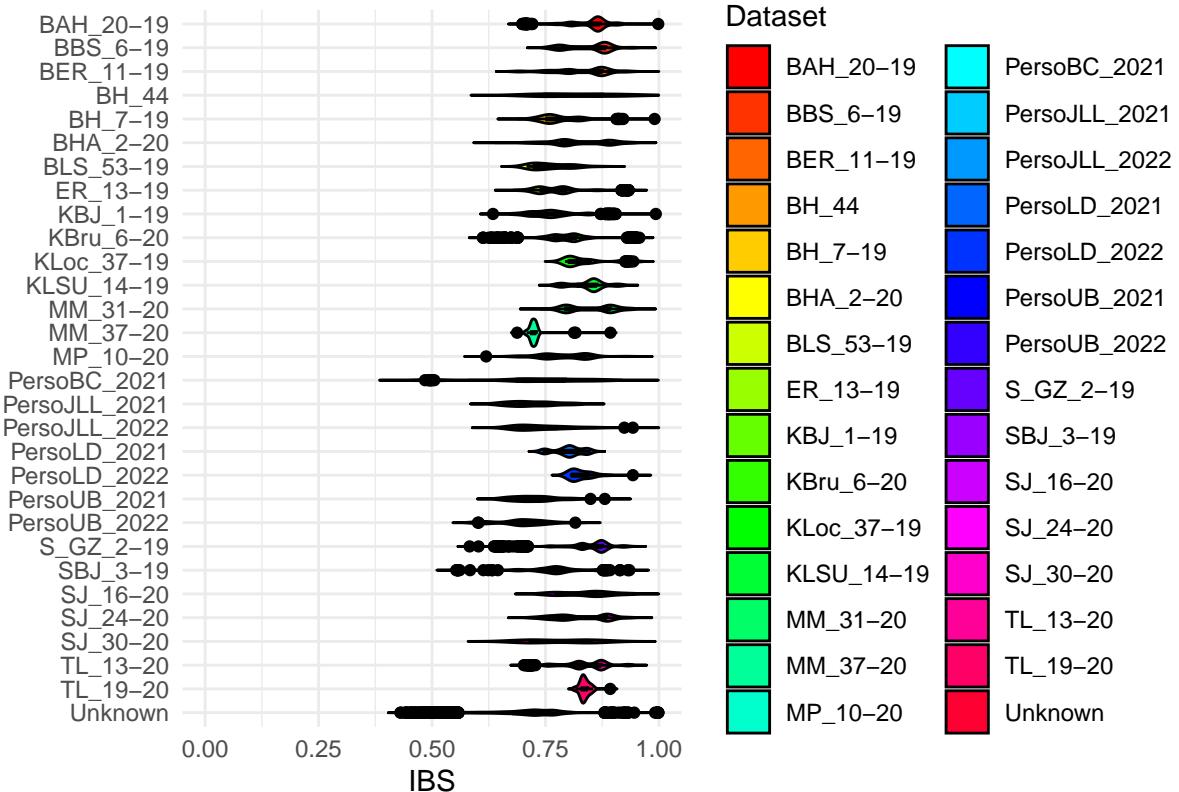
```

```

## Warning: Removed 313 rows containing missing values ('geom_violin()').

```

## Violin Plot – IBS



```

# x15 - 1
# Définir la palette de couleurs pastel
pastel_colors <- c(
  "#FF0000", "#FF3300", "#FF6600", "#FF9900", "#FFCC00",
  "#FFFF00", "#CCFF00", "#99FF00", "#66FF00", "#33FF00",
  "#00FF00", "#00FF33", "#00FF66", "#00FF99", "#00FFCC")

# Création d'un facteur pour distinguer les différentes données
beemuse_BAH_20_19_genome$Dataset <- "BAH_20-19"
beemuse_BBS_6_19_genome$Dataset <- "BBS_6-19"
beemuse_BER_11_19_genome$Dataset <- "BER_11-19"
beemuse_BH_44_genome$Dataset <- "BH_44"
beemuse_BH_7_19_genome$Dataset <- "BH_7-19"
beemuse_BHA_2_20_genome$Dataset <- "BHA_2-20"
beemuse_BLS_53_19_genome$Dataset <- "BLS_53-19"
beemuse_ER_13_19_genome$Dataset <- "ER_13-19"
beemuse_KBJ_1_19_genome$Dataset <- "KBJ_1-19"
beemuse_KBru_6_20_genome$Dataset <- "KBru_6-20"
beemuse_KLoc_37_19_genome$Dataset <- "KLoc_37-19"
beemuse_KLSU_14_19_genome$Dataset <- "KLSU_14-19"
beemuse_MM_31_20_genome$Dataset <- "MM_31-20"
beemuse_MM_37_20_genome$Dataset <- "MM_37-20"
beemuse_MP_10_20_genome$Dataset <- "MP_10-20"

# Combiner les ensembles de données
all_data <- rbind(beemuse_BAH_20_19_genome, beemuse_BBS_6_19_genome, beemuse_BER_11_19_genome, beemuse_

```

```

# Définir l'ordre des groupes
group_order <- c(
  "BAH_20-19", "BBS_6-19", "BER_11-19", "BH_44", "BH_7-19", "BHA_2-20",
  "BLS_53-19", "ER_13-19", "KBJ_1-19", "KBru_6-20", "KLoc_37-19",
  "KLSU_14-19", "MM_31-20", "MM_37-20", "MP_10-20"
)

group_order <- rev(group_order)

# Convertir la variable Dataset en un facteur avec l'ordre spécifié
all_data$Dataset <- factor(all_data$Dataset, levels = group_order)

# Créer le graphique de violon pour IBS
ggplot(all_data, aes(x = DST, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.3) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBS",
      x = "IBS", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

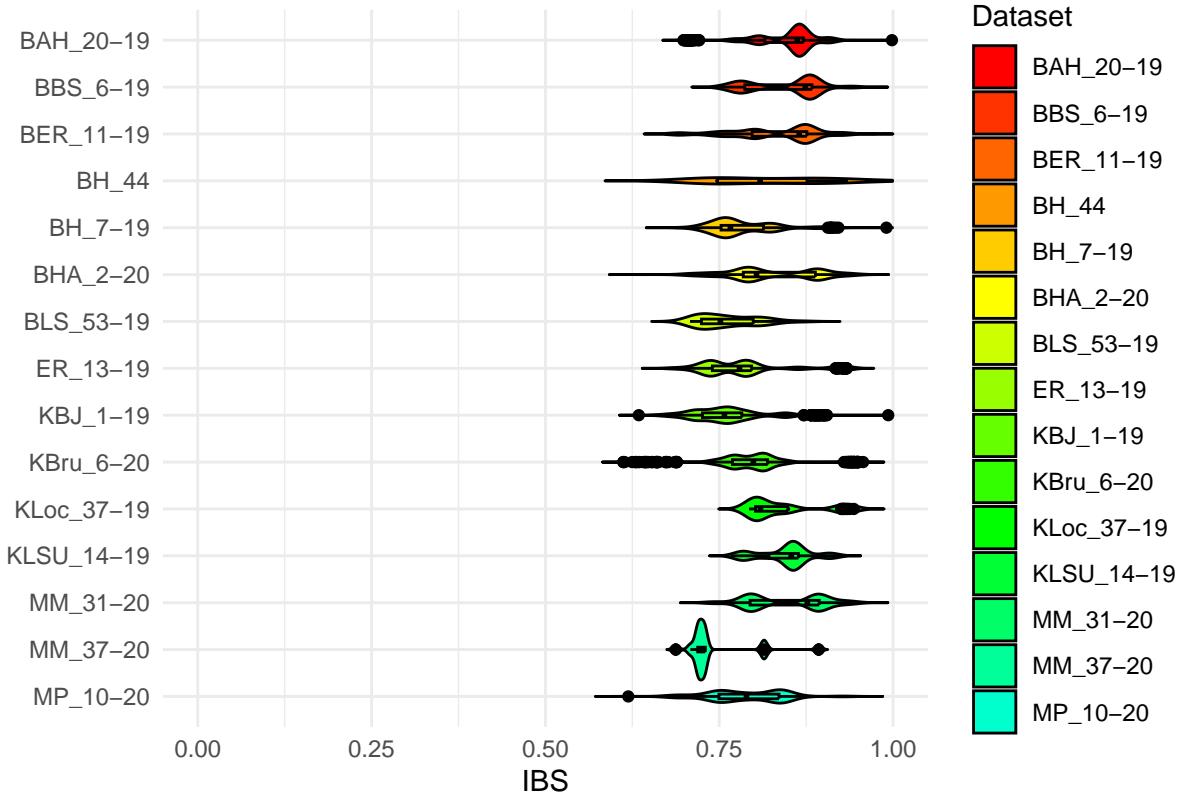
## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

## Warning: `position_dodge()` requires non-overlapping x intervals

## Warning: Removed 232 rows containing missing values (`geom_violin()`).

```

## Violin Plot – IBS



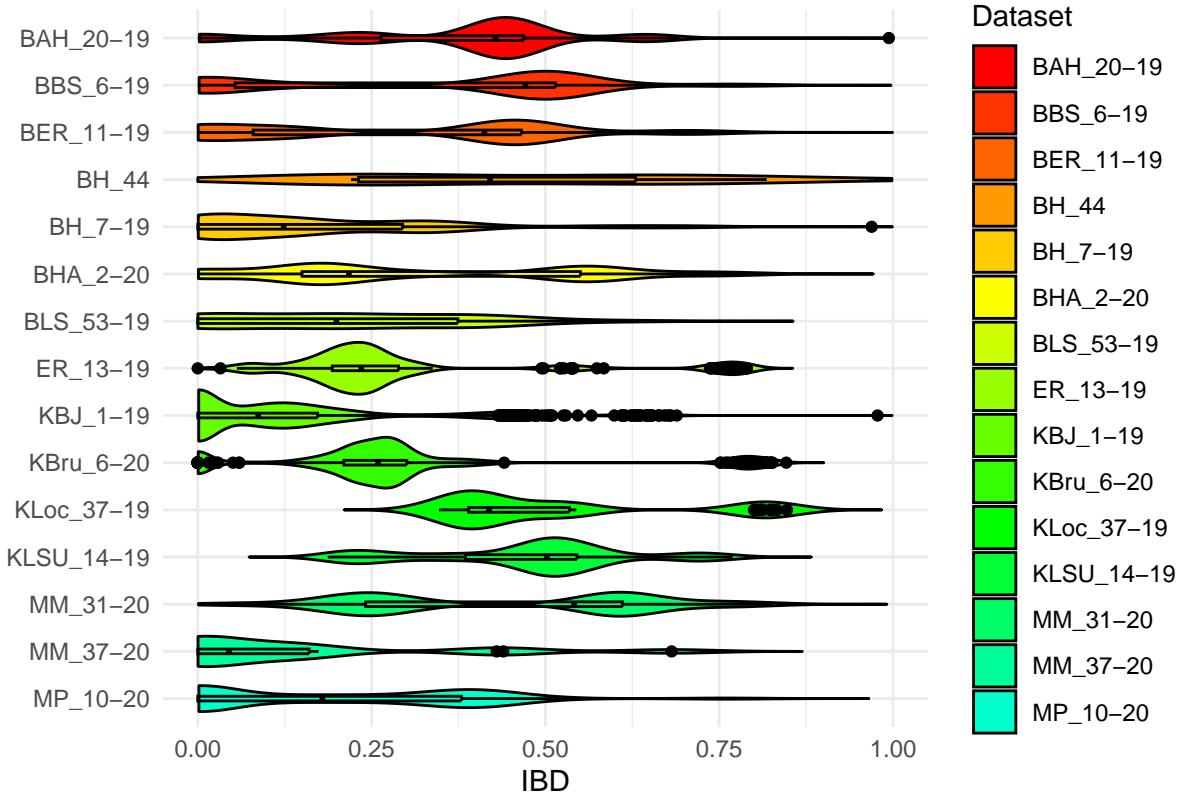
```
# Créer le graphique de violon pour IBD
ggplot(all_data, aes(x = PI_HAT, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.1) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBD",
      x = "IBD", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

## Warning: 'position_dodge()' requires non-overlapping x intervals

## Warning: Removed 1143 rows containing missing values ('geom_violin()').
```

## Violin Plot – IBD



```

# 15 autres groupes
# Définir la palette de couleurs pastel
pastel_colors <- c(
  "#00FFFF", "#00CCFF", "#0099FF", "#0066FF", "#0033FF",
  "#0000FF", "#3300FF", "#6600FF", "#9900FF", "#CC00FF",
  "#FF00FF", "#FF00CC", "#FF0099", "#FF0066", "#FF0033")

# Création d'un facteur pour distinguer les différentes données
beemuse_PersoBC_2021_genome$Dataset <- "PersoBC_2021"
beemuse_PersoJLL_2021_genome$Dataset <- "PersoJLL_2021"
beemuse_PersoJLL_2022_genome$Dataset <- "PersoJLL_2022"
beemuse_PersoLD_2021_genome$Dataset <- "PersoLD_2021"
beemuse_PersoLD_2022_genome$Dataset <- "PersoLD_2022"
beemuse_PersoUB_2021_genome$Dataset <- "PersoUB_2021"
beemuse_PersoUB_2022_genome$Dataset <- "PersoUB_2022"
beemuse_S_GZ_2_19_genome$Dataset <- "S_GZ_2-19"
beemuse_SBJ_3_19_genome$Dataset <- "SBJ_3-19"
beemuse_SJ_16_20_genome$Dataset <- "SJ_16-20"
beemuse_SJ_24_20_genome$Dataset <- "SJ_24-20"
beemuse_SJ_30_20_genome$Dataset <- "SJ_30-20"
beemuse_TL_13_20_genome$Dataset <- "TL_13-20"
beemuse_TL_19_20_genome$Dataset <- "TL_19-20"
beemuse_unknown_genome$Dataset <- "Unknown"

# Combiner les ensembles de données
all_data <- rbind(beemuse_PersoBC_2021_genome, beemuse_PersoJLL_2021_genome, beemuse_PersoJLL_2022_genome)
  
```

```

# Définir l'ordre des groupes
group_order <- c("PersoBC_2021",
  "PersoJLL_2021", "PersoJLL_2022", "PersoLD_2021", "PersoLD_2022",
  "PersoUB_2021", "PersoUB_2022", "S_GZ_2-19", "SBJ_3-19", "SJ_16-20",
  "SJ_24-20", "SJ_30-20", "TL_13-20", "TL_19-20", "Unknown"
)

group_order <- rev(group_order)

# Convertir la variable Dataset en un facteur avec l'ordre spécifié
all_data$Dataset <- factor(all_data$Dataset, levels = group_order)

# Créer le graphique de violon pour IBS
ggplot(all_data, aes(x = DST, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.4) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBS",
      x = "IBS", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

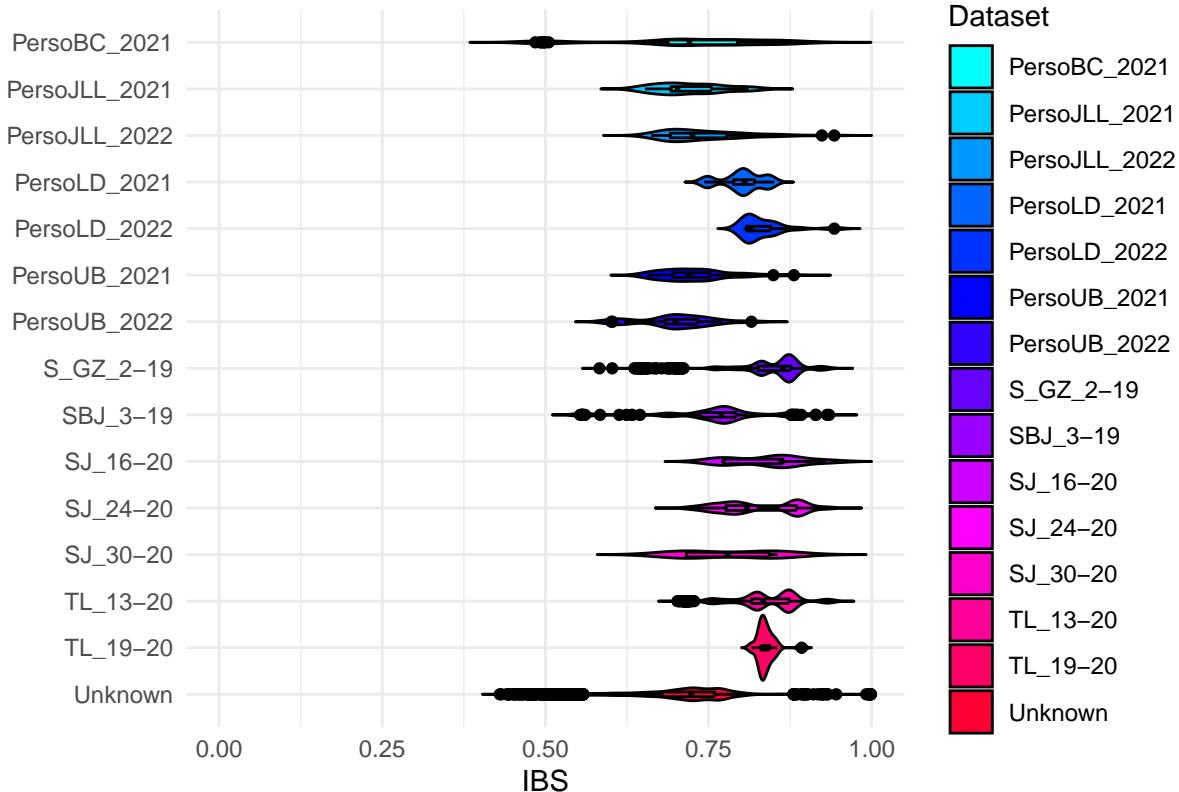
## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

## Warning: `position_dodge()` requires non-overlapping x intervals

## Warning: Removed 81 rows containing missing values (`geom_violin()`).

```

## Violin Plot – IBS



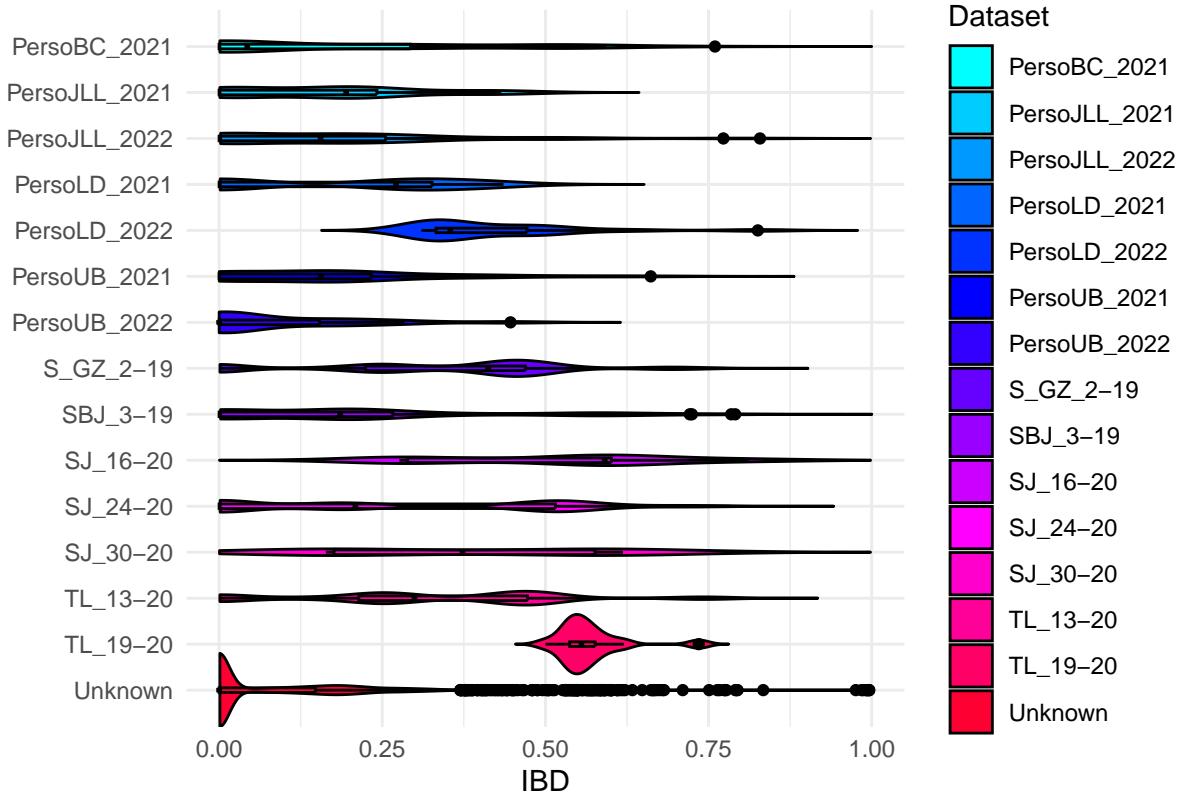
```
# Créer le graphique de violon pour IBD
ggplot(all_data, aes(x = PI_HAT, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.6) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBD",
      x = "IBD", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))
```

