

# DETECT CHROMOSOMAL DUPLICATIONS

Claudia Ziri3n-Mart3nez

## Setup

---

```
library(tidyverse)
library(ggtree)
library(ggtreeExtra)
library(ape)
library(ggnewscale)
library(RColorBrewer)
setwd("/FastData/czirion/Crypto_Diversity_Pipeline/analyses/tree_duplications/scripts")
```

## Metadata

---

Load the necessary data

```
metadata <- read.csv(
  "../../data/processed/metadata_ashton_desj_all_fungalpop_H99.csv",
  header = TRUE)
```

Get one dataframe for each variable to be plotted as a separate metadata column in the tree

```
lineage <- metadata %>%
  select(strain, lineage)%>%
  column_to_rownames("strain")

source <- metadata %>%
  select(strain, source)%>%
  column_to_rownames("strain")

sublineage <- metadata %>%
  select(strain, vni_subdivision)%>%
  column_to_rownames("strain")

dataset <- metadata %>%
  select(strain, dataset)%>%
  column_to_rownames("strain")
```

## Duplications

---

```
duplications <- read.delim(
  "../results/tables/duplications_polished.tsv",
  sep = "\t", header = TRUE, stringsAsFactors = TRUE)
```

```

duplications_full <- duplications %>%
  select(strain, chromosome) %>%
  distinct()

```

Make matrix of duplicated chromosomes

```

dup_chroms <- duplications_full %>%
  select(strain, chromosome)%>%
  mutate(duplicated_full = 1)%>%
  arrange(chromosome)%>%
  pivot_wider(names_from = chromosome, values_from = duplicated_full, values_fill = 0)%>%
  column_to_rownames("strain")%>%
  mutate(across(everything(), ~ ifelse(. == 1, cur_column(),"Euploid")))

euploid_strain <- metadata %>%
  filter(!strain %in% duplications_full$strain)%>%
  select(strain)

for (chrom in colnames(dup_chroms)){
  euploid_strain[chrom] <- "Euploid"
}

dup_chroms <- euploid_strain %>%
  column_to_rownames("strain") %>%
  bind_rows(dup_chroms)

```

## Tree

---

```

merged_tree_path <- "/FastData/czirion/Crypto_Diversity_Pipeline/analyses/data/processed/merged_tree.n
tree <- read.tree(merged_tree_path)

```

Remove tips that are not in metadata\$strain

```

tree <- drop.tip(tree, setdiff(tree$tip.label, metadata$strain))

