Projet de groupe final : Conservation et analyse des métadonnées génomiques du virus de la fièvre de la vallée du Rift (VFVR) en Afrique

Il s'agit d'un flux de travail bioinformatique réel pour l'extraire, conserver et analyser les métadonnées génomiques à l'aide de fichiers GenBank du virus de la fièvre de la vallée du Rift (VFVR) isolé dans des pays africains. Nous avons récupéré les données sur GenBank de VFVR à l'aide des numéros d'accès fournis, ensuite nous avons extrait ces métadonnées structurées, nettoiées et analysées avec Bash, Python et R, et produira un rapport reproductible via Quarto, RMarkdown/R Notebook ou Jupyter.

Pour repondre aux questions, voici la procedure que nous avons utilisés pour avoir des résultats satisfaisants.

1. Configuration de l'environnement

Nous avons utilisés l'environnement quarto-bio où le fichier yml a été crée et ce qui nous a permis d'installer tous ces outils afin d 'eviter les dépendances.

2. Préparation des métadonnées avec Bash

Nous avons crée un repertoire Projet_rvf_Africa


Projet_rvf_Africa


Docs : guide de lecture


results : où le script deposera un fichier des resultats


Data : Contient toutes les metatas du projet


scripts : repertoire contenant tous les scripts a exécutes

Les dossiers ont été créer en utulisant la commande:mkdir -p Docs results data scripts, les fichiers ont été copier ou deplacés en utilisant les commandes cp et mv. Pour repondre aux questions de cette partie, nous avons créer un script metadat_preparation,sh.

Ce script va nous compter les : Compter les numéros GenBank uniques, Compter les séquences complètes et partielles, Compter les segments S, L, M et enfin il va générer un fichier **analyse_resume,txt** qui va contenir tous les resultats.

fichier **analyse_resume,txt**

```
Résumé de l'analyse des métadonnées
==================================
Date : ven. 09 mai 2025 22:06:46 WAT

Nombre de numéros d'accession GenBank uniques : 1435
Nombre de séquences complètes : 833
Nombre de séquences partielles : 602
Nombre de segments S : 454
Nombre de segments L : 368
Nombre de segments M : 476
Nombre de segments S complets : 319
Nombre de segments L complets : 257
Nombre de segments M complets : 254
```

3. Téléchargement de GenBank (Python + Biopython)

Un script python **Telechargement_genebank,py** a été créé et exécuter pour répondre aux questions posées. Il va créer dans le repertoire results un dossier contenant genbank_file contenant les **fichiers ,gb** , il va créer aussi un autre dossier contenant les **fichier fasta** des segments L ains que le fichier **metadata,tsv** dans results.

4. Nettoyage, gestion et analyse des données (R tidyverse) et 5. Visualisation des données

Les résultats de ces deux parties sont contenus dans le rapport notebook en .htm dans le répertoire **results** que nous allons vous les les montrer.

R Notebook

Charger les bibliothèques nécessaires

```
library(dplyr)
```

```
##
## Attachement du package : 'dplyr'
##     intersect, setdiff, setequal, union
```

```
library(readr)
library(stringr)
```

Definir le repertoire de travail

```
setwd("/home/lnsp/Bureau/training_ghana/Travaux_pratique/rvf_africa_projet/donnees")
# Remplace par ton chemin réel


fichier <- "rvf_africa.tsv"


donnees <- read_tsv(fichier, show_col_types = FALSE)


print(colnames(donnees))
```

```
##  [1] "Species"           "GenomeStatus"      "Strain"
##  [4] "Segment"           "GenBank Accessions" "Size"
##  [7] "GC Content"        "Contig N50"        "CollectionDate"
## [10] "CollectionYear"    "IsolationCountry"  "GeographicGroup"
## [13] "HostName"          "Host Common Name"  "HostGroup"
```

Étape 1 : Retirer les entrées sans date de collecte ou pays

```
donnees_filtrees <- donnees %>%
```

```
  filter(!is.na(CollectionDate), !is.na(IsolationCountry))


# Sauvegarder les données filtrées
write_tsv(donnees_filtrees, "donnees_filtrees.tsv")
```