```
## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.
```

```
## Warning: ‘position_dodge()’ requires non-overlapping x intervals
```

```
## Warning: Removed 1215 rows containing missing values ('geom_violin()').
```

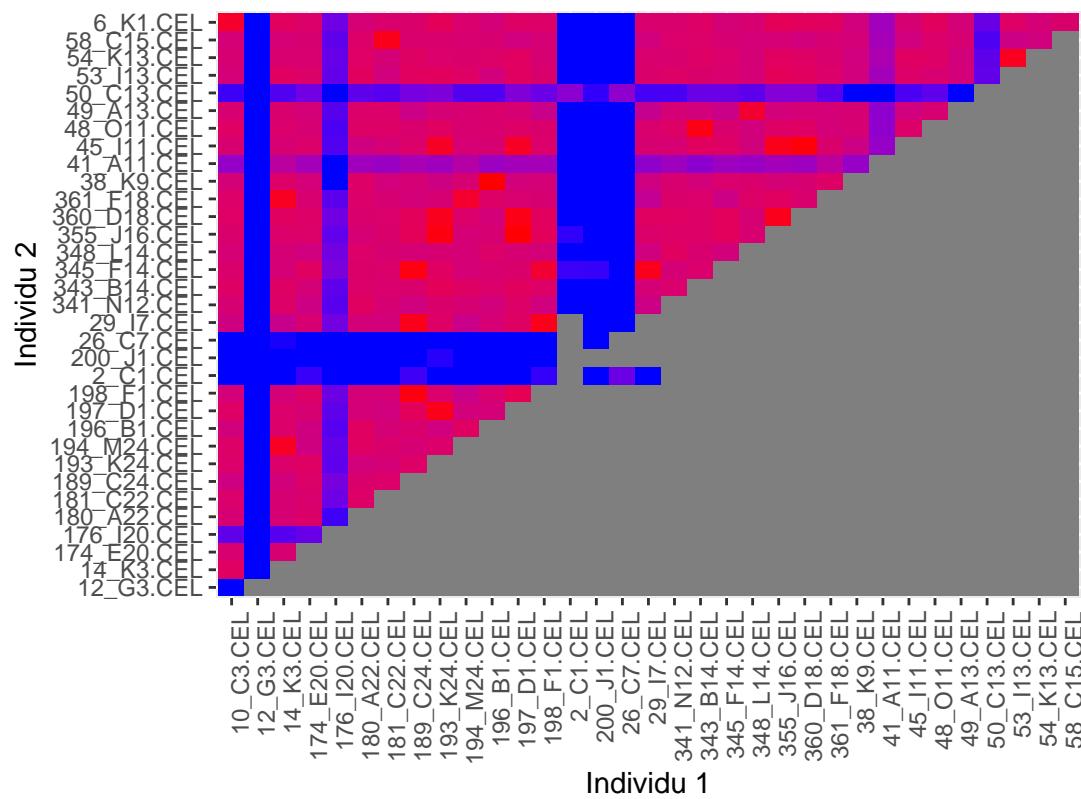
### Violin Plot – IBD



```
similarity_matrix2 <- acast(beemuse_BBS_6_19_genome, IID1 ~ IID2, value.var = "PI_HAT")

# Tracer la heatmap
ggplot(melt(similarity_matrix2), aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap - IBD - BeeMuSe",
       x = "Individu 1",
       y = "Individu 2") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Heatmap – IBD – BeeMuSe



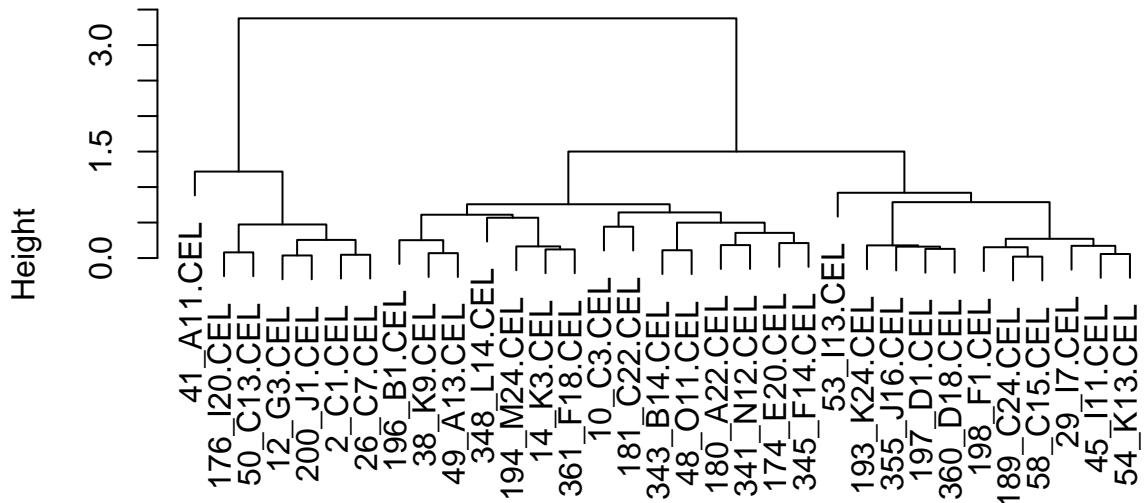
### Analyse IBD - BBS\_6-19

```
# Calcul de la matrice de similarité
similarity_matrix <- as.matrix(similarity_matrix2)

# Calcul de la distance euclidienne
distance <- dist(1 - similarity_matrix)

# Création du dendrogramme
dendrogram <- hclust(distance)
plot(dendrogram, main = "Cluster Dendrogram - IBD Similarity")
```

## Cluster Dendrogram – IBD Similarity

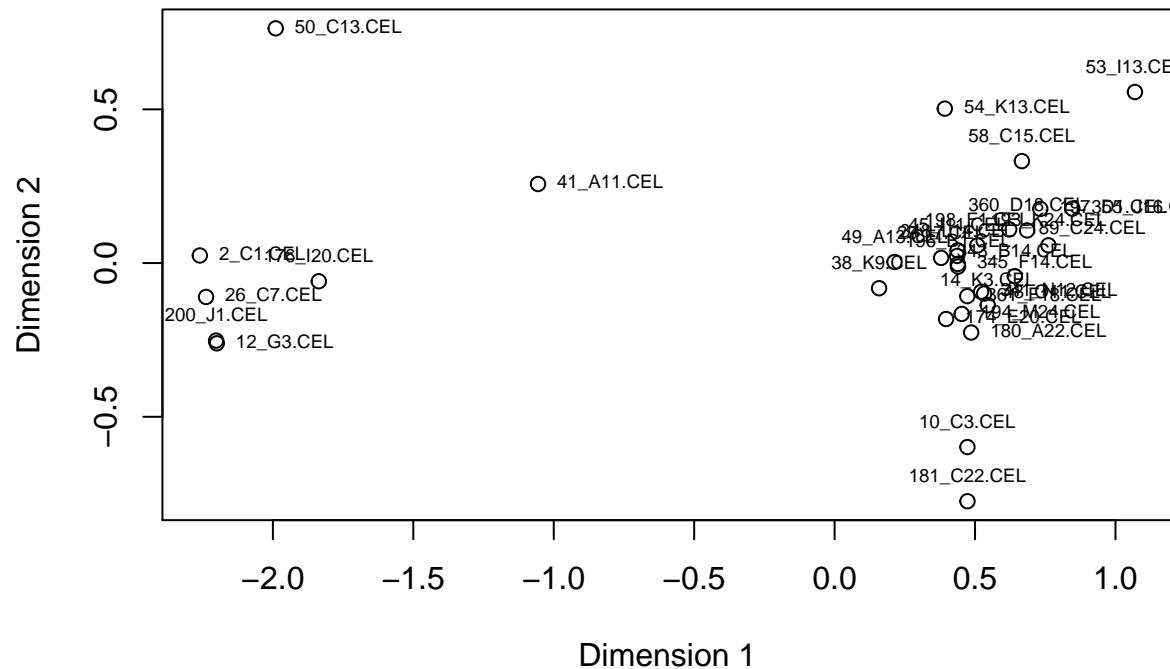


```
distance
hclust (*, "complete")
```

```
#library(MASS)
# MDS
mds <- cmdscale(distance)

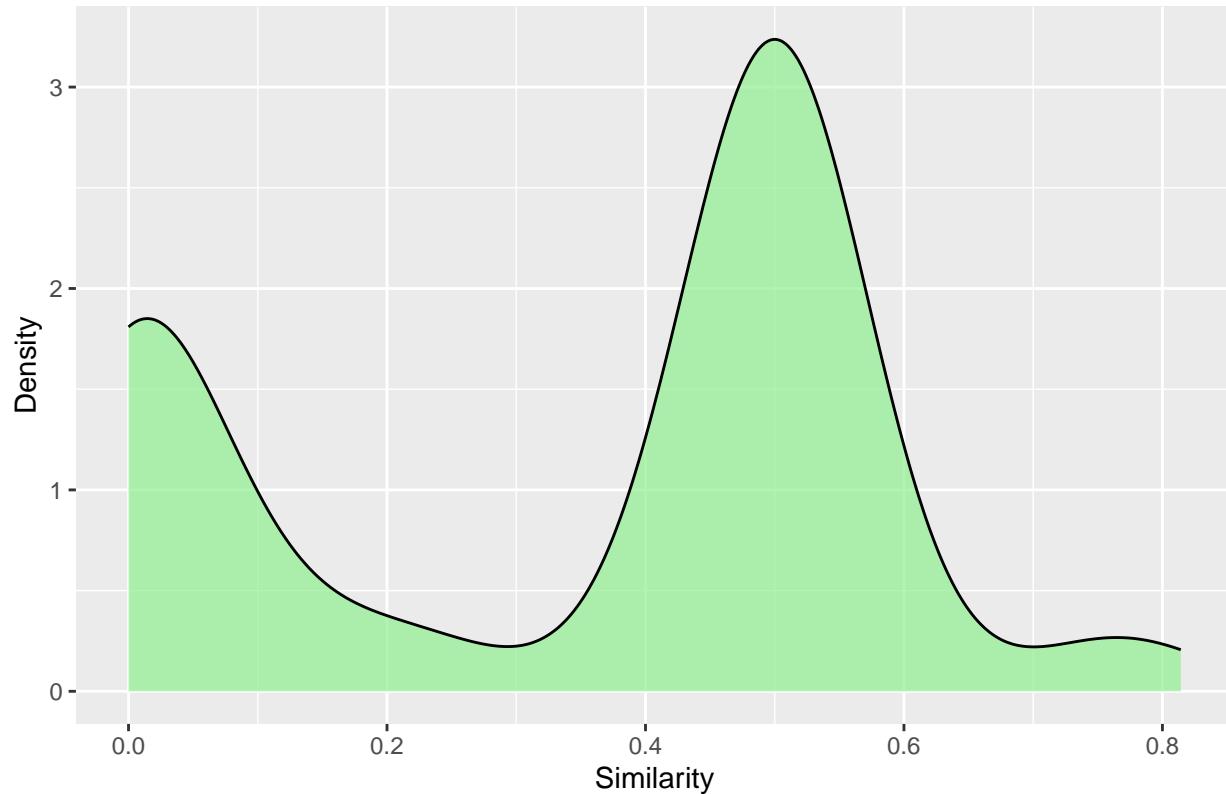
# Création du plot MDS avec noms d'individus
plot(mds, xlab = "Dimension 1", ylab = "Dimension 2", main = "MDS Plot - IBD Similarity")
text(mds, labels = rownames(similarity_matrix), pos = c(3, 4), col = "black", cex = 0.6)
```

## MDS Plot – IBD Similarity



```
# Distribution de la similarité
ggplot(beemuse_BBS_6_19_genome, aes(x = PI_HAT)) +
  geom_density(fill = "lightgreen", alpha = 0.7) +
  labs(title = "Distribution Plot – IBD Similarity", x = "Similarity", y = "Density")
```

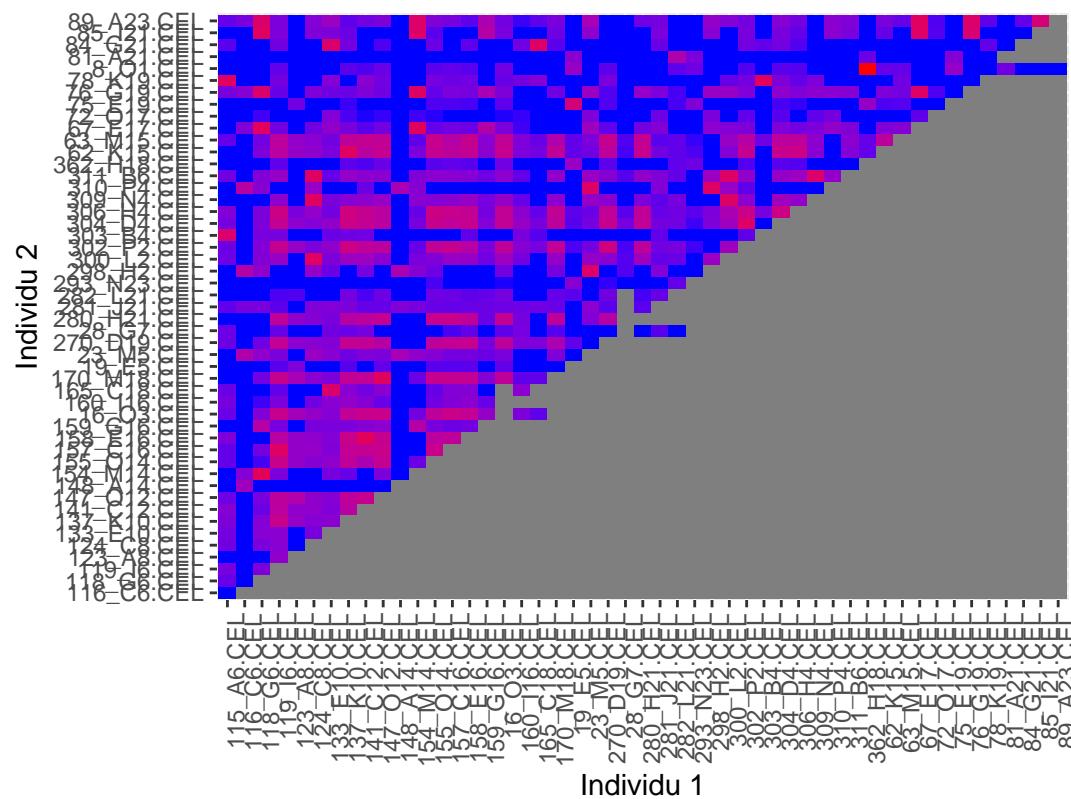
## Distribution Plot – IBD Similarity



```
similarity_matrix3 <- acast(beemuse_KBJ_1_19_genome, IID1 ~ IID2, value.var = "PI_HAT")

# Tracer la heatmap
ggplot(melt(similarity_matrix3), aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap - IBD - BeeMuSe",
       x = "Individu 1",
       y = "Individu 2") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Heatmap – IBD – BeeMuSe



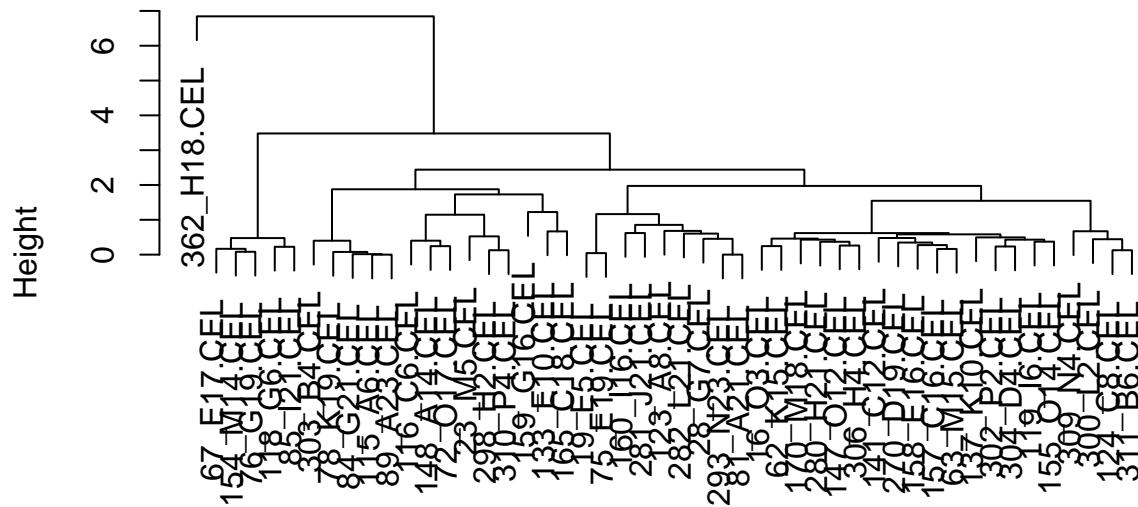
Analyse IBD - KBJ\_1-19

```
# Calcul de la matrice de similarité
similarity_matrix <- as.matrix(similarity_matrix3)

# Calcul de la distance euclidienne
distance <- dist(1 - similarity_matrix)

# Création du dendrogramme
dendrogram <- hclust(distance)
plot(dendrogram, main = "Cluster Dendrogram - IBD Similarity")
```

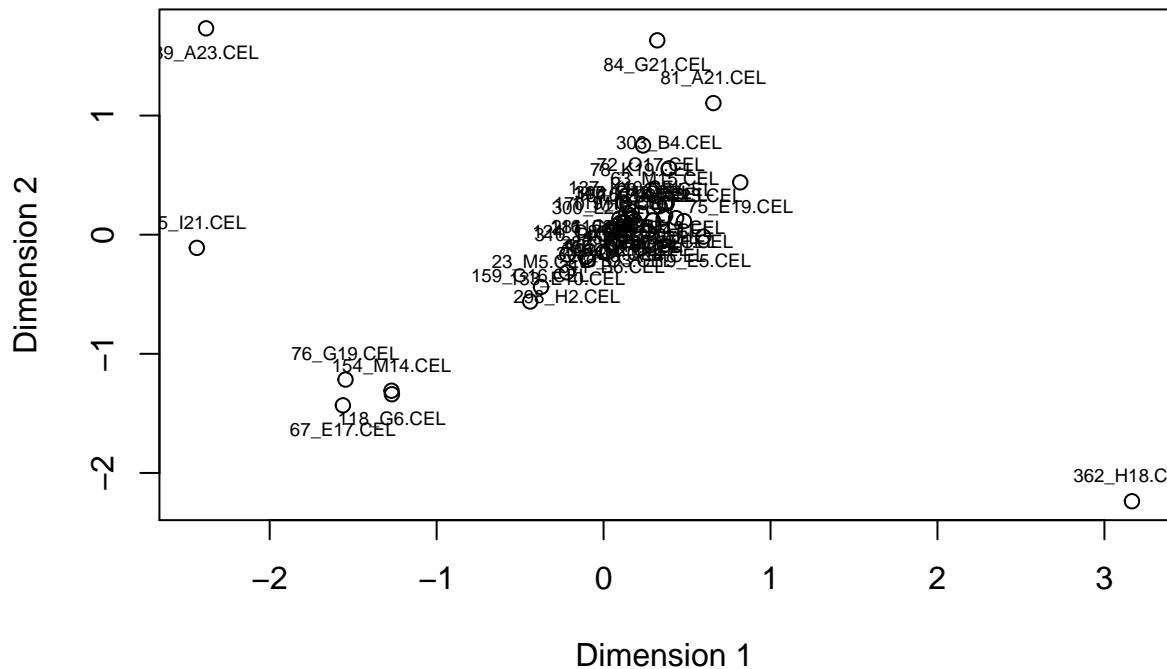
## Cluster Dendrogram – IBD Similarity



```
distance  
hclust (*, "complete")
```

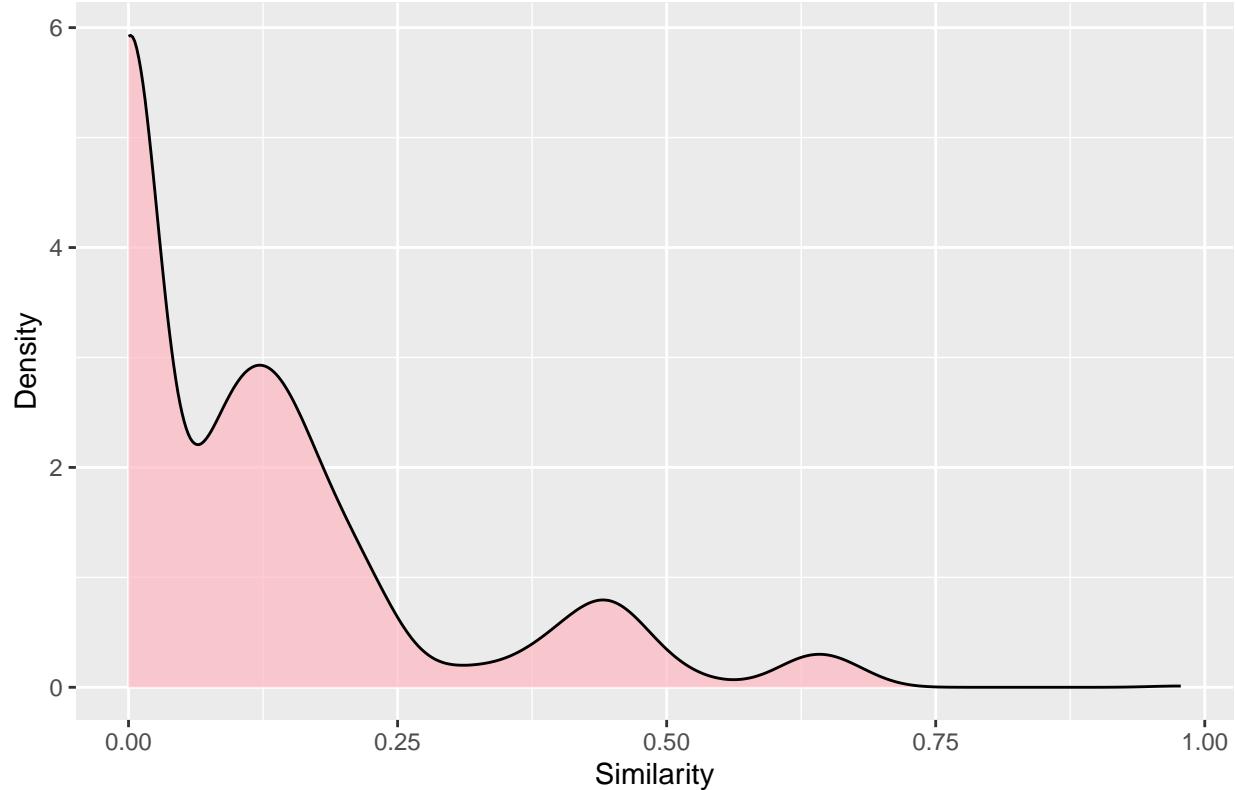
```
# MDS  
mds <- cmdscale(distance)  
  
# Création du plot MDS avec noms d'individus  
plot(mds, xlab = "Dimension 1", ylab = "Dimension 2", main = "MDS Plot – IBD Similarity")  
text(mds, labels = rownames(similarity_matrix), pos = c(1, 3), col = "black", cex = 0.6)
```

## MDS Plot – IBD Similarity