```

## Plots

---

Create vectors of colors for each metadata variable with the name of the levels as names of the colors

```

dataset_colors <- c(brewer.pal(9, "Set1")[c(1, 2)], "white")
names(dataset_colors) <- levels(as.factor(dataset$dataset))

lineage_colors <- brewer.pal(8, "Dark2")[c(1, 2, 3, 4)]
names(lineage_colors) <- levels(as.factor(lineage$lineage))

sublineage_colors <- c(brewer.pal(12, "Set3")[c(1:9)])
names(sublineage_colors) <- levels(sublineage$vni_subdivision)

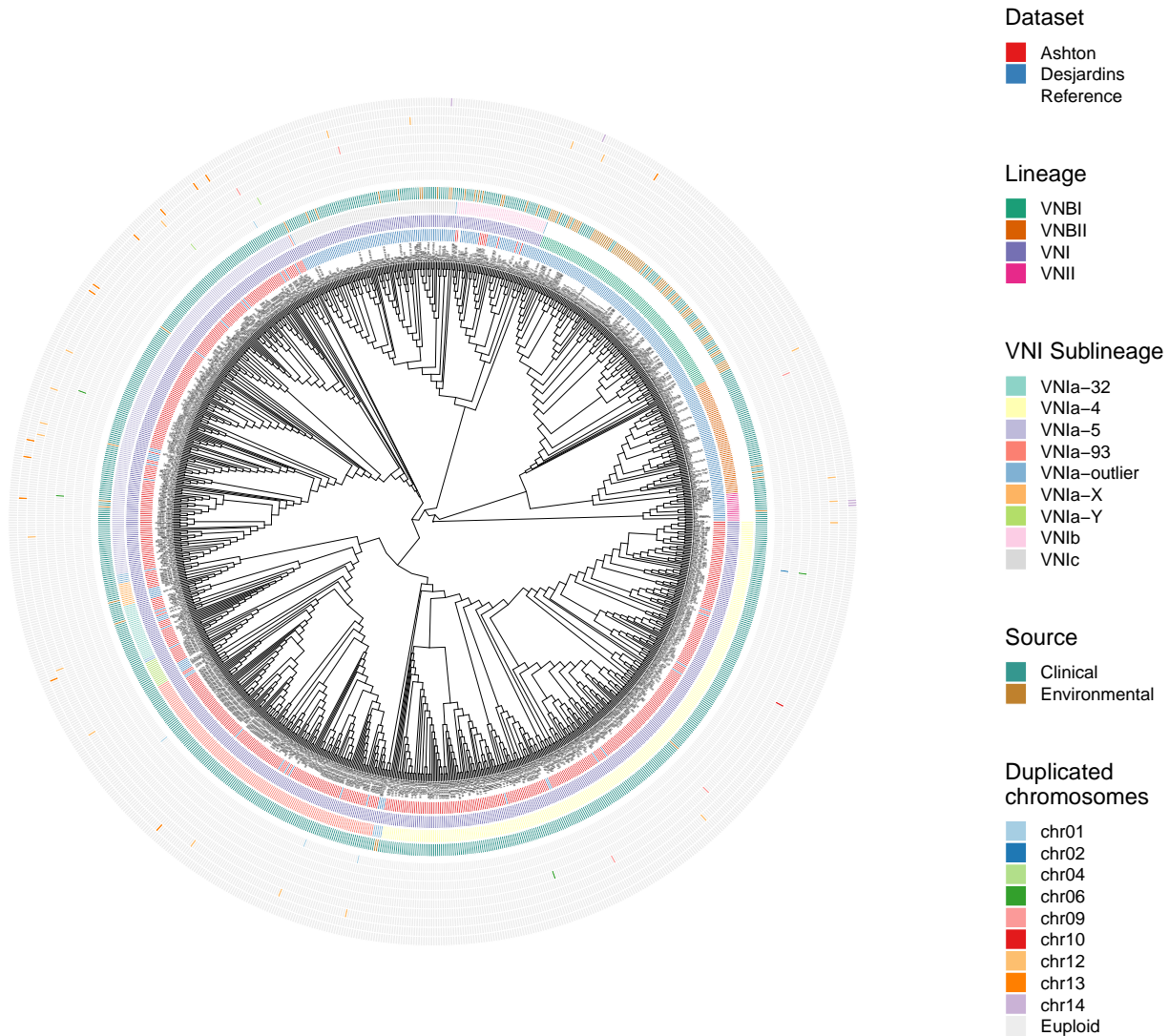
source_colors <- brewer.pal(11, "BrBG")[c(9, 3)] # 9, 3 are the colors for the two sources

```

```
names(source_colors) <- levels(as.factor(source$source))

chrom_colors <- c(brewer.pal(nlevels(duplications$chromosome), "Paired"), "grey93")
names(chrom_colors) <- c(levels(duplications$chromosome), "Euploid")
```

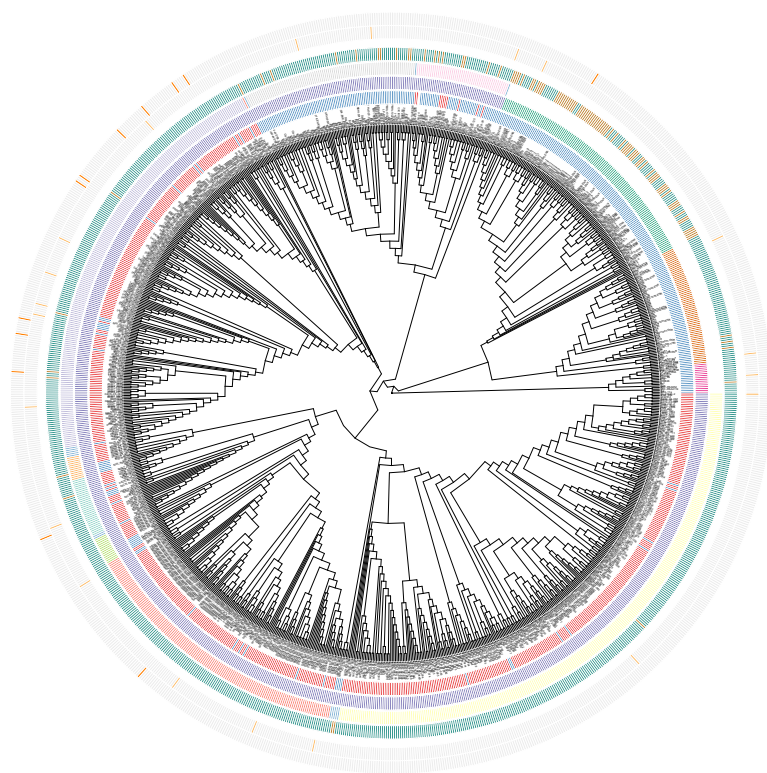
## Tree of all samples with duplications of all chromosomes



## Tree of all samples with duplications of chromosomes 12 and 13

Subset the duplications\_full data frame to only include strains with duplications of chromosomes 12 and 13

```
dup_chroms_12_13 <- dup_chroms %>%
  select(chr12, chr13)
```