Étape 2 : Grouper et résumer le nombre de séquences par pays, année, segment

```
resume_pays_annee_segment <- donnees_filtrees %>%
  group_by(IsolationCountry, CollectionYear, Segment) %>%
  summarise(Nb_sequences = n(), .groups = "drop")


write_tsv(resume_pays_annee_segment, "resume_pays_annee_segment.tsv")
print(resume_pays_annee_segment)
```

```
## # A tibble: 246 × 4
##    IsolationCountry CollectionYear Segment Nb_sequences
##    <chr>                    <dbl> <chr>         <int>
##  1 Angola                    1985 M                 1
##  2 Angola                    1985 S                 1
##  3 Angola                    2016 L                 1
##  4 Angola                    2016 M                 1
##  5 Angola                    2016 S                 1
##  6 Burkina Faso              1983 L                 1
##  7 Burkina Faso              1983 M                 1
##  8 Burkina Faso              1983 S                 2
##  9 Burundi                   2022 L                 5
## 10 Burundi                   2022 M                 7
## #      236 more rows
```

Étape 3 : Créer une colonne dérivée (par exemple : région depuis le pays)

```
donnees_mutation <- donnees_filtrees %>%
  mutate(Region = case_when(
    IsolationCountry %in% c("Kenya", "Sudan", "Tanzania", "Uganda") ~ "Afrique de l'Est",
    IsolationCountry %in% c("Senegal", "Mauritania") ~ "Afrique de l'Ouest",
    TRUE ~ "Autre"
  ))
```

Etape 4: compter le nombre de sequences par hote commun

```
compte_hotes <- donnees_mutation %>%
  group_by(`Host Common Name`) %>%
  summarise(Nb_sequences = n(), .groups = "drop") %>%
  arrange(desc(Nb_sequences))


write_tsv(compte_hotes, "compte_hotes.tsv")
print(compte_hotes)
```

```
## # A tibble: 16 × 2
##    `Host Common Name`      Nb_sequences
##    <chr>                          <int>
##  1 Human                            769
##  2 Cow                              243
##  3 Mosquito                         131
##  4 <NA>                             130
##  5 Sheep                             70
##  6 Goat                              18
##  7 Buffalo                            9
##  8 Bat                                7
```

```
##  9 Null                         6
## 10 Cattle                       5
## 11 Camel                        3
## 12 Mouse                        3
## 13 Tick                         3
## 14 Lab host                     1
## 15 Northern House Mosquito      1
## 16 Sand fly                     1
```

Étape 5 : Filtrer par pays, années et segments spécifiques (exemple : Kenya, 2007, segment L)

```
filtres_specifiques <- donnees_mutation %>%
  filter(IsolationCountry == "Kenya", CollectionYear == 2007, Segment == "L")


print(filtres_specifiques)
```

```
## # A tibble: 44 × 16
##    Species   GenomeStatus Strain Segment `GenBank Accessions`  Size `GC Content`
##    <chr>     <chr>        <chr>  <chr>   <chr>                 <dbl>       <dbl>
##  1 Phlebovi… Partial      K2     L       PP746417               6386        43.8
##  2 Phlebovi… Partial      J8     L       PP746426               6251        43.6
##  3 Phlebovi… Partial      J10    L       PP746425               6232        43.5
##  4 Phlebovi… Partial      J9     L       PP746427               6277        43.5
##  5 Phlebovi… Partial      KEM-JC L       PP746437               6268        43.5
##  6 Phlebovi… Partial      KEM-BR L       PP746436               6232        43.5
##  7 Phlebovi… Partial      KEM-ND L       PP746438               6232        43.4
##  8 Phlebovi… Partial      MO-LE  L       PP746448               6387        43.5
##  9 Phlebovi… Partial      HA-HAR L       PP746447               6373        43.9
```

```
## 10 Phlebovi… Partial      MSA    L       PP746449            6388        43.4
## #      34 more rows
## #      9 more variables: `Contig N50` <dbl>, CollectionDate <chr>,
## #   CollectionYear <dbl>, IsolationCountry <chr>, GeographicGroup <chr>,
## #   HostName <chr>, `Host Common Name` <chr>, HostGroup <chr>, Region <chr>
```

Étape 6 : Renommer la colonne CollectionYear en year

```
donnees_renommee <- donnees_mutation %>%
  rename(year = CollectionYear)
```