```
# Distribution de la similarité
ggplot(beemuse_KBJ_1_19_genome, aes(x = PI_HAT)) +
  geom_density(fill = "lightpink", alpha = 0.7) +
  labs(title = "Distribution Plot – IBD Similarity", x = "Similarity", y = "Density")
```

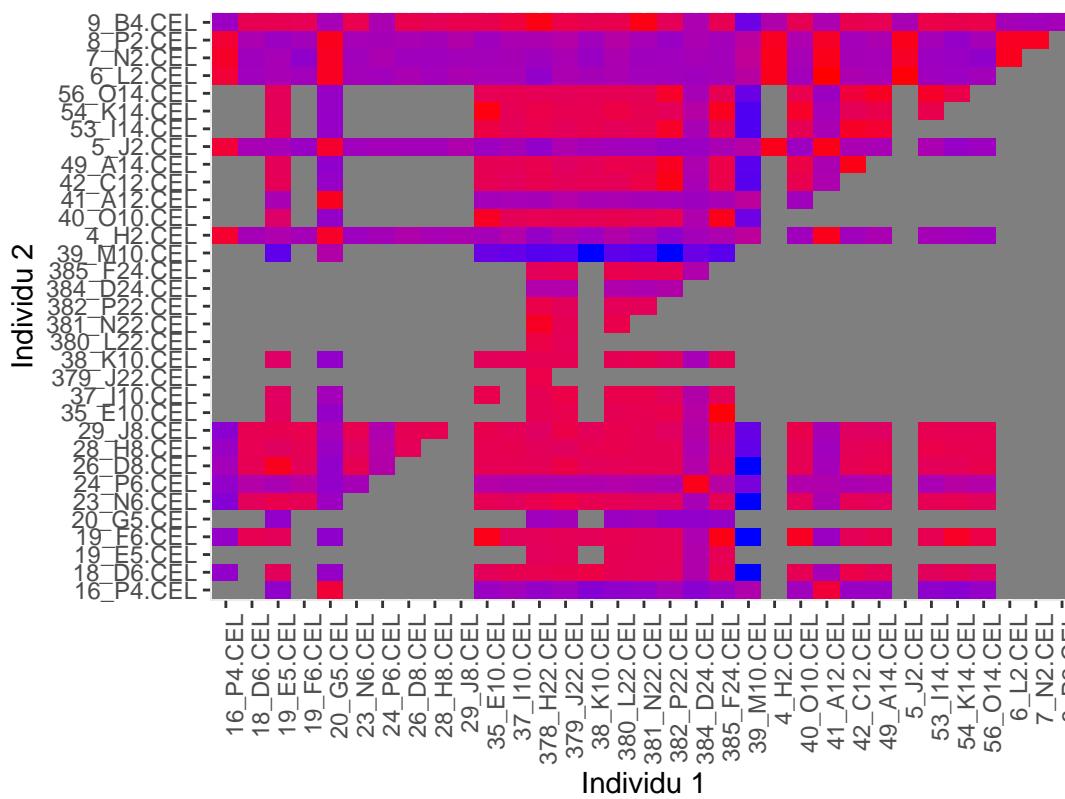
## Distribution Plot – IBD Similarity



```
similarity_matrix2 <- acast(beemuse_MM_31_20_genome, IID1 ~ IID2, value.var = "PI_HAT")

# Tracer la heatmap
ggplot(melt(similarity_matrix2), aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap - IBD - BeeMuSe",
       x = "Individu 1",
       y = "Individu 2") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Heatmap – IBD – BeeMuSe



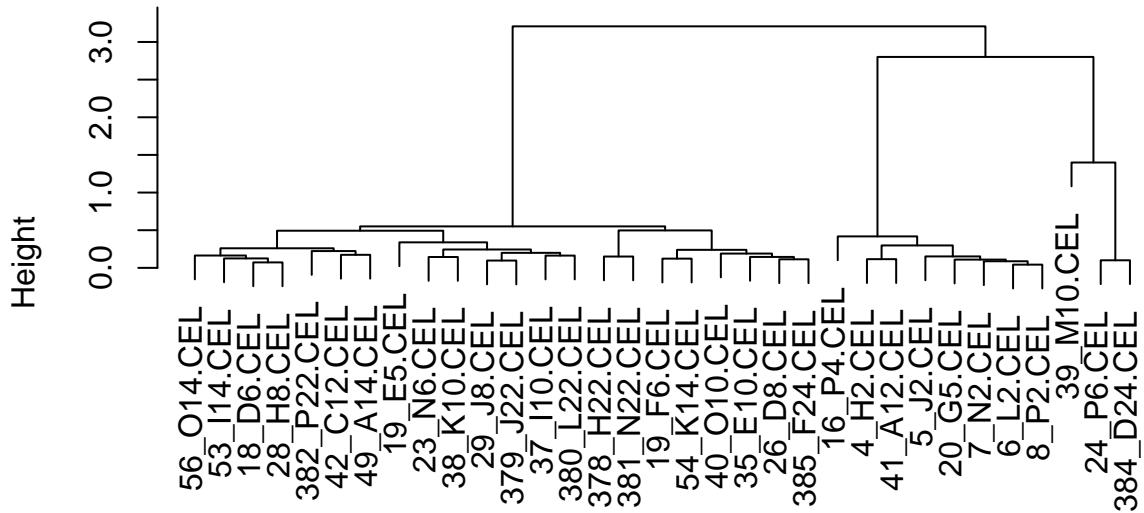
Analyse IBD - MM\_31-20

```
# Calcul de la matrice de similarité
similarity_matrix <- as.matrix(similarity_matrix2)

# Calcul de la distance euclidienne
distance <- dist(1 - similarity_matrix)

# Création du dendrogramme
dendrogram <- hclust(distance)
plot(dendrogram, main = "Cluster Dendrogram - IBD Similarity")
```

## Cluster Dendrogram – IBD Similarity

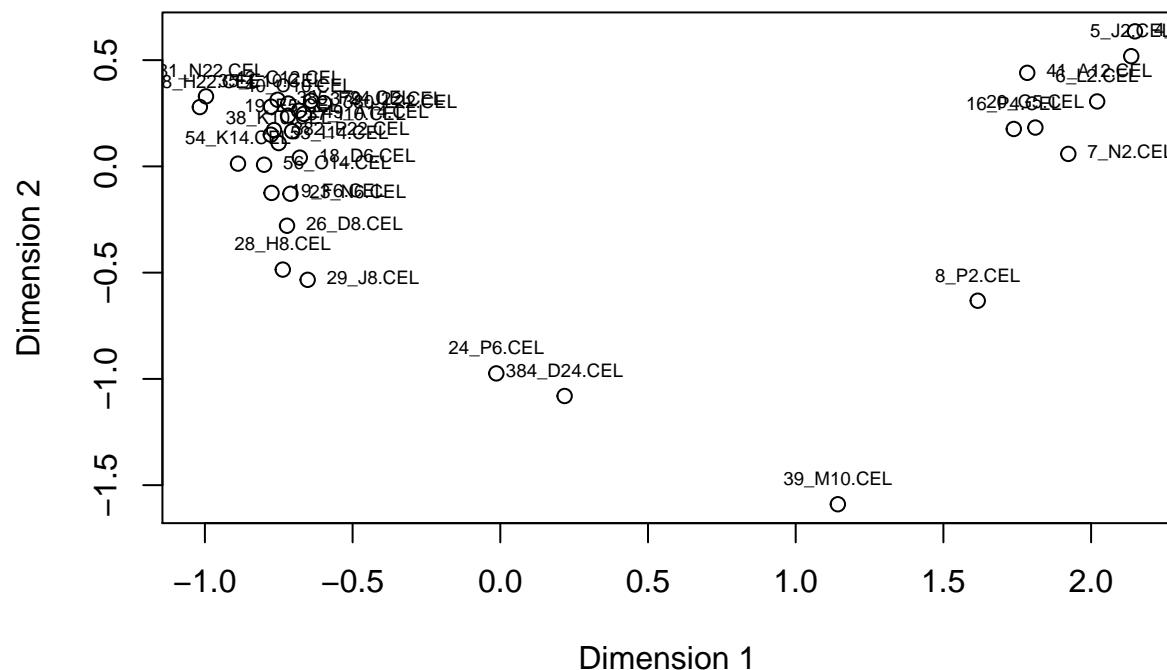


distance  
hclust (\*, "complete")

```
# library(MASS)
# MDS
mds <- cmdscale(distance)

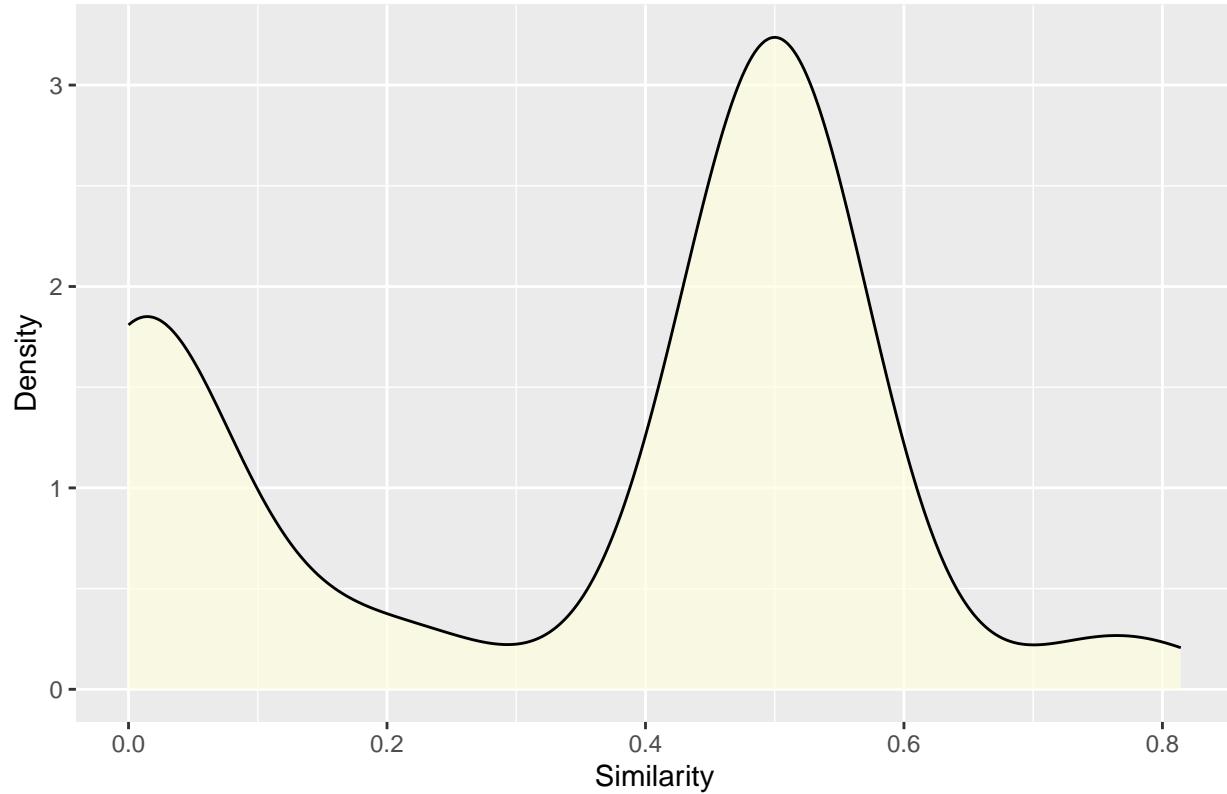
# Création du plot MDS avec noms d'individus
plot(mds, xlab = "Dimension 1", ylab = "Dimension 2", main = "MDS Plot - IBD Similarity")
text(mds, labels = rownames(similarity_matrix), pos = c(3, 4), col = "black", cex = 0.6)
```

## MDS Plot – IBD Similarity



```
# Distribution de la similarité
ggplot(beemuse_BBS_6_19_genome, aes(x = PI_HAT)) +
  geom_density(fill = "lightyellow", alpha = 0.7) +
  labs(title = "Distribution Plot – IBD Similarity", x = "Similarity", y = "Density")
```

## Distribution Plot – IBD Similarity



**“plink –genome”** On extrait les individus de même famille ID\_2a après avoir effectué la commande “plink –genome” pour les 748 individus BeeMuSe. On obtient 30 fichiers ‘genome’ correspondant à chaque famille ID\_2a et les Unknown, contenant les informations sur leur apparentement.

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

beemuse_BAH_20_19_genome <- read.table("BAH_20-19.genome", header=TRUE)
beemuse_BBS_6_19_genome <- read.table("BBS_6-19.genome", header=TRUE)
beemuse_BER_11_19_genome <- read.table("BER_11-19.genome", header=TRUE)
beemuse_BH_44_genome <- read.table("BH_44.genome", header=TRUE)
beemuse_BH_7_19_genome <- read.table("BH_7-19.genome", header=TRUE)
beemuse_BHA_2_20_genome <- read.table("BHA_2-20.genome", header=TRUE)
beemuse_BLS_53_19_genome <- read.table("BLS_53-19.genome", header=TRUE)
beemuse_ER_13_19_genome <- read.table("ER_13-19.genome", header=TRUE)
beemuse_KBJ_1_19_genome <- read.table("KBJ_1-19.genome", header=TRUE)
beemuse_KBru_6_20_genome <- read.table("KBru_6-20.genome", header=TRUE)
beemuse_KLoc_37_19_genome <- read.table("KLoc_37-19.genome", header=TRUE)
beemuse_KLSU_14_19_genome <- read.table("KLSU_14-19.genome", header=TRUE)
beemuse_MM_31_20_genome <- read.table("MM_31-20.genome", header=TRUE)
beemuse_MM_37_20_genome <- read.table("MM_37-20.genome", header=TRUE)
beemuse_MP_10_20_genome <- read.table("MP_10-20.genome", header=TRUE)
beemuse_PersoBC_2021_genome <- read.table("PersoBC_2021.genome", header=TRUE)
beemuse_PersoJLL_2021_genome <- read.table("PersoJLL_2021.genome", header=TRUE)
beemuse_PersoJLL_2022_genome <- read.table("PersoJLL_2022.genome", header=TRUE)
beemuse_PersoLD_2021_genome <- read.table("PersoLD_2021.genome", header=TRUE)
```

```

beemuse_PersoLD_2022_genome <- read.table("PersoLD_2022.genome", header=TRUE)
beemuse_PersoUB_2021_genome <- read.table("PersoUB_2021.genome", header=TRUE)
beemuse_PersoUB_2022_genome <- read.table("PersoUB_2022.genome", header=TRUE)
beemuse_S_GZ_2_19_genome <- read.table("S_GZ_2-19.genome", header=TRUE)
beemuse_SBJ_3_19_genome <- read.table("SBJ_3-19.genome", header=TRUE)
beemuse_SJ_16_20_genome <- read.table("SJ_16-20.genome", header=TRUE)
beemuse_SJ_24_20_genome <- read.table("SJ_24-20.genome", header=TRUE)
beemuse_SJ_30_20_genome <- read.table("SJ_30-20.genome", header=TRUE)
beemuse_TL_13_20_genome <- read.table("TL_13-20.genome", header=TRUE)
beemuse_TL_19_20_genome <- read.table("TL_19-20.genome", header=TRUE)
beemuse_unknown_genome <- read.table("Unknown.genome", header=TRUE)

```

```
similarity_matrix <- acast(beemuse_genome, IID1 ~ IID2, value.var = "PI_HAT")
```

### Analyse IBD entre les 748 échantillons

```
## Aggregation function missing: defaulting to length
```

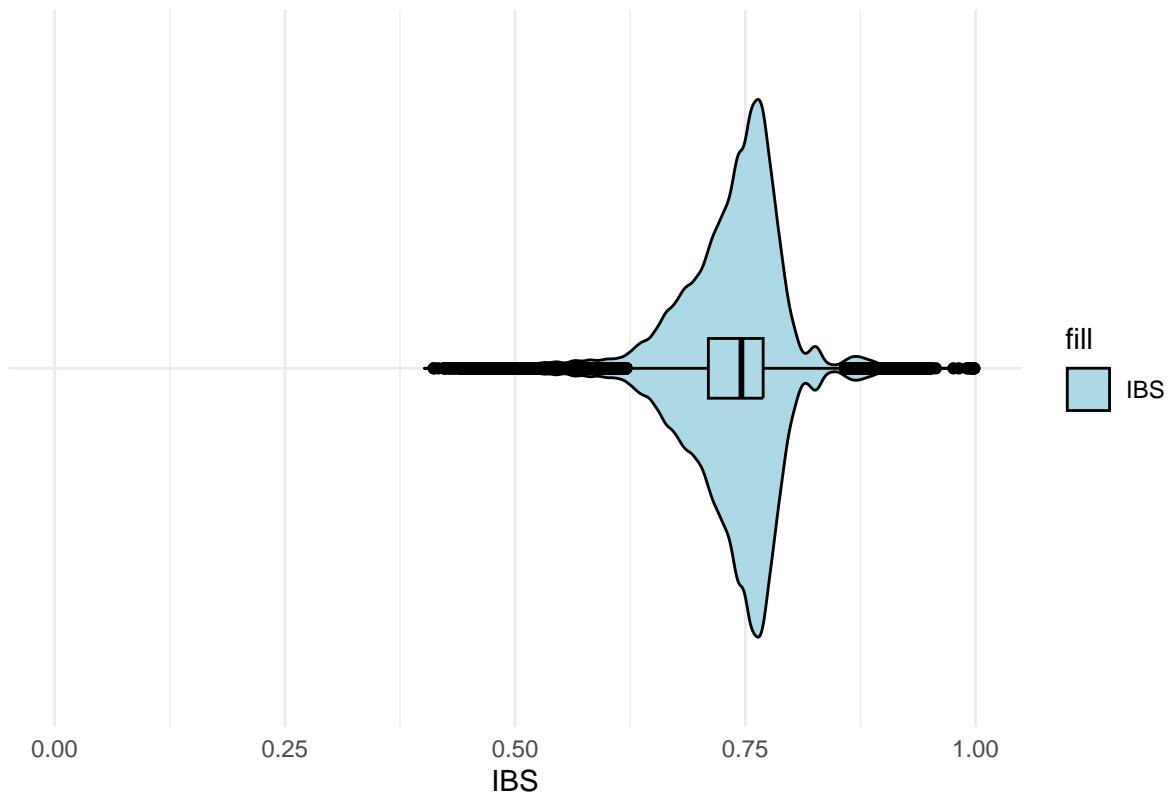
```

ggplot(beemuse_genome, aes(x = "", y = DST, fill = "IBS")) +
  geom_violin(trim = FALSE, color = "black") +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = "lightblue") +
  labs(title = "Violin Plot - IBS",
       x = "", y = "IBS") +
  theme_minimal() +
  scale_y_continuous(limits = c(0, 1)) +
  coord_flip()

```

```
## Warning: Removed 8 rows containing missing values ('geom_violin()').
```

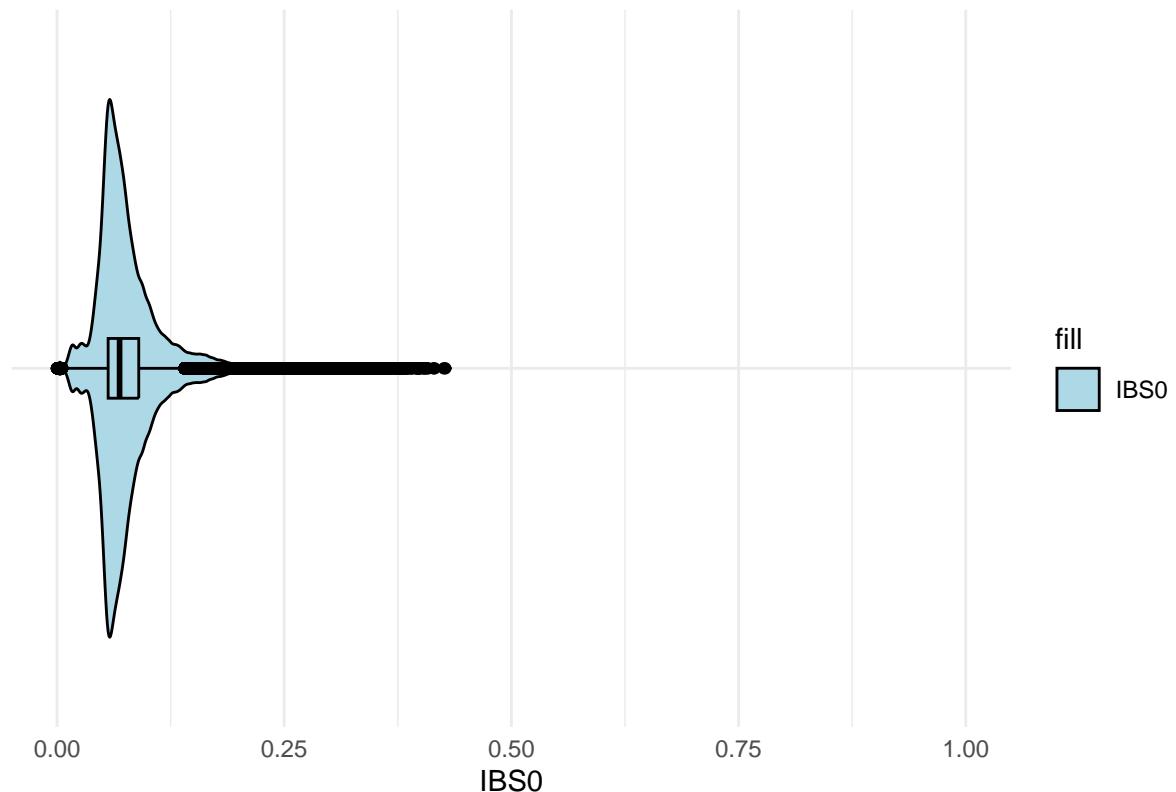
## Violin Plot – IBS