#### Dataset

Ashton  
Desjardins  
Reference

#### Lineage

VNBI  
VNBII  
VNI  
VNII

#### VNI Sublineage

VNIIa-32  
VNIIa-4  
VNIIa-5  
VNIIa-93  
VNIIa-outlier  
VNIIa-X  
VNIIa-Y  
VNIIb  
VNIIc

#### Source

Clinical  
Environmental

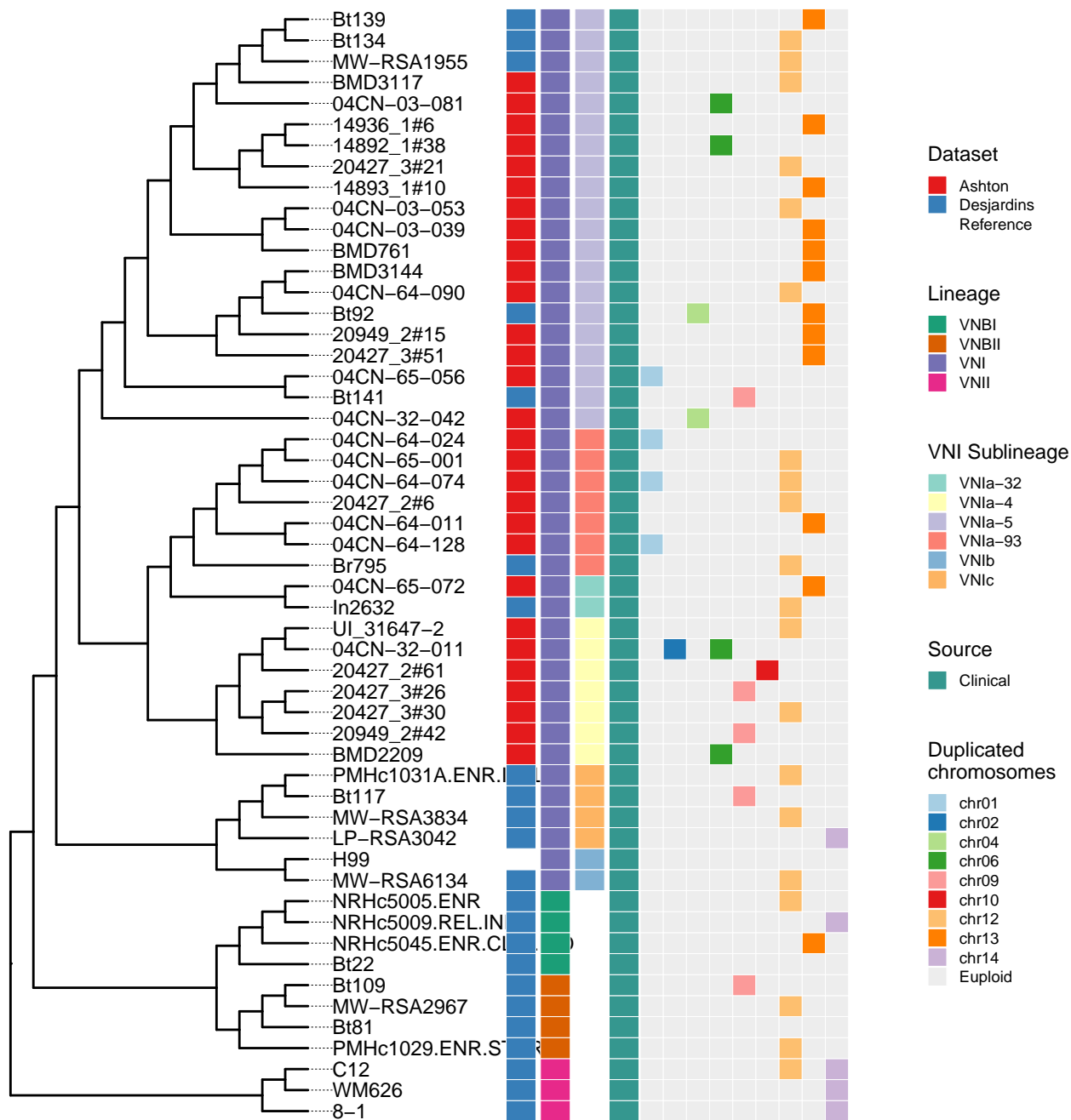
#### Duplicated chromosomes

chr12  
chr13  
Euploid

### Tree with only the samples that have duplications and the references

```
keep_strains <- c(levels(duplications_full$strain), "H99", "Bt22", "Bt81")
tree_dups <- drop.tip(tree, setdiff(tree$tip.label, keep_strains))
```

### Dataset, lineage, sublineage, source, duplications



Lineage, sublineage, duplications