Étape 7 : Grouper et résumer par pays, année et segment

```
resume_final <- donnees_renommee %>%
  group_by(IsolationCountry, year, Segment) %>%
  summarise(Nb_sequences = n(), .groups = "drop")


write_tsv(resume_final, "resume_final.tsv")
print(resume_final)
## # A tibble: 246 × 4
##    IsolationCountry  year Segment Nb_sequences
##    <chr>            <dbl> <chr>         <int>
##  1 Angola            1985 M                 1
##  2 Angola            1985 S                 1
##  3 Angola            2016 L                 1
##  4 Angola            2016 M                 1
##  5 Angola            2016 S                 1
##  6 Burkina Faso      1983 L                 1
##  7 Burkina Faso      1983 M                 1
##  8 Burkina Faso      1983 S                 2
```

```
##  9 Burundi          2022 L                5
## 10 Burundi          2022 M                7
## #       236 more rows
```

Étape 8 : Identifier les pays avec le plus grand nombre de séquences complètes (indépendamment du segment)

```
pays_plus_sequences <- donnees_filtrees %>%
  filter(GenomeStatus == "Complete") %>%
  group_by(IsolationCountry) %>%
  summarise(Nb_sequences_completes = n(), .groups = "drop") %>%
  arrange(desc(Nb_sequences_completes))


write_tsv(pays_plus_sequences, "pays_plus_sequences.tsv")
print(pays_plus_sequences)
```

```
## # A tibble: 19 × 2
##    IsolationCountry        Nb_sequences_completes
##    <chr>                                    <int>
##  1 South Africa                               389
##  2 Kenya                                      178
##  3 Madagascar                                  38
##  4 Zimbabwe                                    35
##  5 Central African Republic                    33
##  6 Egypt                                      33
##  7 Uganda                                      28
##  8 Sudan                                      22
##  9 Mauritania                                  15
## 10 Namibia                                     12
```

```
## 11 Tanzania                    12
## 12 Guinea                       7
## 13 Senegal                      7
## 14 Angola                       4
## 15 Burkina Faso                 4
## 16 Gabon                        4
## 17 Mayotte                      2
## 18 Somalia                      1
## 19 Zambia                       1
```

Étape 9 : Compter les hôtes les plus fréquents (top 10)

```
top_hotes <- donnees_filtrees %>%
  group_by(`Host Common Name`) %>%
  summarise(Nb_sequences = n(), .groups = "drop") %>%
  arrange(desc(Nb_sequences)) %>%
  slice_head(n = 10)


write_tsv(top_hotes, "top_hotes.tsv")
print(top_hotes)
```

```
## # A tibble: 10 × 2
##    `Host Common Name` Nb_sequences
##    <chr>                    <int>
##  1 Human                      769
##  2 Cow                        243
##  3 Mosquito                   131
##  4 <NA>                       130
##  5 Sheep                       70
```

```
##  6 Goat                  18
##  7 Buffalo                9
##  8 Bat                    7
##  9 Null                   6
## 10 Cattle                 5
```

```
cat("     Analyse terminée. Tous les fichiers résumés ont été sauvegardés dans le répertoire : ", getwd(), "\n")
```

```
##      Analyse terminée. Tous les fichiers résumés ont été sauvegardés dans le répertoire :  /home/lnsp/Bureau/training_ghana/Travaux_pratique/rvf_africa_projet/resultats
```

Telechargement de librairie

```
library(dplyr)
library(readr)
# repertoire de travail


setwd("/home/lnsp/Bureau/training_ghana/Travaux_pratique/rvf_africa_projet/resultats")


donnees <- read_tsv(fichier, show_col_types = FALSE)


print(colnames(donnees))
```

```
##  [1] "Species"           "GenomeStatus"      "Strain"
##  [4] "Segment"           "GenBank Accessions" "Size"
##  [7] "GC Content"        "Contig N50"        "CollectionDate"
## [10] "CollectionYear"    "IsolationCountry"  "GeographicGroup"
## [13] "HostName"          "Host Common Name"  "HostGroup"
```