```
ggplot(beemuse_genome, aes(x = "", y = IBS0/(IBS0+IBS1+IBS2), fill = "IBS0")) +  
  geom_violin(trim = FALSE, color = "black") +  
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =  
    scale_fill_manual(values = "lightblue") +  
  labs(title = "Violin Plot - IBS0",  
    x = "", y = "IBS0") +  
  theme_minimal() +  
  scale_y_continuous(limits = c(0, 1)) +  
  coord_flip()
```

## Warning: Removed 7 rows containing missing values ('geom\_violin()').

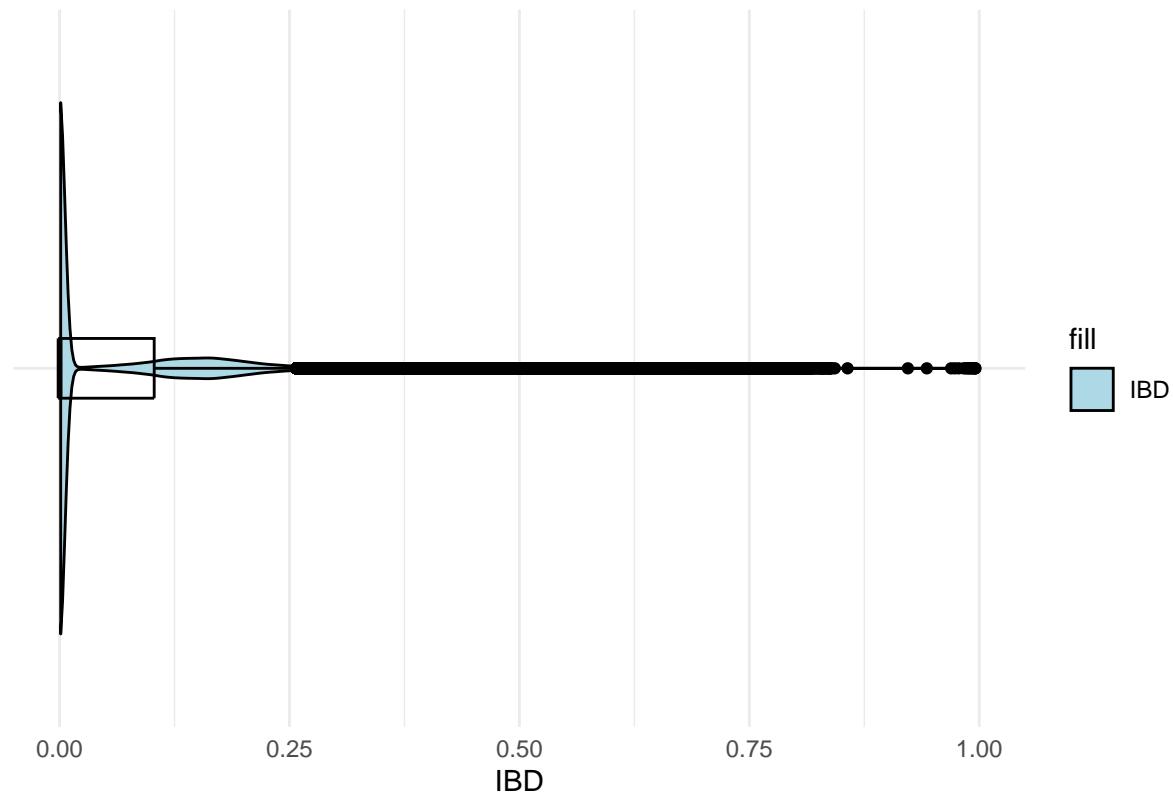
## Violin Plot – IBS0



```
ggplot(beemuse_genome, aes(x = "", y = PI_HAT, fill = "IBD")) +  
  geom_violin(trim = FALSE, color = "black") +  
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =  
    scale_fill_manual(values = "lightblue") +  
  labs(title = "Violin Plot - IBD",  
    x = "", y = "IBD") +  
  theme_minimal() +  
  scale_y_continuous(limits = c(0, 1)) +  
  coord_flip()
```

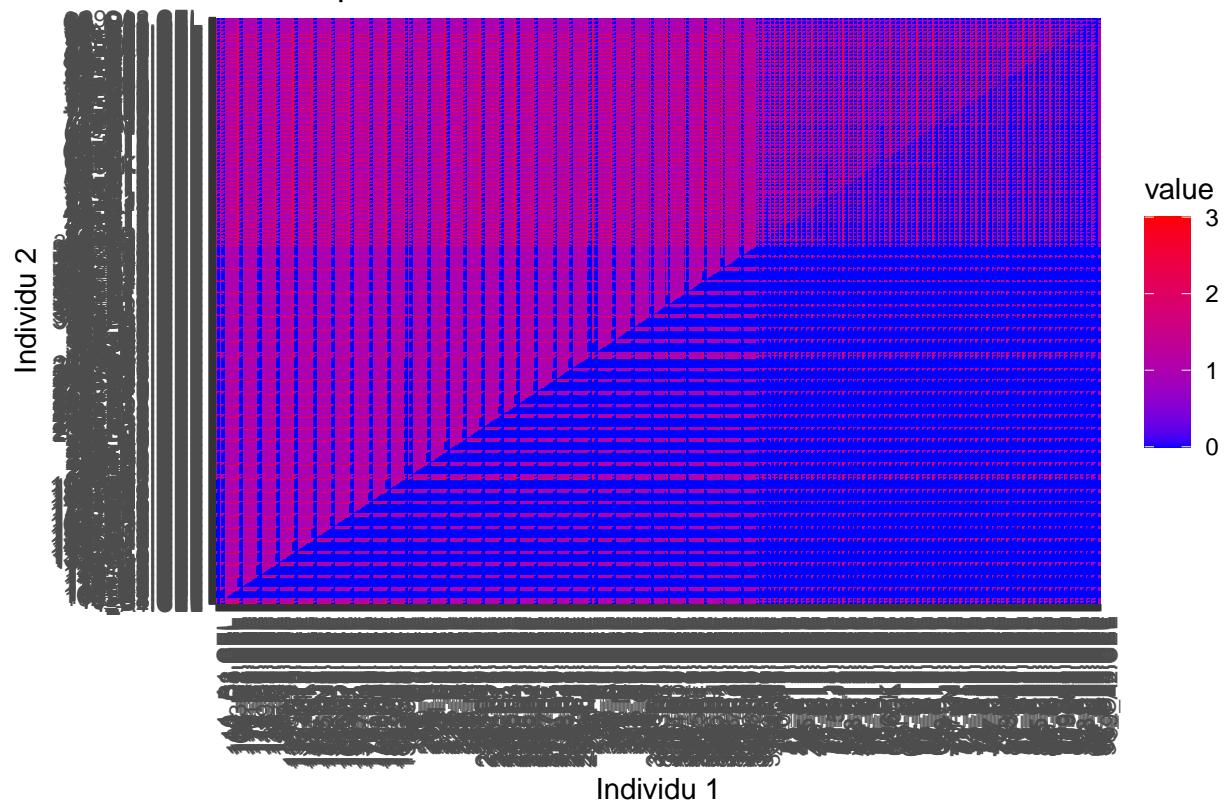
## Warning: Removed 16 rows containing missing values ('geom\_violin()').

## Violin Plot – IBD



```
# Tracer la heatmap
ggplot(melt(similarity_matrix), aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap - IBD - BeeMuSe",
       x = "Individu 1",
       y = "Individu 2") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

## Heatmap – IBD – BeeMuSe

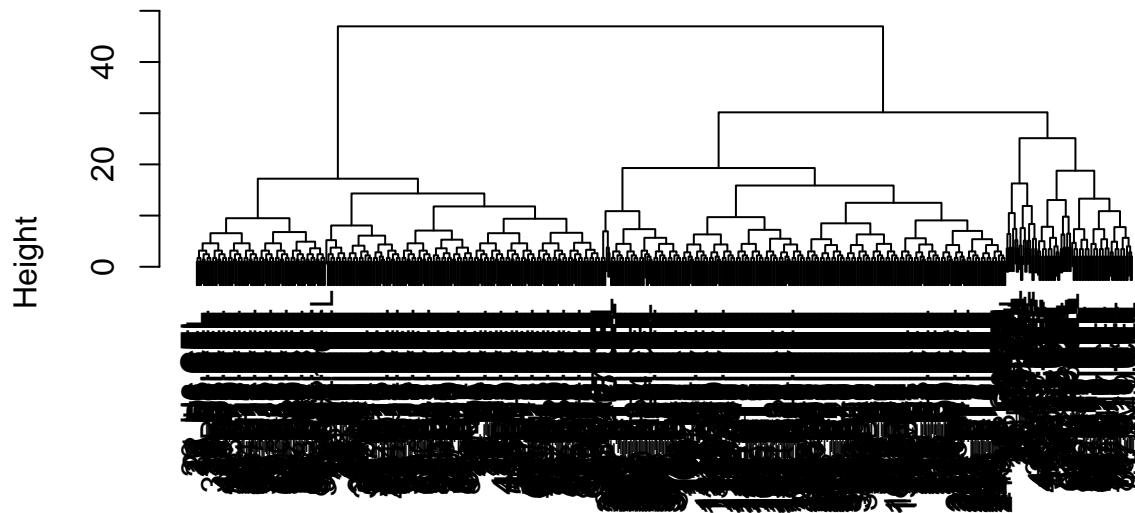


```
# Calcul de la matrice de similarité
similarity_matrix <- as.matrix(similarity_matrix)

# Calcul de la distance euclidienne
distance <- dist(1 - similarity_matrix)

# Création du dendrogramme
dendrogram <- hclust(distance)
plot(dendrogram, main = "Cluster Dendrogram - IBD Similarity")
```

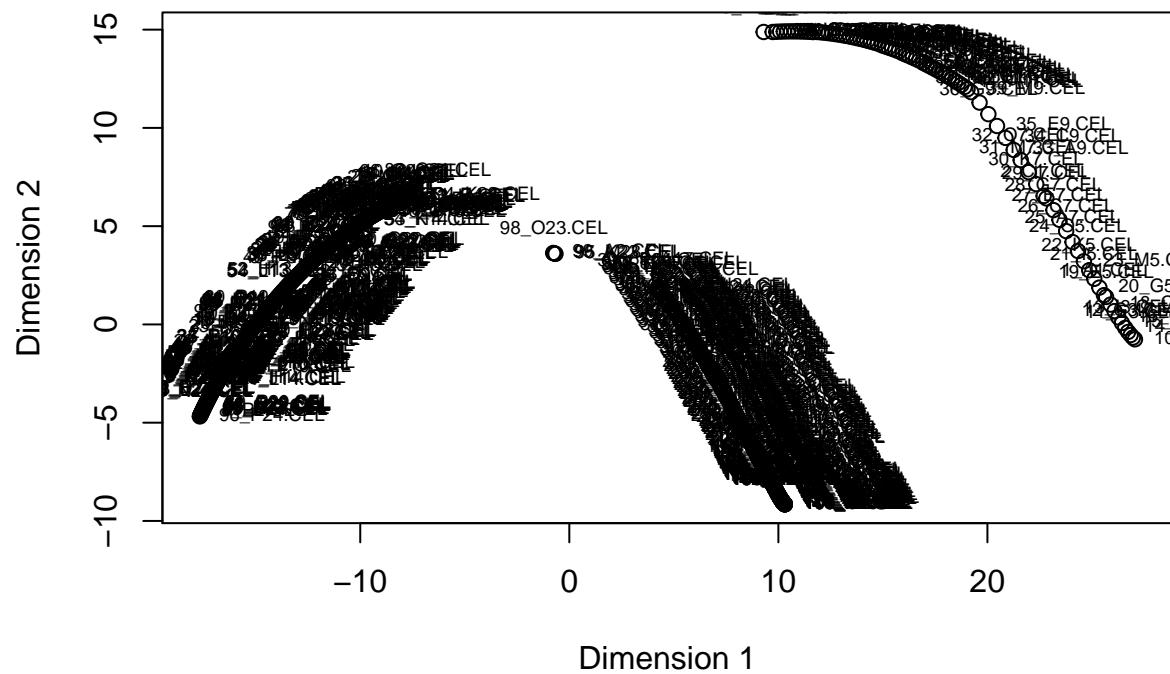
## Cluster Dendrogram – IBD Similarity



```
distance  
hclust (*, "complete")
```

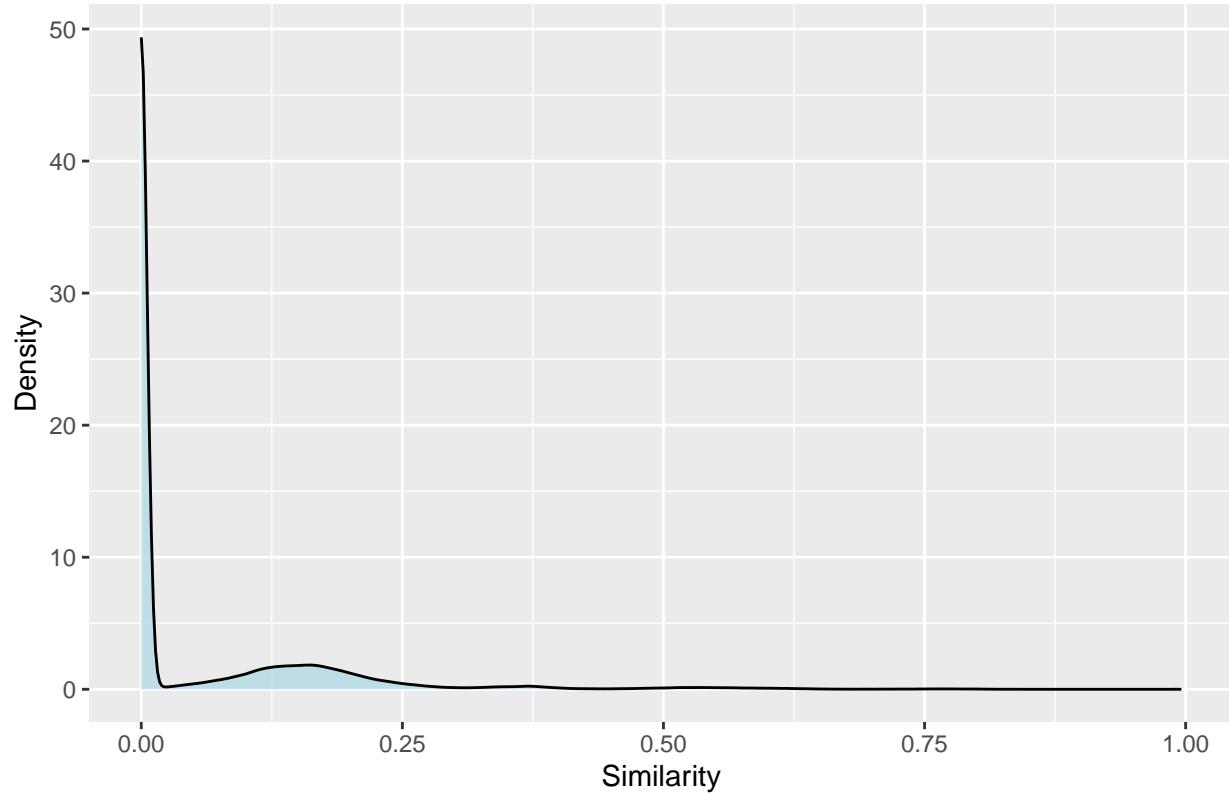
```
#library(MASS)  
# MDS  
mds <- cmdscale(distance)  
  
# Cr ation du plot MDS avec noms d'individus  
plot(mds, xlab = "Dimension 1", ylab = "Dimension 2", main = "MDS Plot - IBD Similarity")  
text(mds, labels = rownames(similarity_matrix), pos = c(3, 4), col = "black", cex = 0.6)
```

## MDS Plot – IBD Similarity



```
# Distribution de la similarité
ggplot(beemuse_genome, aes(x = PI_HAT)) +
  geom_density(fill = "lightblue", alpha = 0.7) +
  labs(title = "Distribution Plot – IBD Similarity", x = "Similarity", y = "Density")
```

## Distribution Plot – IBD Similarity



```
# Définir la palette de couleurs pastel
pastel_colors <- c(
  "#FF0000", "#FF3300", "#FF6600", "#FF9900", "#FFCC00",
  "#FFFF00", "#CCFF00", "#99FF00", "#66FF00", "#33FF00",
  "#00FF00", "#00FF33", "#00FF66", "#00FF99", "#00FFCC",
  "#00FFF", "#00CCFF", "#0099FF", "#0066FF", "#0033FF",
  "#0000FF", "#3300FF", "#6600FF", "#9900FF", "#CC00FF",
  "#FF00FF", "#FF00CC", "#FF0099", "#FF0066"
)

# Création d'un facteur pour distinguer les différentes données
beemuse_BAH_20_19_genome$Dataset <- "BAH_20-19"
beemuse_BBS_6_19_genome$Dataset <- "BBS_6-19"
beemuse_BER_11_19_genome$Dataset <- "BER_11-19"
beemuse_BH_44_genome$Dataset <- "BH_44"
beemuse_BH_7_19_genome$Dataset <- "BH_7-19"
beemuse_BHA_2_20_genome$Dataset <- "BHA_2-20"
beemuse_BLS_53_19_genome$Dataset <- "BLS_53-19"
beemuse_ER_13_19_genome$Dataset <- "ER_13-19"
beemuse_KBJ_1_19_genome$Dataset <- "KBJ_1-19"
beemuse_KBru_6_20_genome$Dataset <- "KBru_6-20"
beemuse_KLoc_37_19_genome$Dataset <- "KLoc_37-19"
beemuse_KLSU_14_19_genome$Dataset <- "KLSU_14-19"
```

```

beemuse_MM_31_20_genome$Dataset <- "MM_31-20"
beemuse_MM_37_20_genome$Dataset <- "MM_37-20"
beemuse_MP_10_20_genome$Dataset <- "MP_10-20"
beemuse_PersoBC_2021_genome$Dataset <- "PersoBC_2021"
beemuse_PersoJLL_2021_genome$Dataset <- "PersoJLL_2021"
beemuse_PersoJLL_2022_genome$Dataset <- "PersoJLL_2022"
beemuse_PersoLD_2021_genome$Dataset <- "PersoLD_2021"
beemuse_PersoLD_2022_genome$Dataset <- "PersoLD_2022"
beemuse_PersoUB_2021_genome$Dataset <- "PersoUB_2021"
beemuse_PersoUB_2022_genome$Dataset <- "PersoUB_2022"
beemuse_S_GZ_2_19_genome$Dataset <- "S_GZ_2-19"
beemuse_SBJ_3_19_genome$Dataset <- "SBJ_3-19"
beemuse_SJ_16_20_genome$Dataset <- "SJ_16-20"
beemuse_SJ_24_20_genome$Dataset <- "SJ_24-20"
beemuse_SJ_30_20_genome$Dataset <- "SJ_30-20"
beemuse_TL_13_20_genome$Dataset <- "TL_13-20"
beemuse_TL_19_20_genome$Dataset <- "TL_19-20"
#beemuse_unknown_genome$Dataset <- "Unknown"

# Combiner les ensembles de données
all_data <- rbind(beemuse_BAH_20_19_genome, beemuse_BBS_6_19_genome, beemuse_BER_11_19_genome, beemuse_BLA_53_19_genome, beemuse_ER_13_19_genome, beemuse_KBJ_1_19_genome, beemuse_KBru_6_20_genome, beemuse_KLoc_37_19_genome, beemuse_KLSU_14_19_genome, beemuse_MM_31_20_genome, beemuse_MM_37_20_genome, beemuse_MP_10_20_genome, beemuse_PersoBC_2021_genome, beemuse_PersoJLL_2021_genome, beemuse_PersoJLL_2022_genome, beemuse_PersoLD_2021_genome, beemuse_PersoLD_2022_genome, beemuse_PersoUB_2021_genome, beemuse_PersoUB_2022_genome, beemuse_S_GZ_2_19_genome, beemuse_SBJ_3_19_genome, beemuse_SJ_16_20_genome, beemuse_SJ_24_20_genome, beemuse_SJ_30_20_genome, beemuse_TL_13_20_genome, beemuse_TL_19_20_genome, beemuse_unknown_genome)

# Define the order of the groups and reverse it
group_order <- c(
  "BAH_20-19", "BBS_6-19", "BER_11-19", "BH_44", "BH_7-19", "BHA_2-20",
  "BLS_53-19", "ER_13-19", "KBJ_1-19", "KBru_6-20", "KLoc_37-19",
  "KLSU_14-19", "MM_31-20", "MM_37-20", "MP_10-20", "PersoBC_2021",
  "PersoJLL_2021", "PersoJLL_2022", "PersoLD_2021", "PersoLD_2022",
  "PersoUB_2021", "PersoUB_2022", "S_GZ_2-19", "SBJ_3-19", "SJ_16-20",
  "SJ_24-20", "SJ_30-20", "TL_13-20", "TL_19-20"
)
group_order <- rev(group_order)

# Convert the Dataset variable to a factor with the reversed order
all_data$Dataset <- factor(all_data$Dataset, levels = group_order)

# Créer le graphique de violon pour IBD
ggplot(all_data, aes(x = PI_HAT, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.1) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width = 0.9)) +
  scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBD",
       x = "IBD", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

```

## Analyse IBD - ID\_2a - 29 groupes + Unknown (pedigree inconnue)

```

## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

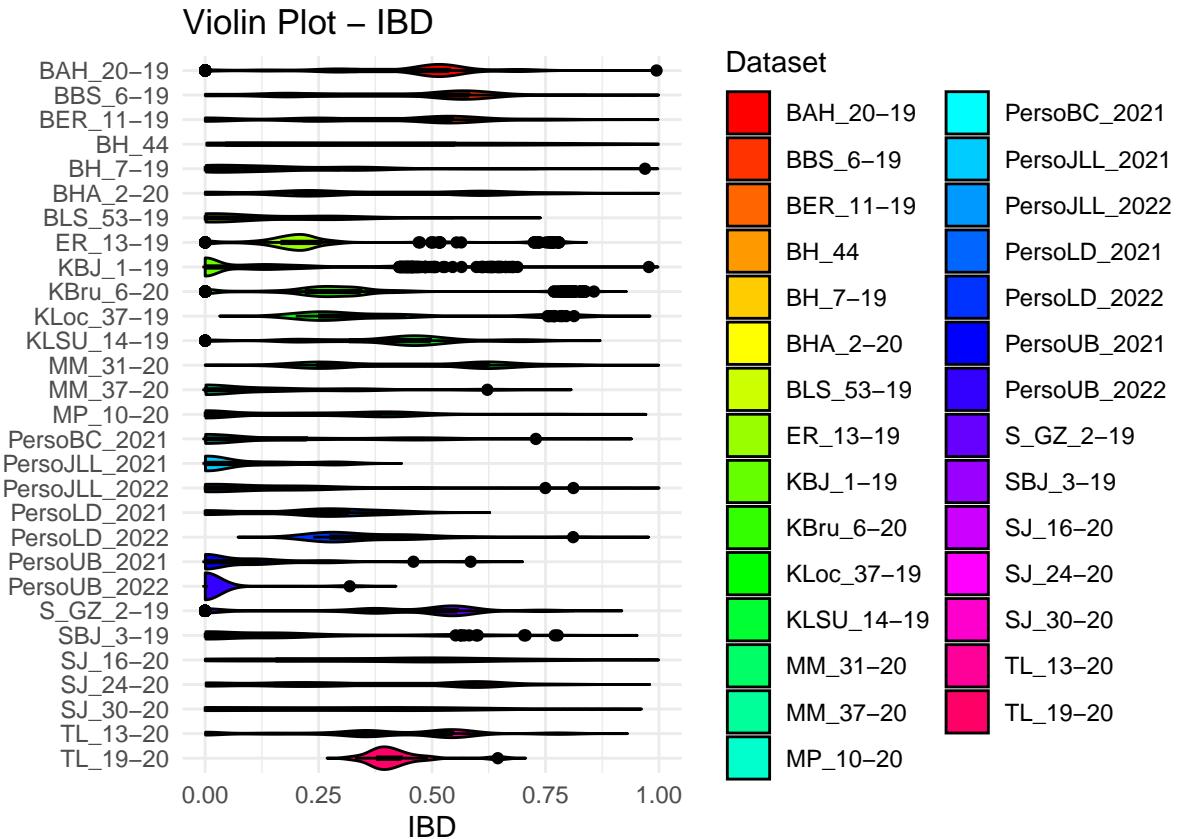
```

```

## Warning: 'position_dodge()' requires non-overlapping x intervals

## Warning: Removed 2495 rows containing missing values ('geom_violin()').

```



```

# Create the violin plot for IBS
ggplot(all_data, aes(x = DST, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.2) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBS",
      x = "IBS", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

```

```

## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

```

```

## Warning: 'position_dodge()' requires non-overlapping x intervals

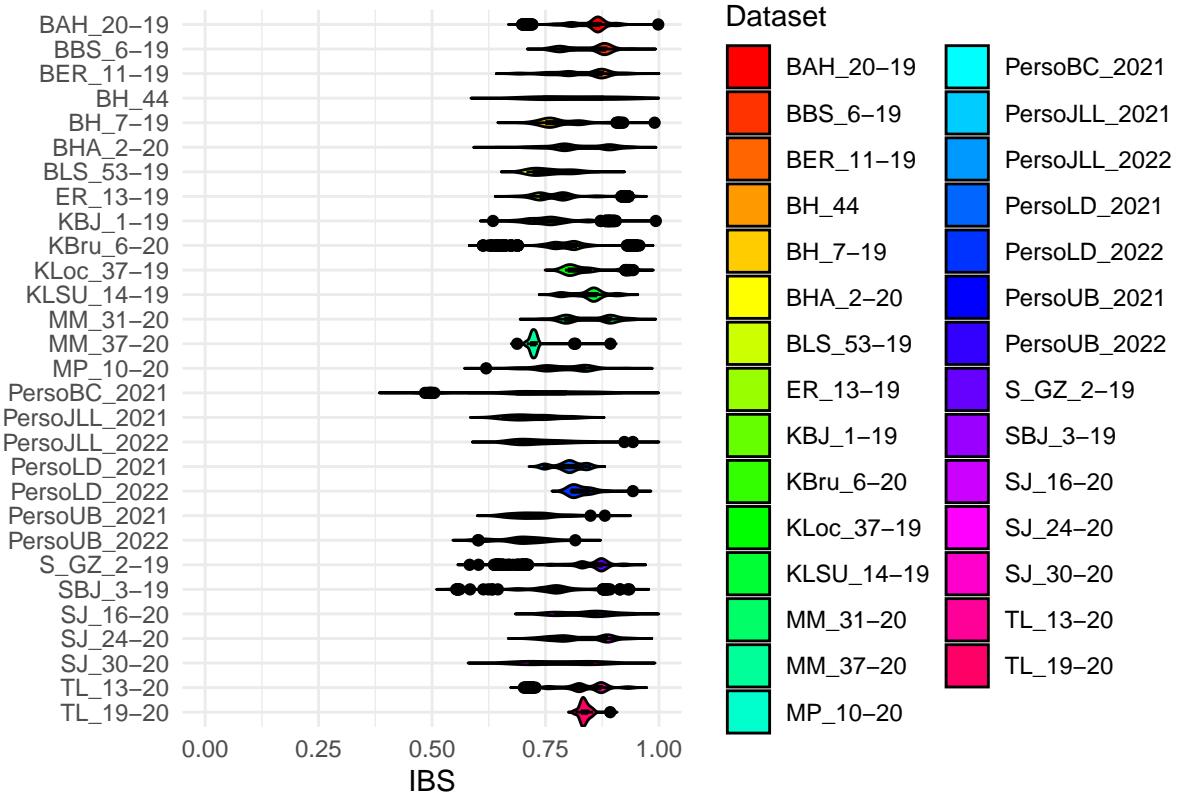
```

```

## Warning: Removed 292 rows containing missing values ('geom_violin()').

```

## Violin Plot – IBS



```

# 15 premières familles
# Définir la palette de couleurs pastel
pastel_colors <- c(
  "#FF0000", "#FF3300", "#FF6600", "#FF9900", "#FFCC00",
  "#FFFF00", "#CCFF00", "#99FF00", "#66FF00", "#33FF00",
  "#00FF00", "#00FF33", "#00FF66", "#00FF99", "#00FFCC")

# Création d'un facteur pour distinguer les différentes données
beemuse_BAH_20_19_genome$Dataset <- "BAH_20-19"
beemuse_BBS_6_19_genome$Dataset <- "BBS_6-19"
beemuse_BER_11_19_genome$Dataset <- "BER_11-19"
beemuse_BH_44_genome$Dataset <- "BH_44"
beemuse_BH_7_19_genome$Dataset <- "BH_7-19"
beemuse_BHA_2_20_genome$Dataset <- "BHA_2-20"
beemuse_BLS_53_19_genome$Dataset <- "BLS_53-19"
beemuse_ER_13_19_genome$Dataset <- "ER_13-19"
beemuse_KBJ_1_19_genome$Dataset <- "KBJ_1-19"
beemuse_KBru_6_20_genome$Dataset <- "KBru_6-20"
beemuse_KLoc_37_19_genome$Dataset <- "KLoc_37-19"
beemuse_KLSU_14_19_genome$Dataset <- "KLSU_14-19"
beemuse_MM_31_20_genome$Dataset <- "MM_31-20"
beemuse_MM_37_20_genome$Dataset <- "MM_37-20"
beemuse_MP_10_20_genome$Dataset <- "MP_10-20"

# Combiner les ensembles de données
all_data <- rbind(beemuse_BAH_20_19_genome, beemuse_BBS_6_19_genome, beemuse_BER_11_19_genome, beemuse_BH_44_genome, beemuse_BH_7_19_genome, beemuse_BHA_2_20_genome, beemuse_BLS_53_19_genome, beemuse_ER_13_19_genome, beemuse_KBJ_1_19_genome, beemuse_KBru_6_20_genome, beemuse_KLoc_37_19_genome, beemuse_KLSU_14_19_genome, beemuse_MM_31_20_genome, beemuse_MM_37_20_genome, beemuse_MP_10_20_genome)
  
```

```

# Définir l'ordre des groupes
group_order <- c(
  "BAH_20-19", "BBS_6-19", "BER_11-19", "BH_44", "BH_7-19", "BHA_2-20",
  "BLS_53-19", "ER_13-19", "KBJ_1-19", "KBru_6-20", "KLoc_37-19",
  "KLSU_14-19", "MM_31-20", "MM_37-20", "MP_10-20"
)

group_order <- rev(group_order)

# Convertir la variable Dataset en un facteur avec l'ordre spécifié
all_data$Dataset <- factor(all_data$Dataset, levels = group_order)

# Créer le graphique de violon pour IBS
ggplot(all_data, aes(x = DST, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.3) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBS",
      x = "IBS", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

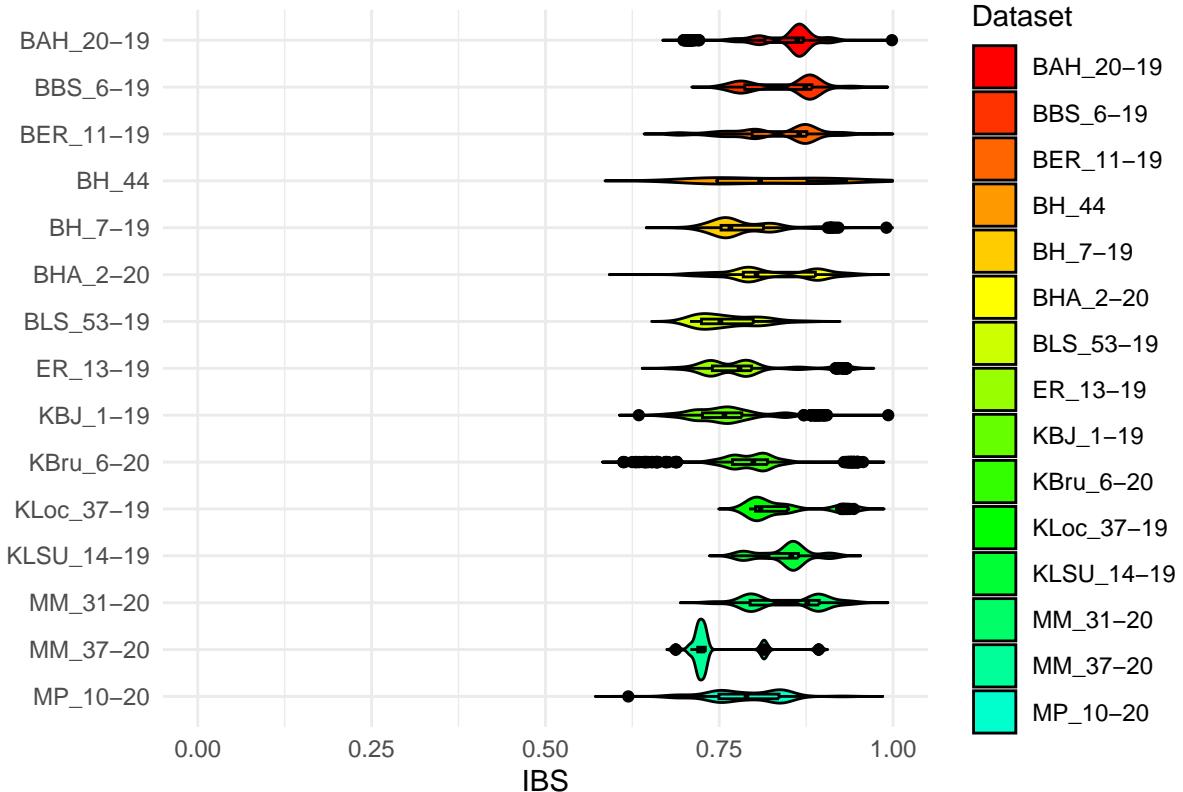
## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

## Warning: `position_dodge()` requires non-overlapping x intervals

## Warning: Removed 232 rows containing missing values (`geom_violin()`).

```

## Violin Plot – IBS



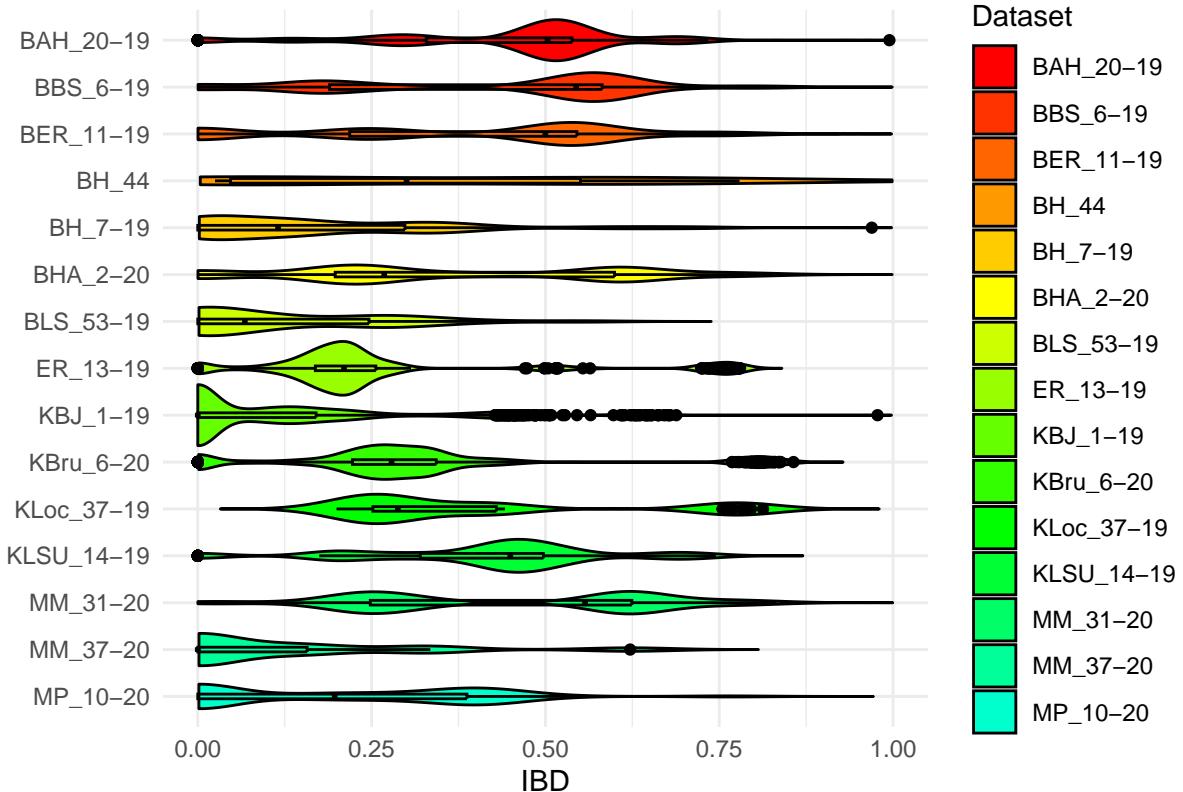
```
# Créer le graphique de violon pour IBD
ggplot(all_data, aes(x = PI_HAT, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.3) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBD",
      x = "IBD", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

## Warning: 'position_dodge()' requires non-overlapping x intervals

## Warning: Removed 1295 rows containing missing values ('geom_violin()').
```

## Violin Plot – IBD



```

# 15 autres familles (16 - 30)
# Définir la palette de couleurs pastel
pastel_colors <- c(
  "#00FFFF", "#00CCFF", "#0099FF", "#0066FF", "#0033FF",
  "#0000FF", "#3300FF", "#6600FF", "#9900FF", "#CC00FF",
  "#FF00FF", "#FF00CC", "#FF0099", "#FF0066", "#FF0033")

# Création d'un facteur pour distinguer les différentes données
beemuse_PersoBC_2021_genome$Dataset <- "PersoBC_2021"
beemuse_PersoJLL_2021_genome$Dataset <- "PersoJLL_2021"
beemuse_PersoJLL_2022_genome$Dataset <- "PersoJLL_2022"
beemuse_PersoLD_2021_genome$Dataset <- "PersoLD_2021"
beemuse_PersoLD_2022_genome$Dataset <- "PersoLD_2022"
beemuse_PersoUB_2021_genome$Dataset <- "PersoUB_2021"
beemuse_PersoUB_2022_genome$Dataset <- "PersoUB_2022"
beemuse_S_GZ_2_19_genome$Dataset <- "S_GZ_2-19"
beemuse_SBJ_3_19_genome$Dataset <- "SBJ_3-19"
beemuse_SJ_16_20_genome$Dataset <- "SJ_16-20"
beemuse_SJ_24_20_genome$Dataset <- "SJ_24-20"
beemuse_SJ_30_20_genome$Dataset <- "SJ_30-20"
beemuse_TL_13_20_genome$Dataset <- "TL_13-20"
beemuse_TL_19_20_genome$Dataset <- "TL_19-20"
beemuse_unknown_genome$Dataset <- "Unknown"

# Combiner les ensembles de données
all_data <- rbind(beemuse_PersoBC_2021_genome, beemuse_PersoJLL_2021_genome, beemuse_PersoJLL_2022_genome)
  
```

```

# Définir l'ordre des groupes
group_order <- c("PersoBC_2021",
  "PersoJLL_2021", "PersoJLL_2022", "PersoLD_2021", "PersoLD_2022",
  "PersoUB_2021", "PersoUB_2022", "S_GZ_2-19", "SBJ_3-19", "SJ_16-20",
  "SJ_24-20", "SJ_30-20", "TL_13-20", "TL_19-20", "Unknown"
)

group_order <- rev(group_order)

# Convertir la variable Dataset en un facteur avec l'ordre spécifié
all_data$Dataset <- factor(all_data$Dataset, levels = group_order)

# Créer le graphique de violon pour IBS
ggplot(all_data, aes(x = DST, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.4) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBS",
      x = "IBS", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))

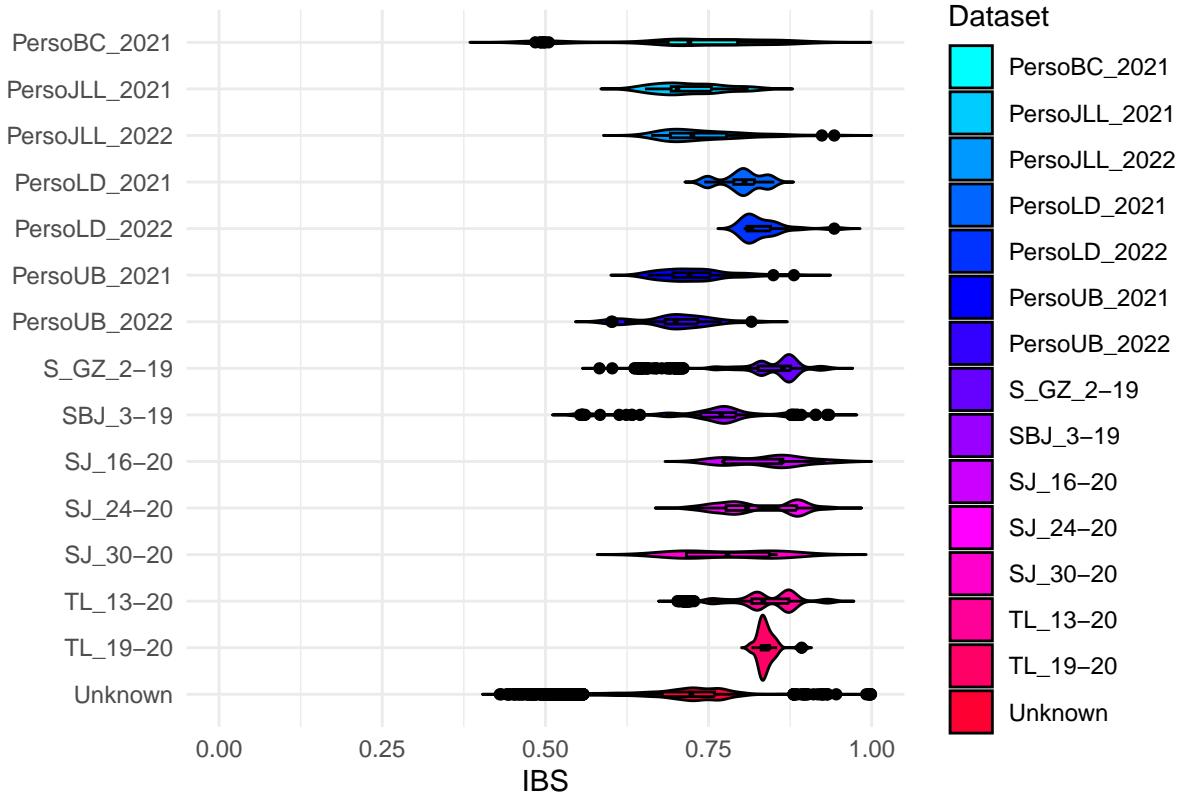
## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

## Warning: `position_dodge()` requires non-overlapping x intervals

## Warning: Removed 81 rows containing missing values (`geom_violin()`).