Nombre d’isolats par pays

```
isolats_par_pays <- donnees %>%
```

```
  group_by(IsolationCountry) %>%
  summarise(Nombre_isolats = n(), .groups = "drop") %>%
  arrange(desc(Nombre_isolats))
print("=== Nombre d'isolats par pays ===")
```

```
## [1] "=== Nombre d'isolats par pays ==="
```

```
print(isolats_par_pays)
```

```
## # A tibble: 25 × 2
##    IsolationCountry        Nombre_isolats
##    <chr>                            <int>
##  1 South Africa                       434
##  2 Kenya                              260
##  3 Madagascar                         198
##  4 Mauritania                         118
##  5 Uganda                             111
##  6 Senegal                             52
##  7 Egypt                               51
##  8 Zimbabwe                            51
##  9 Central African Republic            37
## 10 Sudan                               23
## #      15 more rows
```

Sauvegarder le résumé

```
write_tsv(isolats_par_pays, "nombre_isolats_par_pays.tsv")
```

Nombre d’isolats par an

```
isolats_par_an <- donnees %>%
  group_by(CollectionYear) %>%
  summarise(Nombre_isolats = n(), .groups = "drop") %>%
```

```
  arrange(CollectionYear)

print("=== Nombre d'isolats par an ===")
```

```
## [1] "=== Nombre d'isolats par an ==="
```

```
print(isolats_par_an)
```

```
## # A tibble: 54 × 2
##    CollectionYear Nombre_isolats
##             <dbl>          <int>
##  1           1944             10
##  2           1951              8
##  3           1955             24
##  4           1956              4
##  5           1962              5
##  6           1963              2
##  7           1964              1
##  8           1965              1
##  9           1969             11
## 10           1970              5
## #       44 more rows
```

```
write_tsv(isolats_par_an, "nombre_isolats_par_an.tsv")
```

Répartition des segments (S, M, L) par pays

```
repartition_segments <- donnees %>%
  group_by(IsolationCountry, Segment) %>%
  summarise(Nombre = n(), .groups = "drop") %>%
  arrange(IsolationCountry, Segment)


print("=== Répartition des segments par pays ===")
```

```
## [1] "=== Répartition des segments par pays ==="
```

```
print(repartition_segments)
```

```
## # A tibble: 67 × 3
##    IsolationCountry        Segment Nombre
##    <chr>                   <chr>    <int>
##  1 Angola                  L            1
##  2 Angola                  M            2
##  3 Angola                  S            2
##  4 Burkina Faso            L            1
##  5 Burkina Faso            M            1
##  6 Burkina Faso            S            2
##  7 Burundi                 L            5
##  8 Burundi                 M            7
##  9 Burundi                 <NA>         5
## 10 Central African Republic L           11
## #      57 more rows
```

```
# Sauvegarder le résumé
write_tsv(repartition_segments, "repartition_segments_par_pays.tsv")


cat("     Analyse descriptive terminée. Résultats sauvegardés dans le dossier : ",
    getwd(), "\n")
```

```
##      Analyse descriptive terminée. Résultats sauvegardés dans le dossier :
/home/lnsp/Bureau/training_ghana/Travaux_pratique/rvf_africa_projet/resultats
```

Assurez-vous que les colonnes sont bien de type approprié

```
donnees <- donnees %>%
  mutate(CollectionYear = as.integer(CollectionYear),
```

```
        CollectionDate = as.Date(CollectionDate, format = "%Y-%m-%d"))
```