```

## Violin Plot – IBS



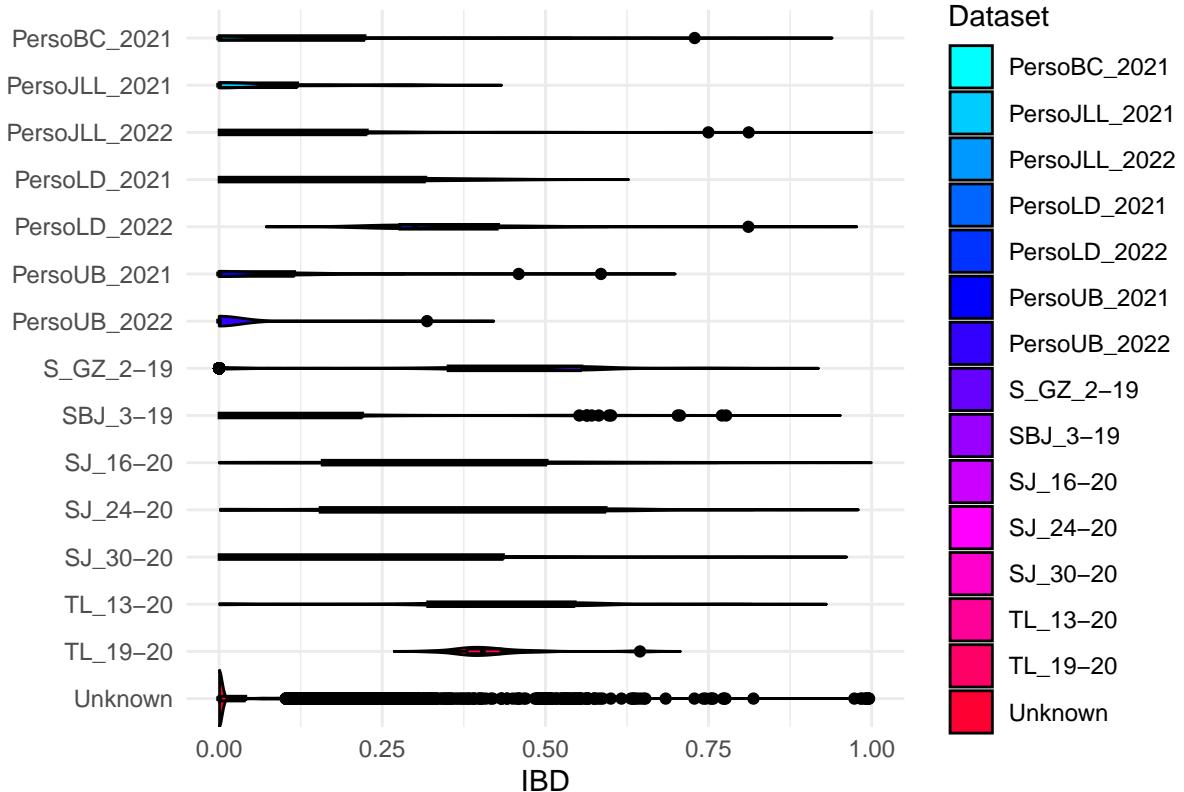
```
# Créer le graphique de violon pour IBD
ggplot(all_data, aes(x = PI_HAT, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.2) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - IBD",
      x = "IBD", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))
```

```
## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.
```

```
## Warning: ‘position_dodge()’ requires non-overlapping x intervals
```

```
## Warning: Removed 1212 rows containing missing values ('geom_violin()').
```

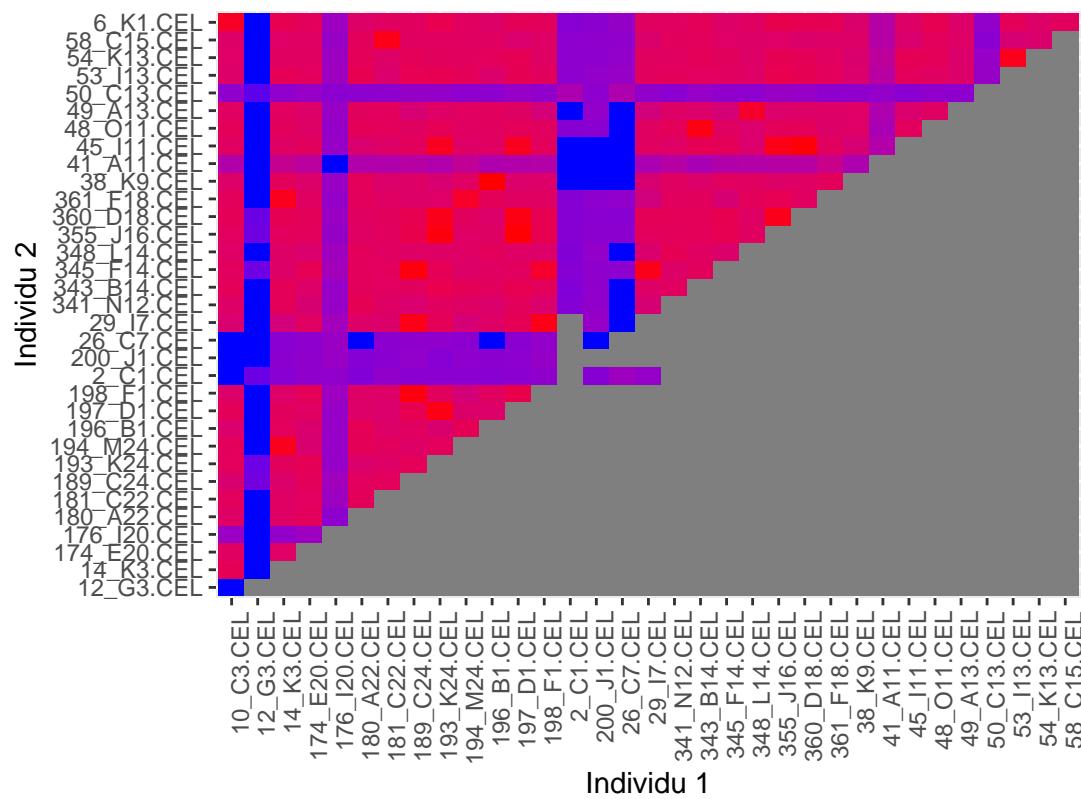
Violin Plot – IBD



```
similarity_matrix2 <- acast(beemuse_BBS_6_19_genome, IID1 ~ IID2, value.var = "PI_HAT")

# Tracer la heatmap
ggplot(melt(similarity_matrix2), aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap - IBD - BeeMuSe",
       x = "Individu 1",
       y = "Individu 2") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Heatmap – IBD – BeeMuSe



Analyse IBD - BBS\_6-19

```
# Calcul de la matrice de similarité
similarity_matrix <- as.matrix(similarity_matrix2)

# Calcul de la distance euclidienne
distance <- dist(1 - similarity_matrix)

# Création du dendrogramme
dendrogram <- hclust(distance)
plot(dendrogram, main = "Cluster Dendrogram - IBD Similarity")
```

## Cluster Dendrogram – IBD Similarity

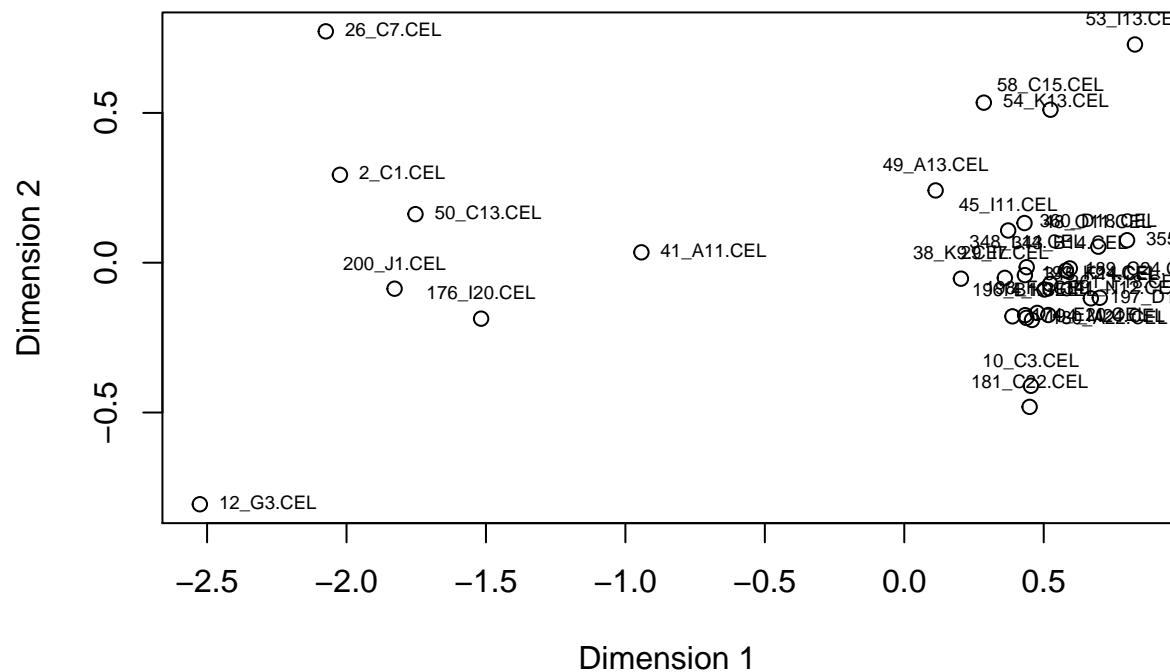


distance  
hclust (\*, "complete")

```
#library(MASS)
# MDS
mds <- cmdscale(distance)

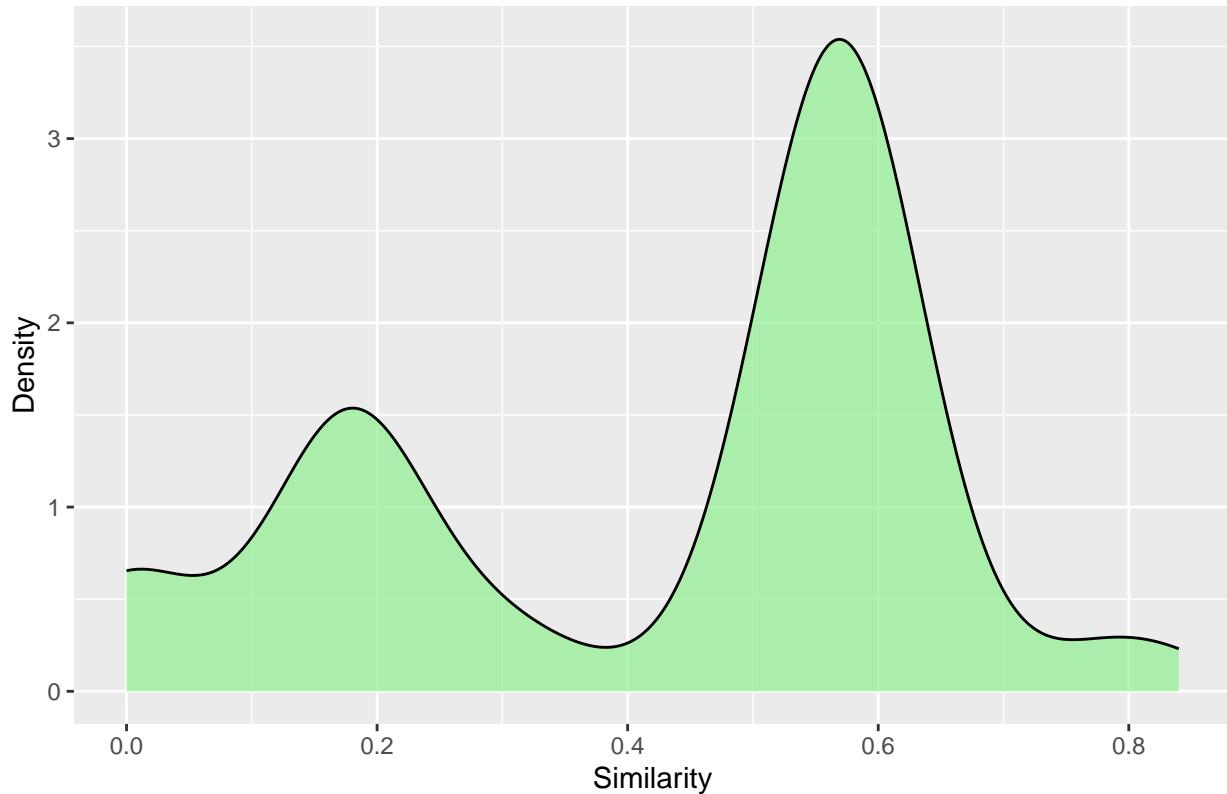
# Création du plot MDS avec noms d'individus
plot(mds, xlab = "Dimension 1", ylab = "Dimension 2", main = "MDS Plot - IBD Similarity")
text(mds, labels = rownames(similarity_matrix), pos = c(3, 4), col = "black", cex = 0.6)
```

## MDS Plot – IBD Similarity



```
# Distribution de la similarité
ggplot(beemuse_BBS_6_19_genome, aes(x = PI_HAT)) +
  geom_density(fill = "lightgreen", alpha = 0.7) +
  labs(title = "Distribution Plot – IBD Similarity", x = "Similarity", y = "Density")
```

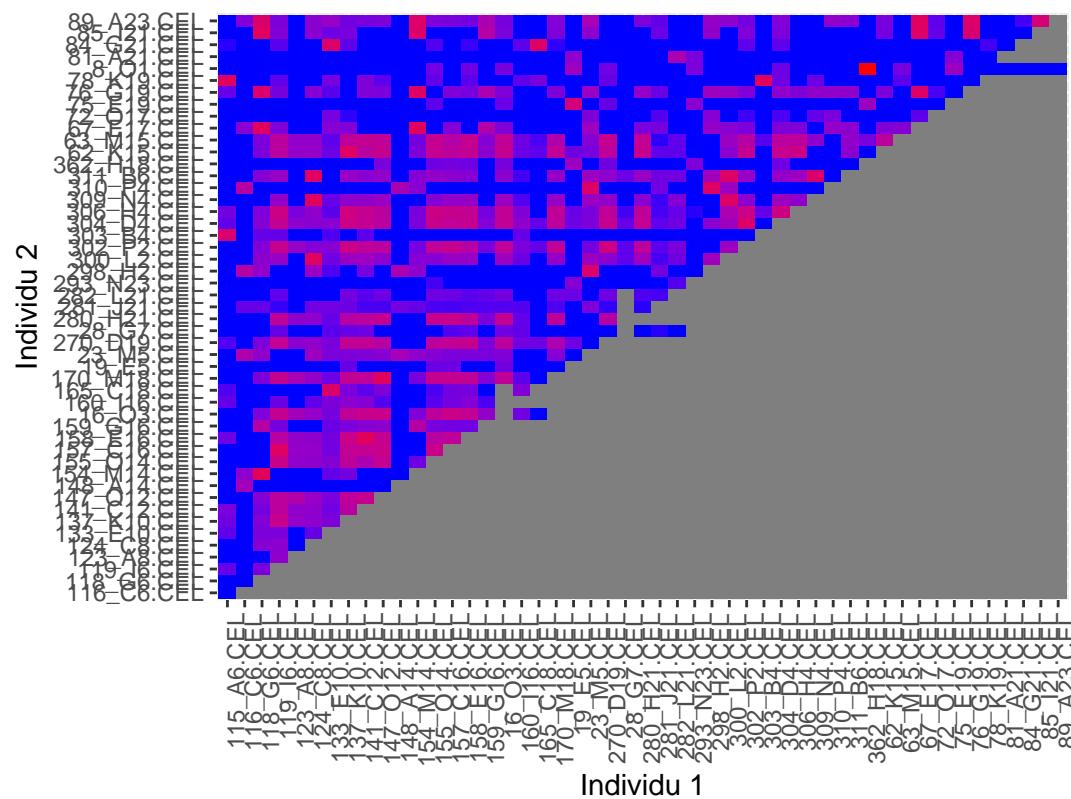
## Distribution Plot – IBD Similarity



```
#KBJ_1-19
similarity_matrix3 <- acast(beemuse_KBJ_1_19_genome, IID1 ~ IID2, value.var = "PI_HAT")

# Tracer la heatmap
ggplot(melt(similarity_matrix3), aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap - IBD - BeeMuSe",
       x = "Individu 1",
       y = "Individu 2") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Heatmap – IBD – BeeMuSe



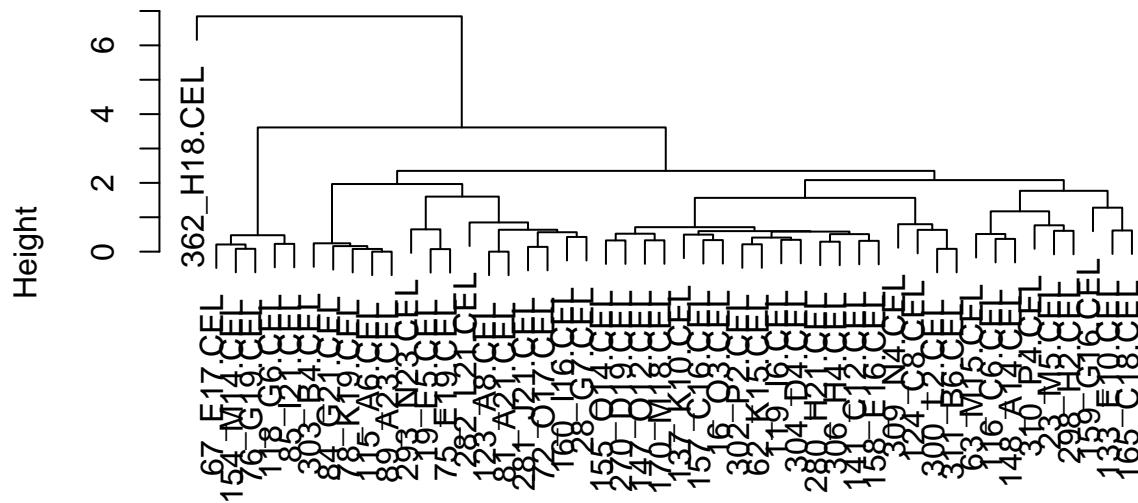
Analyse IBD - KBJ\_1-19

```
# Calcul de la matrice de similarité
similarity_matrix <- as.matrix(similarity_matrix3)

# Calcul de la distance euclidienne
distance <- dist(1 - similarity_matrix)

# Création du dendrogramme
dendrogram <- hclust(distance)
plot(dendrogram, main = "Cluster Dendrogram - IBD Similarity")
```

## Cluster Dendrogram – IBD Similarity

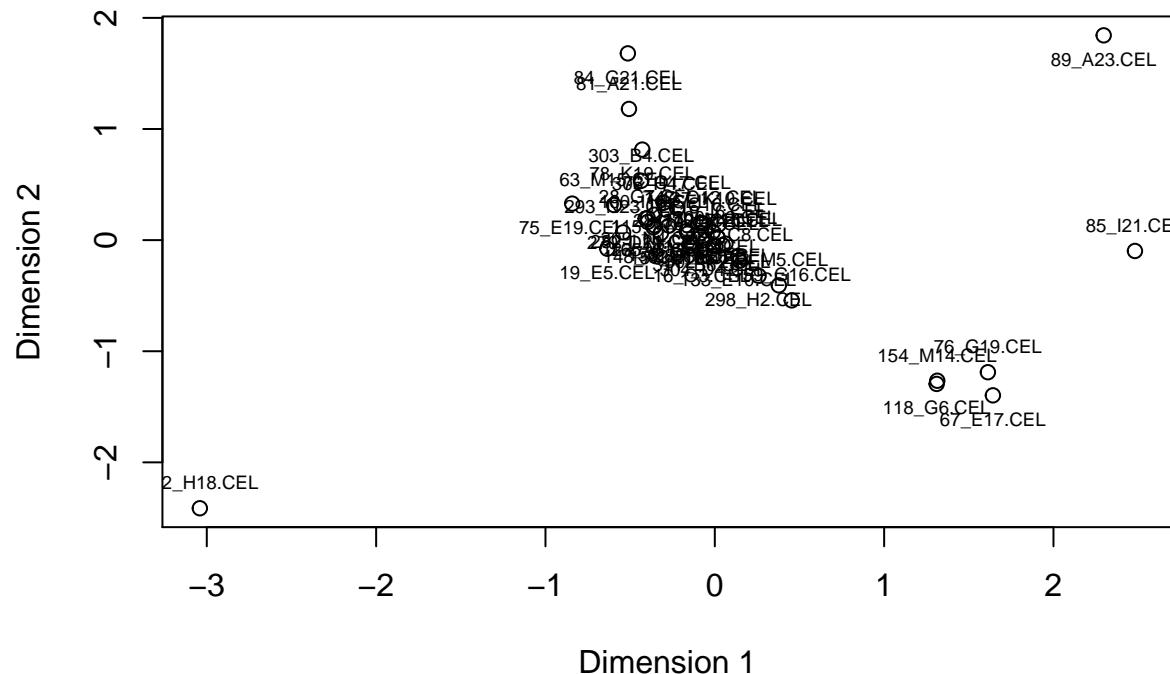


distance  
hclust (\*, "complete")

```
# library(MASS)
# MDS
mds <- cmdscale(distance)

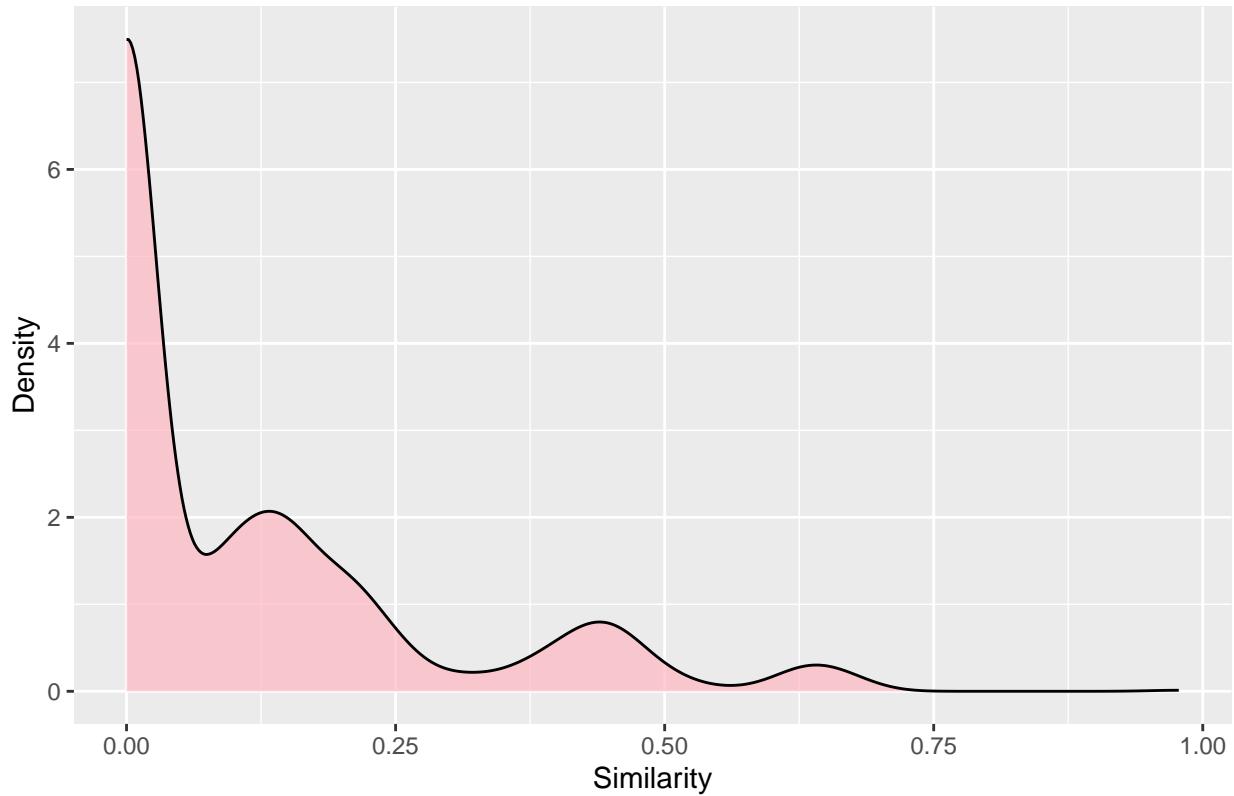
# Création du plot MDS avec noms d'individus
plot(mds, xlab = "Dimension 1", ylab = "Dimension 2", main = "MDS Plot - IBD Similarity")
text(mds, labels = rownames(similarity_matrix), pos = c(1, 3), col = "black", cex = 0.6)
```

## MDS Plot – IBD Similarity



```
# Distribution de la similarité
ggplot(beemuse_KBJ_1_19_genome, aes(x = PI_HAT)) +
  geom_density(fill = "lightpink", alpha = 0.7) +
  labs(title = "Distribution Plot – IBD Similarity", x = "Similarity", y = "Density")
```

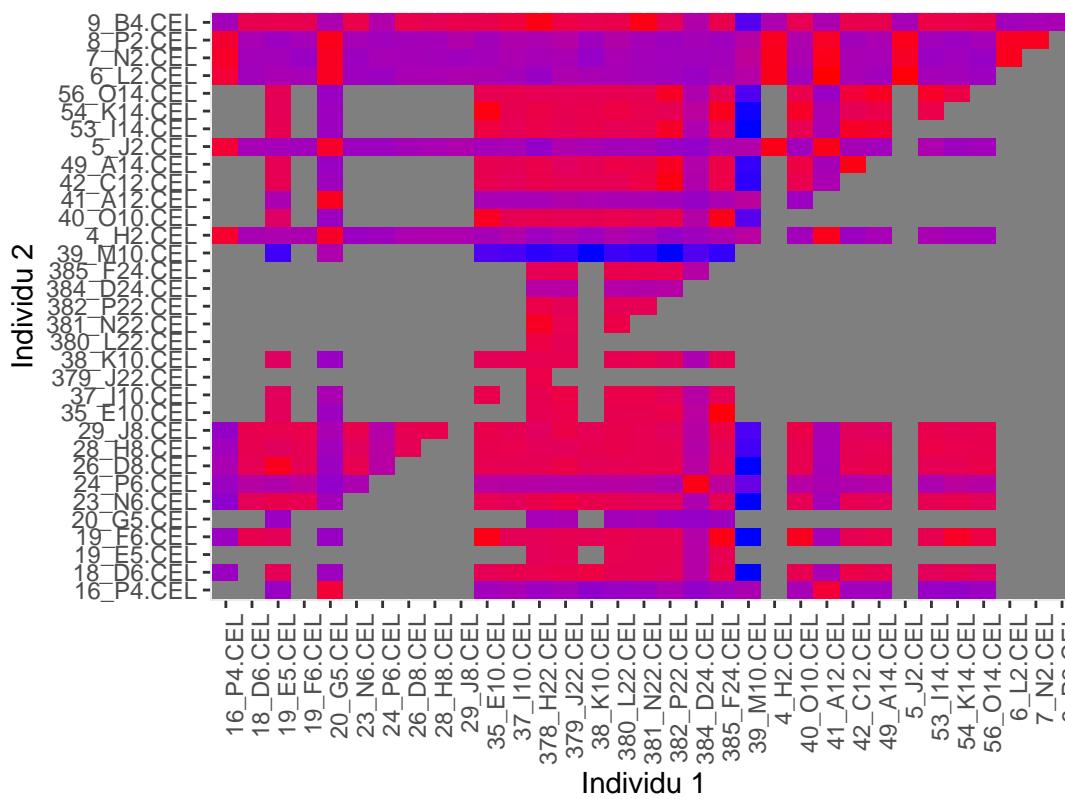
## Distribution Plot – IBD Similarity



```
similarity_matrix2 <- acast(beemuse_MM_31_20_genome, IID1 ~ IID2, value.var = "PI_HAT")

# Tracer la heatmap
ggplot(melt(similarity_matrix2), aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap - IBD - BeeMuSe",
       x = "Individu 1",
       y = "Individu 2") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Heatmap – IBD – BeeMuSe



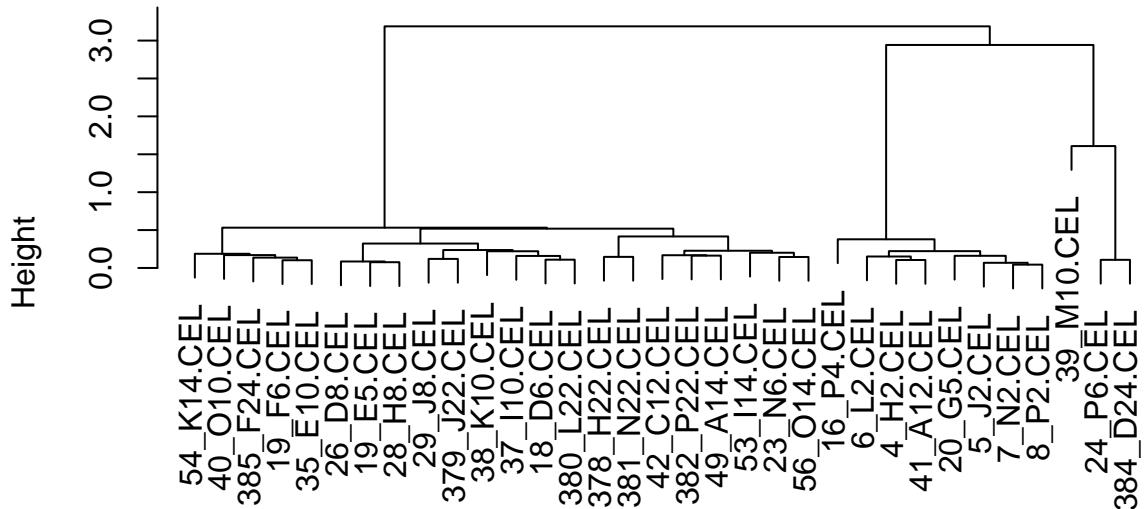
Analyse IBD - MM\_31-20

```
# Calcul de la matrice de similarité
similarity_matrix <- as.matrix(similarity_matrix2)

# Calcul de la distance euclidienne
distance <- dist(1 - similarity_matrix)

# Création du dendrogramme
dendrogram <- hclust(distance)
plot(dendrogram, main = "Cluster Dendrogram - IBD Similarity")
```

## Cluster Dendrogram – IBD Similarity

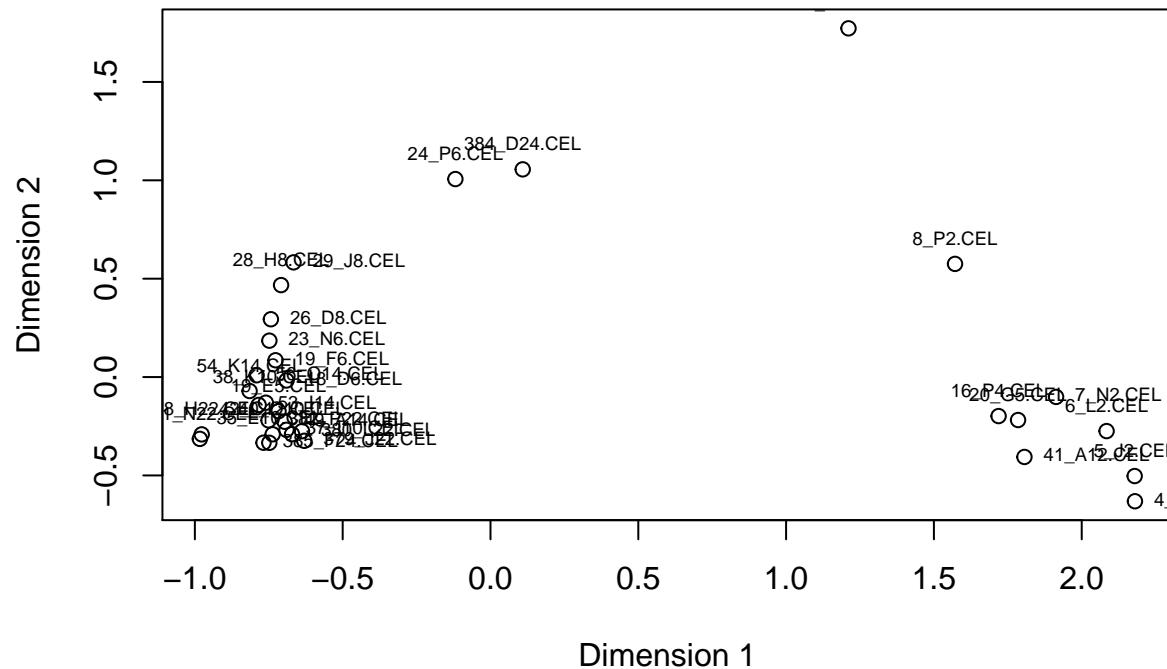


distance  
hclust (\*, "complete")

```
# library(MASS)
# MDS
mds <- cmdscale(distance)

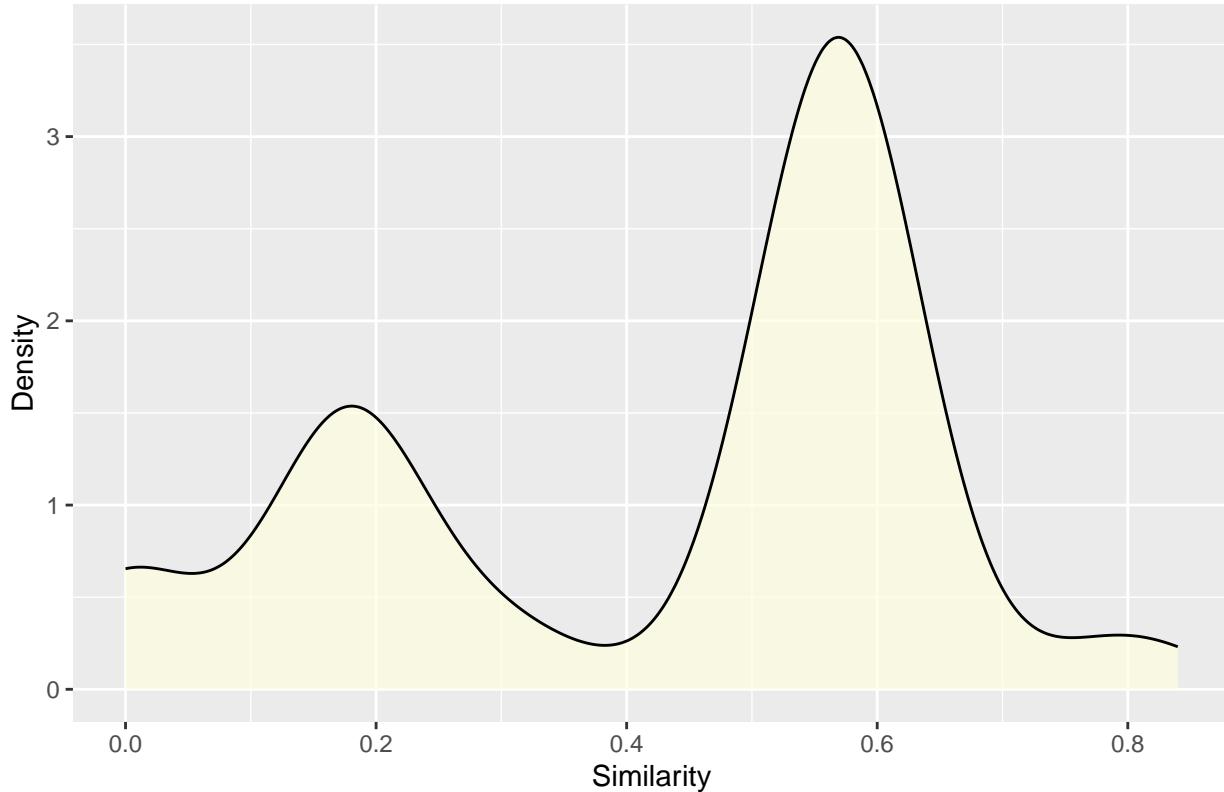
# Création du plot MDS avec noms d'individus
plot(mds, xlab = "Dimension 1", ylab = "Dimension 2", main = "MDS Plot - IBD Similarity")
text(mds, labels = rownames(similarity_matrix), pos = c(3, 4), col = "black", cex = 0.6)
```

## MDS Plot – IBD Similarity



```
# Distribution de la similarité
ggplot(beemuse_BBS_6_19_genome, aes(x = PI_HAT)) +
  geom_density(fill = "lightyellow", alpha = 0.7) +
  labs(title = "Distribution Plot – IBD Similarity", x = "Similarity", y = "Density")
```

## Distribution Plot – IBD Similarity



**“plink2 –make-king-table”** Enfin, on utilise la commande “plink2 –make-king-table” pour les 748 individus BeeMuSe, afin d’analyser le KINSHIP et IBS0.

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

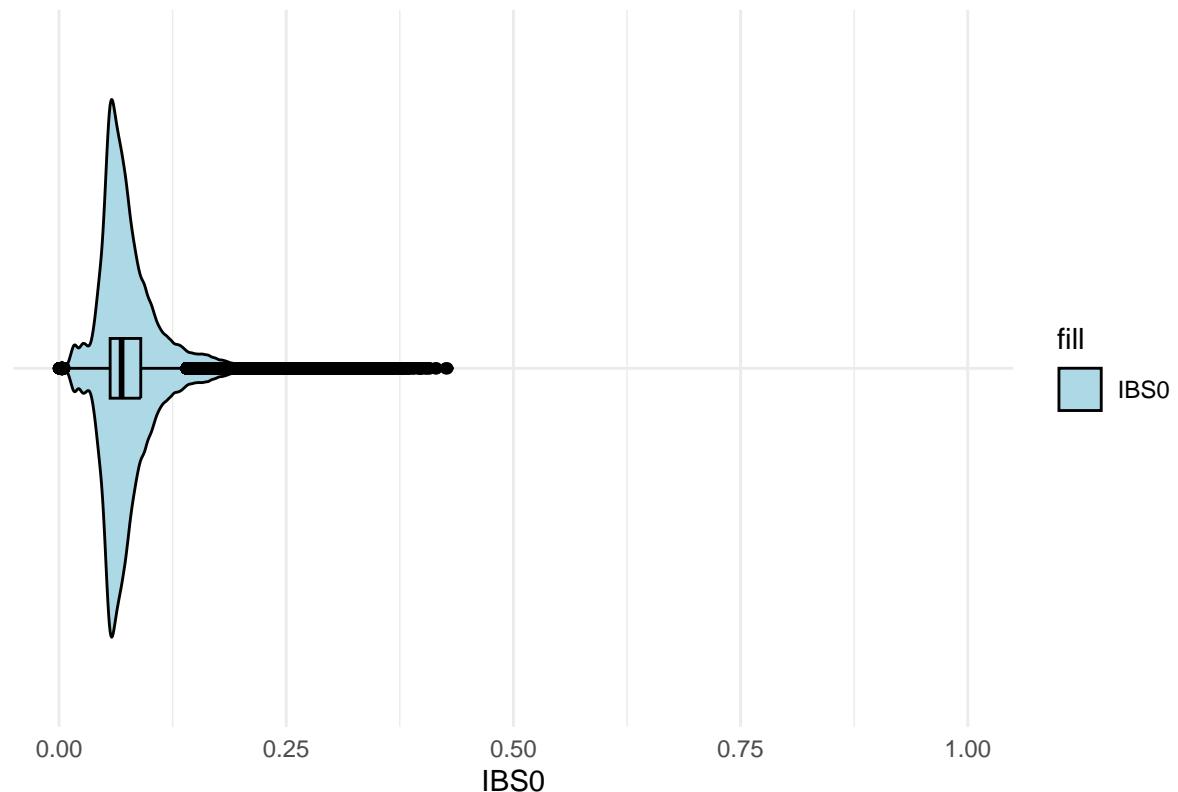
beemuse_p2_genome <- read.table("BeeMuse_plink2_genome.kin0", header=FALSE, sep="\t",
                                   col.names=c("FID1", "IID1", "FID2", "IID2", "NSNP", "HETHET", "IBS0", "KINSHIP"))

ggplot(beemuse_p2_genome, aes(x = "", y = IBS0, fill = "IBS0")) +
  geom_violin(trim = FALSE, color = "black") +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
    scale_fill_manual(values = "lightblue") +
  labs(title = "Violin Plot - IBS0",
       x = "", y = "IBS0") +
  theme_minimal() +
  scale_y_continuous(limits = c(0, 1)) +
  coord_flip()
```

Analyse KINSHIP + IBS0 - 748 échantillons

```
## Warning: Removed 7 rows containing missing values ('geom_violin()'').
```

## Violin Plot – IBS0



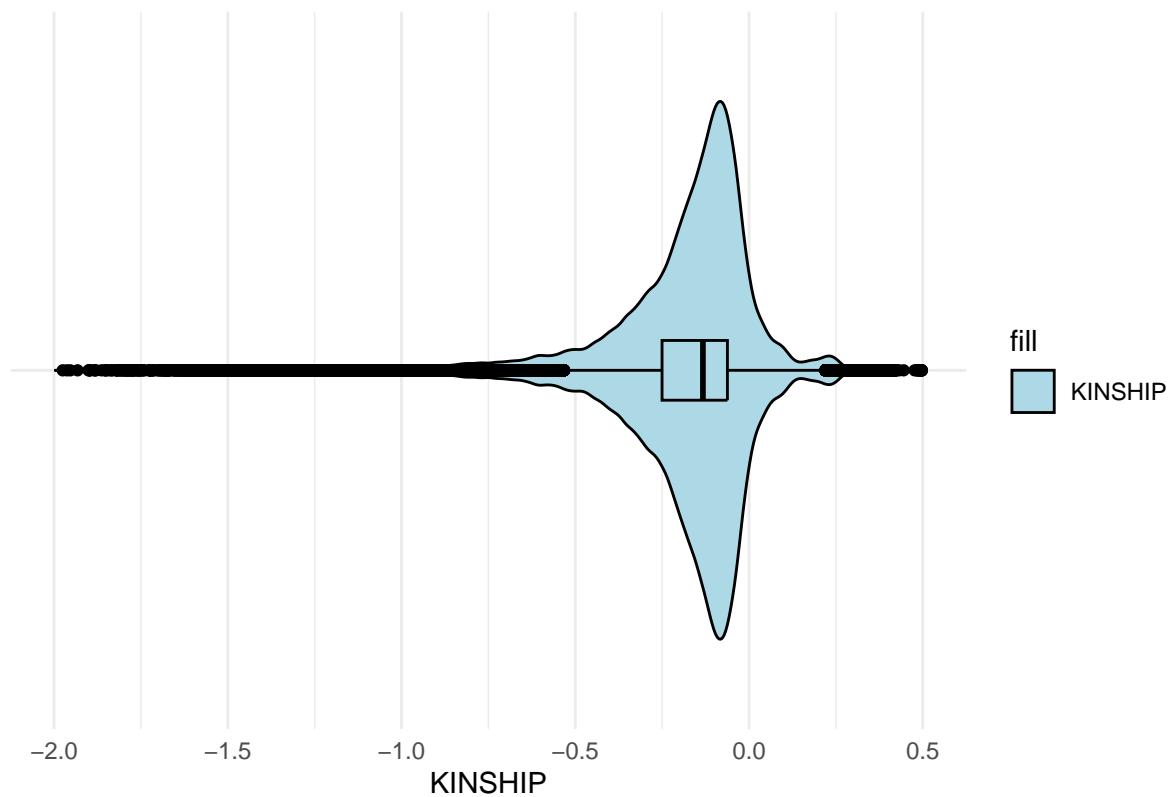
```
ggplot(beemuse_p2_genome, aes(x = "", y = KINSHIP, fill = "KINSHIP")) +  
  geom_violin(trim = FALSE, color = "black") +  
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =  
    scale_fill_manual(values = "lightblue") +  
  labs(title = "Violin Plot - KINSHIP",  
    x = "", y = "KINSHIP") +  
  theme_minimal() +  
  scale_y_continuous(limits = c(-2, 0.5)) +  
  coord_flip()
```

## Warning: Removed 776 rows containing non-finite values ('stat\_ydensity()').

## Warning: Removed 776 rows containing non-finite values ('stat\_boxplot()').

## Warning: Removed 8 rows containing missing values ('geom\_violin()').

## Violin Plot – KINSHIP

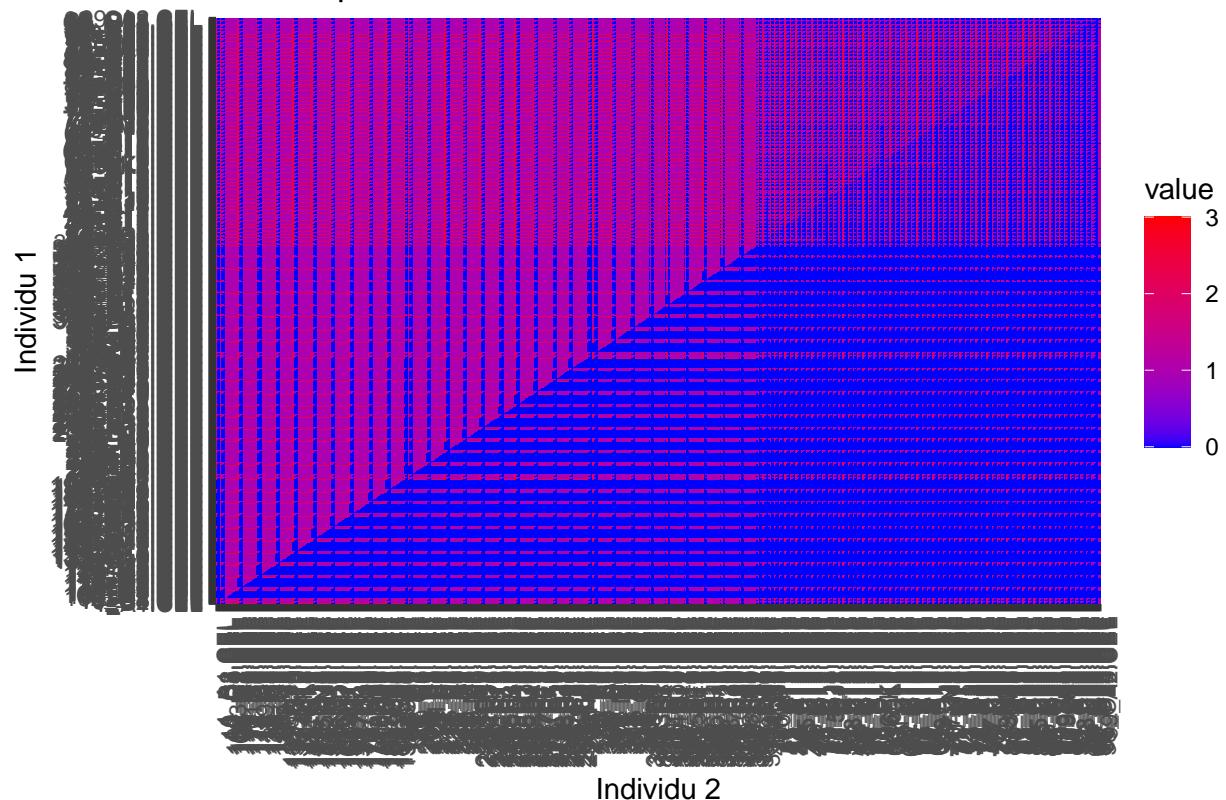


```
#library(ggplot2)
#library(reshape2)
similarity_matrix <- acast(beemuse_p2_genome, IID2 ~ IID1, value.var = "KINSHIP")
```

```
## Aggregation function missing: defaulting to length
```

```
# Tracer la heatmap
ggplot(melt(similarity_matrix), aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient(low = "blue", high = "red") +
  labs(title = "Heatmap - KINSHIP - BeeMuSe",
       x = "Individu 2",
       y = "Individu 1") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Heatmap – KINSHIP – BeeMuSe



```
# Calcul de la matrice de similarité
similarity_matrix <- as.matrix(similarity_matrix)

# Calcul de la distance euclidienne
distance <- dist(1 - similarity_matrix)

# Création du dendrogramme
dendrogram <- hclust(distance)
plot(dendrogram, main = "Cluster Dendrogram - KINSHIP Similarity")
```

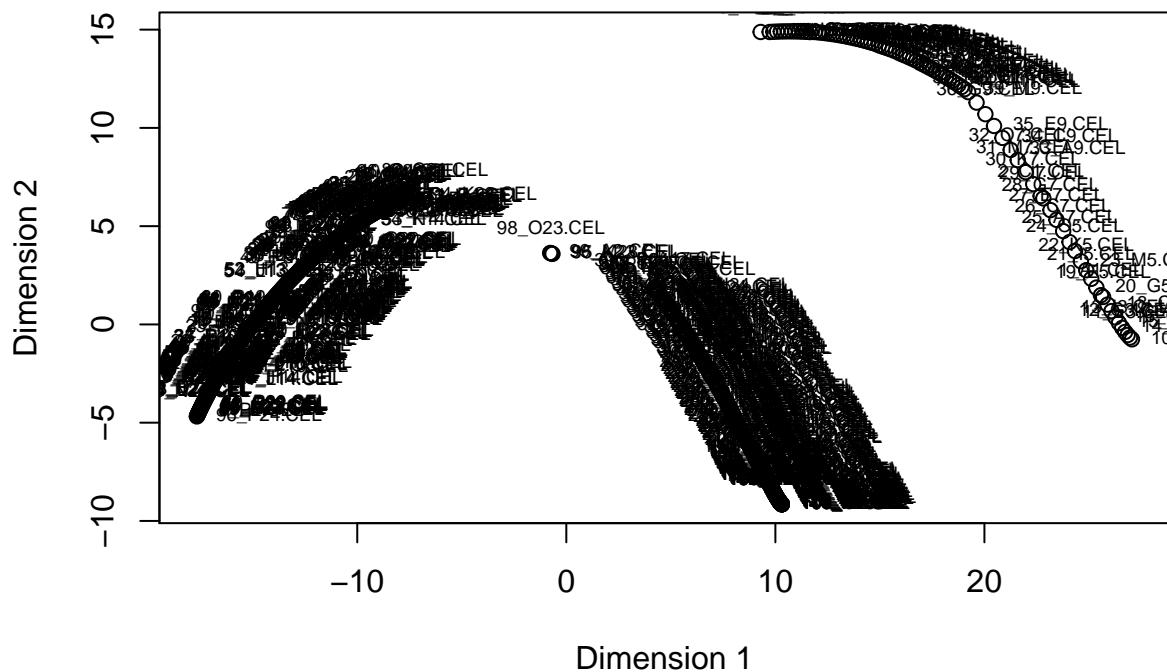
## Cluster Dendrogram – KINSHIP Similarity



```
distance  
hclust (*, "complete")
```

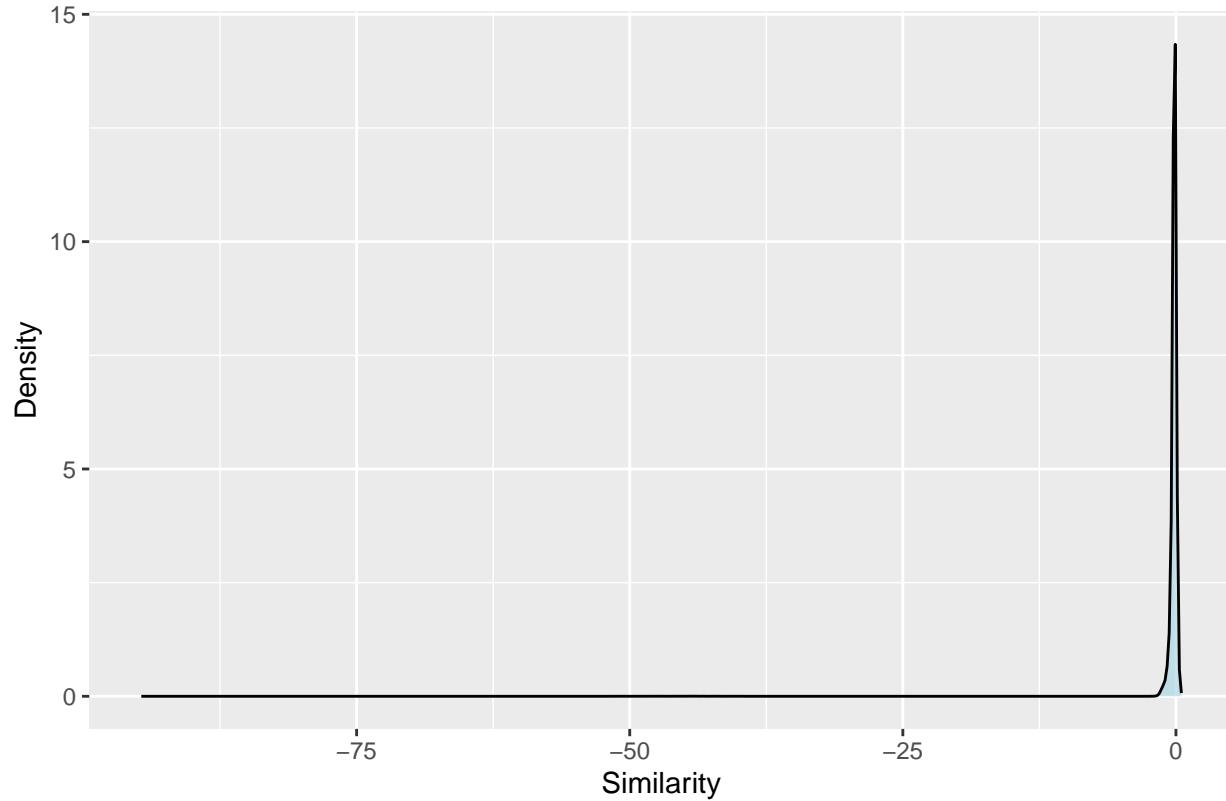
```
# library(MASS)  
# MDS  
mds <- cmdscale(distance)  
  
# Cr ation du plot MDS avec noms d'individus  
plot(mds, xlab = "Dimension 1", ylab = "Dimension 2", main = "MDS Plot - KINSHIP Similarity")  
text(mds, labels = rownames(similarity_matrix), pos = c(3, 4), col = "black", cex = 0.6)
```

## MDS Plot – KINSHIP Similarity



```
# Distribution de la similarité
ggplot(beemuse_p2_genome, aes(x = KINSHIP)) +
  geom_density(fill = "lightblue", alpha = 0.7) +
  labs(title = "Distribution Plot – KINSHIP Similarity", x = "Similarity", y = "Density")
```

## Distribution Plot – KINSHIP Similarity



**Analyse IBD intra-familles - ID\_2a - 29 groupes + Unknown (pedigree inconnue)** Après avoir effectué la commande “plink2 –make-king-table” pour les 748 individus BeeMuSe, on extrait les familles ID\_2a. On obtient 30 fichiers ‘plink2.genome’ pour chaque famille ID\_2a et les Unknown de pedigree inconnue regroupés ensemble.

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

beemuse_BAH_20_19_genome <- read.table("BAH_20-19_plink2.genome", header=TRUE)
beemuse_BBS_6_19_genome <- read.table("BBS_6-19_plink2.genome", header=TRUE)
beemuse_BER_11_19_genome <- read.table("BER_11-19_plink2.genome", header=TRUE)
beemuse_BH_44_genome <- read.table("BH_44_plink2.genome", header=TRUE)
beemuse_BH_7_19_genome <- read.table("BH_7-19_plink2.genome", header=TRUE)
beemuse_BHA_2_20_genome <- read.table("BHA_2-20_plink2.genome", header=TRUE)
beemuse_BLS_53_19_genome <- read.table("BLS_53-19_plink2.genome", header=TRUE)
beemuse_ER_13_19_genome <- read.table("ER_13-19_plink2.genome", header=TRUE)
beemuse_KBJ_1_19_genome <- read.table("KBJ_1-19_plink2.genome", header=TRUE)
beemuse_KBru_6_20_genome <- read.table("KBru_6-20_plink2.genome", header=TRUE)
beemuse_KLoc_37_19_genome <- read.table("KLoc_37-19_plink2.genome", header=TRUE)
beemuse_KLSU_14_19_genome <- read.table("KLSU_14-19_plink2.genome", header=TRUE)
beemuse_MM_31_20_genome <- read.table("MM_31-20_plink2.genome", header=TRUE)
beemuse_MM_37_20_genome <- read.table("MM_37-20_plink2.genome", header=TRUE)
beemuse_MP_10_20_genome <- read.table("MP_10-20_plink2.genome", header=TRUE)
beemuse_PersoBC_2021_genome <- read.table("PersoBC_2021_plink2.genome", header=TRUE)
beemuse_PersoJLL_2021_genome <- read.table("PersoJLL_2021_plink2.genome", header=TRUE)
beemuse_PersoJLL_2022_genome <- read.table("PersoJLL_2022_plink2.genome", header=TRUE)
```

```

beemuse_PersoLD_2021_genome <- read.table("PersoLD_2021_plink2.genome", header=TRUE)
beemuse_PersoLD_2022_genome <- read.table("PersoLD_2022_plink2.genome", header=TRUE)
beemuse_PersoUB_2021_genome <- read.table("PersoUB_2021_plink2.genome", header=TRUE)
beemuse_PersoUB_2022_genome <- read.table("PersoUB_2022_plink2.genome", header=TRUE)
beemuse_S_GZ_2_19_genome <- read.table("S_GZ_2-19_plink2.genome", header=TRUE)
beemuse_SBJ_3_19_genome <- read.table("SBJ_3-19_plink2.genome", header=TRUE)
beemuse_SJ_16_20_genome <- read.table("SJ_16-20_plink2.genome", header=TRUE)
beemuse_SJ_24_20_genome <- read.table("SJ_24-20_plink2.genome", header=TRUE)
beemuse_SJ_30_20_genome <- read.table("SJ_30-20_plink2.genome", header=TRUE)
beemuse_TL_13_20_genome <- read.table("TL_13-20_plink2.genome", header=TRUE)
beemuse_TL_19_20_genome <- read.table("TL_19-20_plink2.genome", header=TRUE)
beemuse_unknown_genome <- read.table("Unknown_plink2.genome", header=TRUE)

```

```

# Définir la palette de couleurs pastel
pastel_colors <- c(
    "#FF0000", "#FF3300", "#FF6600", "#FF9900", "#FFCC00",
    "#FFFF00", "#CCFF00", "#99FF00", "#66FF00", "#33FF00",
    "#00FF00", "#00FF33", "#00FF66", "#00FF99", "#00FFCC",
    "#00FFFF", "#00CCFF", "#0099FF", "#0066FF", "#0033FF",
    "#0000FF", "#3300FF", "#6600FF", "#9900FF", "#CC00FF",
    "#FF00FF", "#FF00CC", "#FF0099", "#FF0066", "#FF0033"
)

# Création d'un facteur pour distinguer les différentes données
beemuse_BAH_20_19_genome$Dataset <- "BAH_20-19"
beemuse_BBS_6_19_genome$Dataset <- "BBS_6-19"
beemuse_BER_11_19_genome$Dataset <- "BER_11-19"
beemuse_BH_44_genome$Dataset <- "BH_44"
beemuse_BH_7_19_genome$Dataset <- "BH_7-19"
beemuse_BHA_2_20_genome$Dataset <- "BHA_2-20"
beemuse_BLS_53_19_genome$Dataset <- "BLS_53-19"
beemuse_ER_13_19_genome$Dataset <- "ER_13-19"
beemuse_KBJ_1_19_genome$Dataset <- "KBJ_1-19"
beemuse_KBru_6_20_genome$Dataset <- "KBru_6-20"
beemuse_KLoc_37_19_genome$Dataset <- "KLoc_37-19"
beemuse_KLSU_14_19_genome$Dataset <- "KLSU_14-19"
beemuse_MM_31_20_genome$Dataset <- "MM_31-20"
beemuse_MM_37_20_genome$Dataset <- "MM_37-20"
beemuse_MP_10_20_genome$Dataset <- "MP_10-20"
beemuse_PersoBC_2021_genome$Dataset <- "PersoBC_2021"
beemuse_PersoJLL_2021_genome$Dataset <- "PersoJLL_2021"
beemuse_PersoJLL_2022_genome$Dataset <- "PersoJLL_2022"
beemuse_PersoLD_2021_genome$Dataset <- "PersoLD_2021"
beemuse_PersoLD_2022_genome$Dataset <- "PersoLD_2022"
beemuse_PersoUB_2021_genome$Dataset <- "PersoUB_2021"
beemuse_PersoUB_2022_genome$Dataset <- "PersoUB_2022"
beemuse_S_GZ_2_19_genome$Dataset <- "S_GZ_2-19"
beemuse_SBJ_3_19_genome$Dataset <- "SBJ_3-19"
beemuse_SJ_16_20_genome$Dataset <- "SJ_16-20"
beemuse_SJ_24_20_genome$Dataset <- "SJ_24-20"
beemuse_SJ_30_20_genome$Dataset <- "SJ_30-20"
beemuse_TL_13_20_genome$Dataset <- "TL_13-20"
beemuse_TL_19_20_genome$Dataset <- "TL_19-20"

```

```

beemuse_unknown_genome$Dataset <- "Unknown"

# Combiner les ensembles de données
all_data <- rbind(beemuse_BAH_20_19_genome, beemuse_BBS_6_19_genome, beemuse_BER_11_19_genome, beemuse_BLA_19_genome, beemuse_BLS_53_19_genome, beemuse_EG_19_genome, beemuse_ER_13_19_genome, beemuse_KBJ_1_19_genome, beemuse_KBru_6_20_genome, beemuse_KLoc_37_19_genome, beemuse_KLSU_14_19_genome, beemuse_MM_31_20_genome, beemuse_MM_37_20_genome, beemuse_MP_10_20_genome, beemuse_PersoBC_2021_genome, beemuse_PersoJLL_2021_genome, beemuse_PersoJLL_2022_genome, beemuse_PersoLD_2021_genome, beemuse_PersoLD_2022_genome, beemuse_PersoUB_2021_genome, beemuse_PersoUB_2022_genome, beemuse_S_GZ_2_19_genome, beemuse_SBJ_3_19_genome, beemuse_SJ_16_20_genome, beemuse_SJ_24_20_genome, beemuse_SJ_30_20_genome, beemuse_TL_13_20_genome, beemuse_TL_19_20_genome, beemuse.Unknown_genome)

# Define the order of the groups and reverse it
group_order <- c(
  "BAH_20-19", "BBS_6-19", "BER_11-19", "BH_44", "BH_7-19", "BHA_2-20",
  "BLS_53-19", "ER_13-19", "KBJ_1-19", "KBru_6-20", "KLoc_37-19",
  "KLSU_14-19", "MM_31-20", "MM_37-20", "MP_10-20", "PersoBC_2021",
  "PersoJLL_2021", "PersoJLL_2022", "PersoLD_2021", "PersoLD_2022",
  "PersoUB_2021", "PersoUB_2022", "S_GZ_2-19", "SBJ_3-19", "SJ_16-20",
  "SJ_24-20", "SJ_30-20", "TL_13-20", "TL_19-20", "Unknown"
)
group_order <- rev(group_order)

# Convert the Dataset variable to a factor with the reversed order
all_data$Dataset <- factor(all_data$Dataset, levels = group_order)

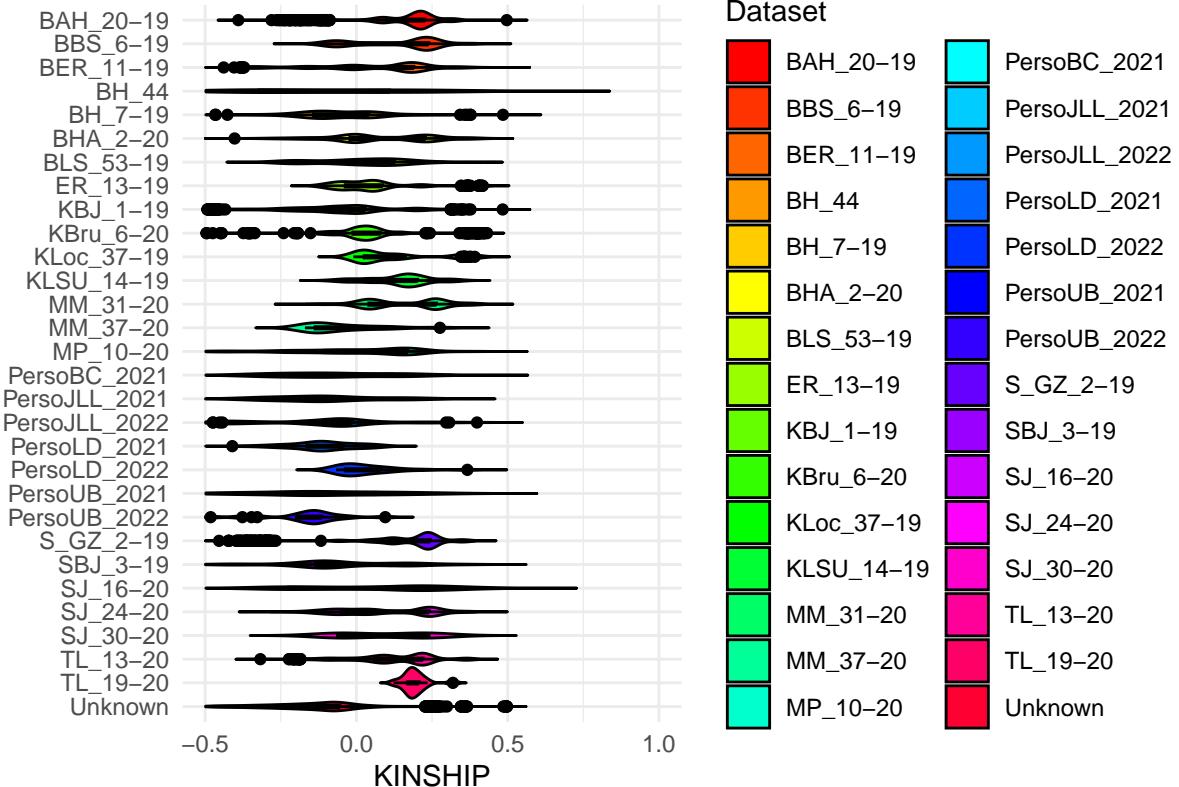
ggplot(all_data, aes(x = KINSHIP, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.3) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width = 0.5)) +
  scale_fill_manual(values = pastel.colors, limits = rev(group_order)) +
  labs(title = "Violin Plot - KINSHIP",
       x = "KINSHIP", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(-0.5, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(-0.5, 1))

## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.

## Warning: Removed 1729 rows containing non-finite values ('stat_ydensity()').
## Warning: Removed 1729 rows containing non-finite values ('stat_boxplot()').
## Warning: 'position_dodge()' requires non-overlapping x intervals
## Warning: Removed 815 rows containing missing values ('geom_violin()').

```

## Violin Plot – KINSHIP



```
ggplot(all_data, aes(x = IBS0, y = Dataset, fill = Dataset)) +
  geom_violin(trim = FALSE, color = "black", width = 1.2) +
  geom_boxplot(width = 0.1, fill = alpha("white", 0), color = "black", position = position_dodge(width =
  scale_fill_manual(values = pastel_colors, limits = rev(group_order)) +
  labs(title = "Violin Plot – IBS0",
  x = "IBS0", y = "") +
  theme_minimal() +
  scale_x_continuous(limits = c(0, 1)) +
  coord_flip() +
  coord_cartesian(xlim = c(0, 1))
```

```
## Coordinate system already present. Adding new coordinate system, which will
## replace the existing one.
```

```
## Warning: ‘position_dodge()’ requires non-overlapping x intervals
```

```
## Warning: Removed 1556 rows containing missing values ('geom_violin()').
```

### Violin Plot – IBS0