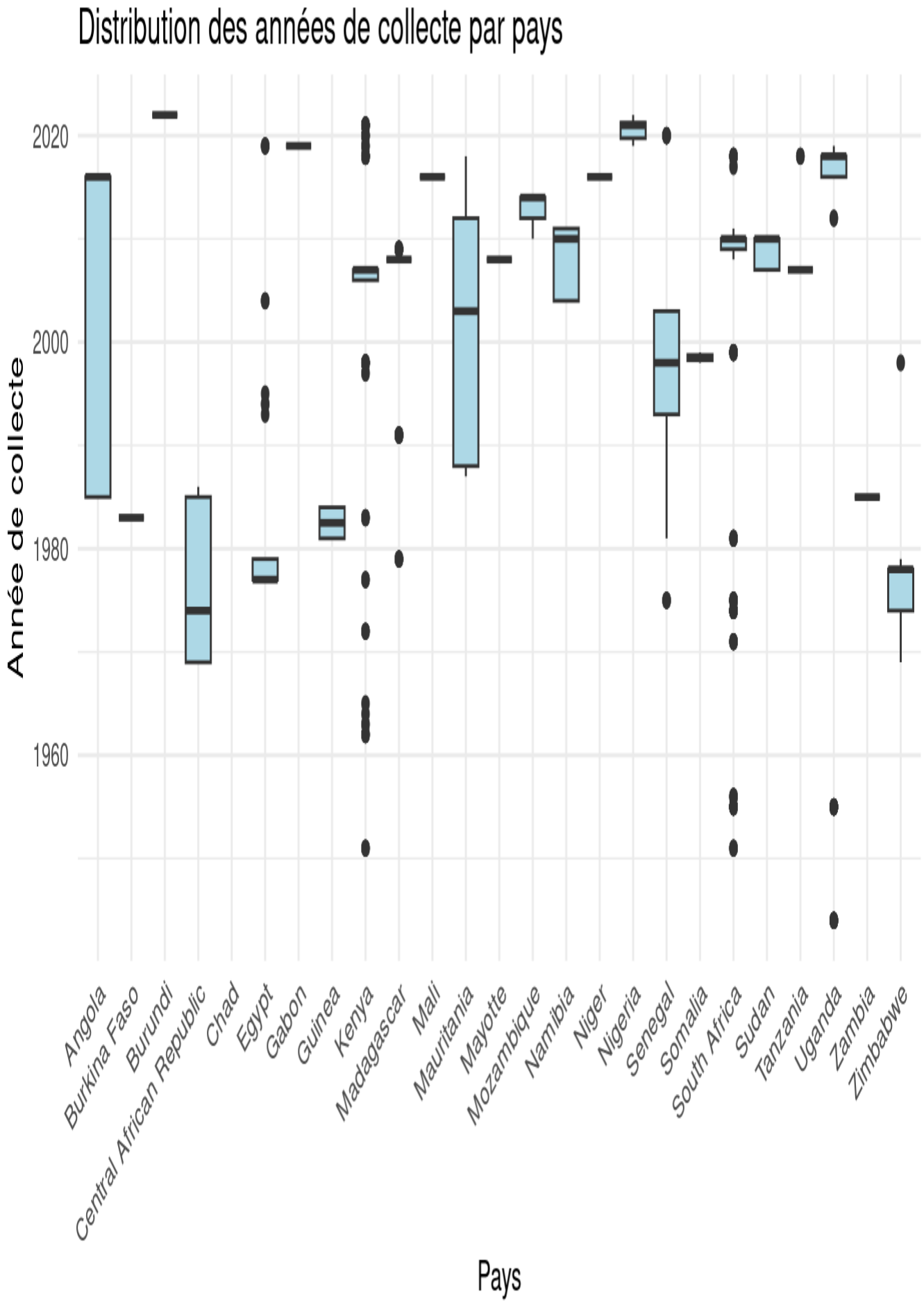
Visualisation des données

Boxplot : distribution des dates de collecte par pays

```
library(ggplot2)


ggplot(donnees, aes(x = IsolationCountry, y = CollectionYear)) +
  geom_boxplot(fill = "lightblue") +
  labs(title = "Distribution des années de collecte par pays",
       x = "Pays",
       y = "Année de collecte") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```
## Warning: Removed 35 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```


Distribution des années de collecte par pays
Année de collecte
2020
2000
1980
1960
Angola
Burkina Faso
Burundi
Central African Republic
Chad
Egypt
Gabon
Guinea
Kenya
Madagascar
Mali
Mauritania
Mayotte
Mozambique
Namibia
Niger
Nigeria
Senegal
Somalia
South Africa
Sudan
Tanzania
Uganda
Zambia
Zimbabwe
Pays

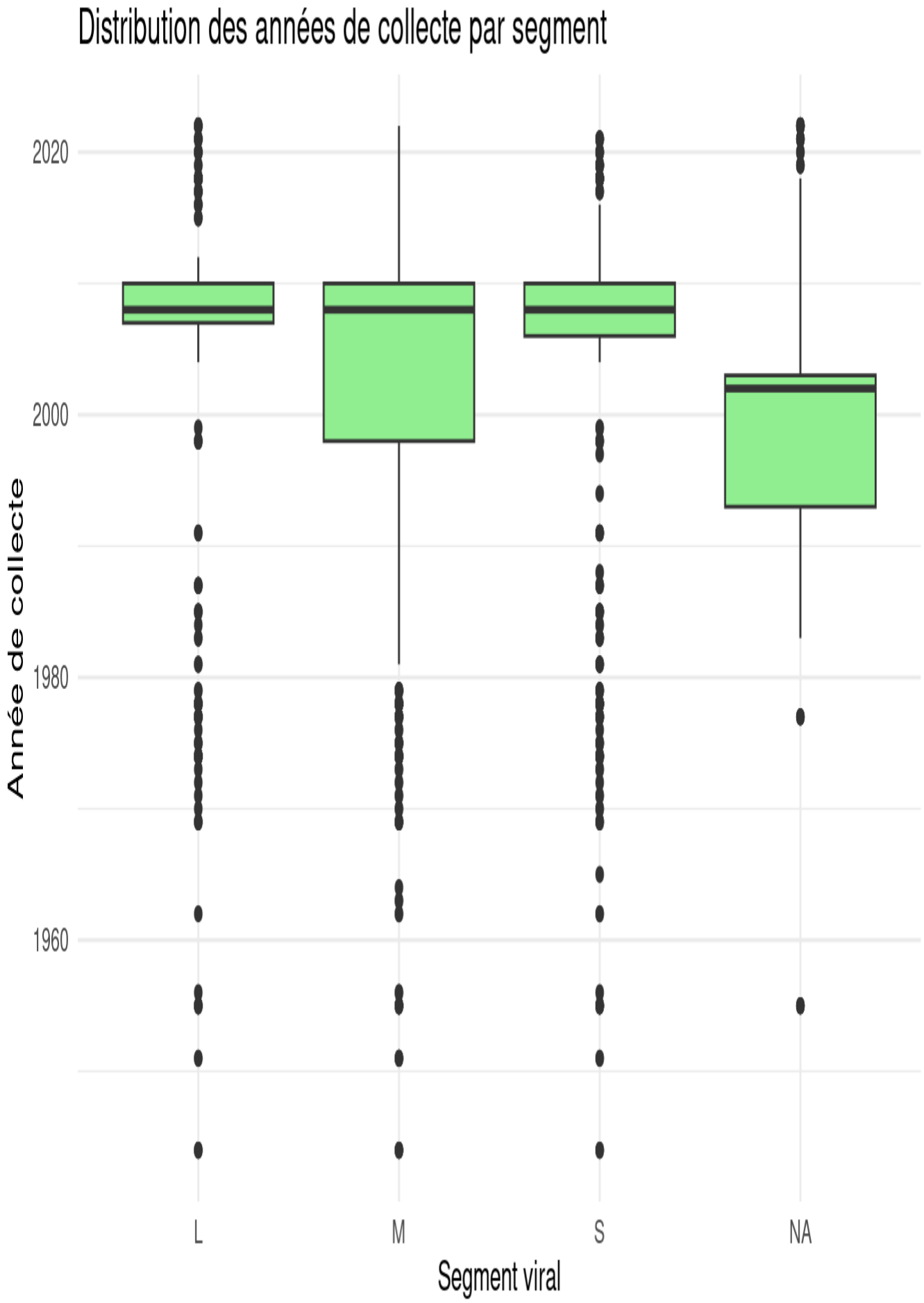
```
ggsave("boxplot_annee_par_pays.png", width = 8, height = 5)
```

```
## Warning: Removed 35 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```

Boxplot : distribution des années de collecte par segment

```
ggplot(donnees, aes(x = Segment, y = CollectionYear)) +
  geom_boxplot(fill = "lightgreen") +
  labs(title = "Distribution des années de collecte par segment",
       x = "Segment viral",
       y = "Année de collecte") +
  theme_minimal()
```

```
## Warning: Removed 35 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```


Distribution des années de collecte par segment
Année de collecte
2020
2000
1980
1960
L
M
S
NA
Segment viral

```
ggsave("boxplot_annee_par_segment.png", width = 6, height = 5)
```

```
## Warning: Removed 35 rows containing non-finite outside the scale range
## (`stat_boxplot()`).
```

Résumé : nombre d’isolats par pays et année

```
resume_pays_annee <- donnees %>%
  group_by(IsolationCountry, CollectionYear) %>%
  summarise(Nb_isolats = n(), .groups = "drop") %>%
  arrange(desc(Nb_isolats))


print("Résumé des isolats par pays et année :")
```

```
## [1] "Résumé des isolats par pays et année :"
```

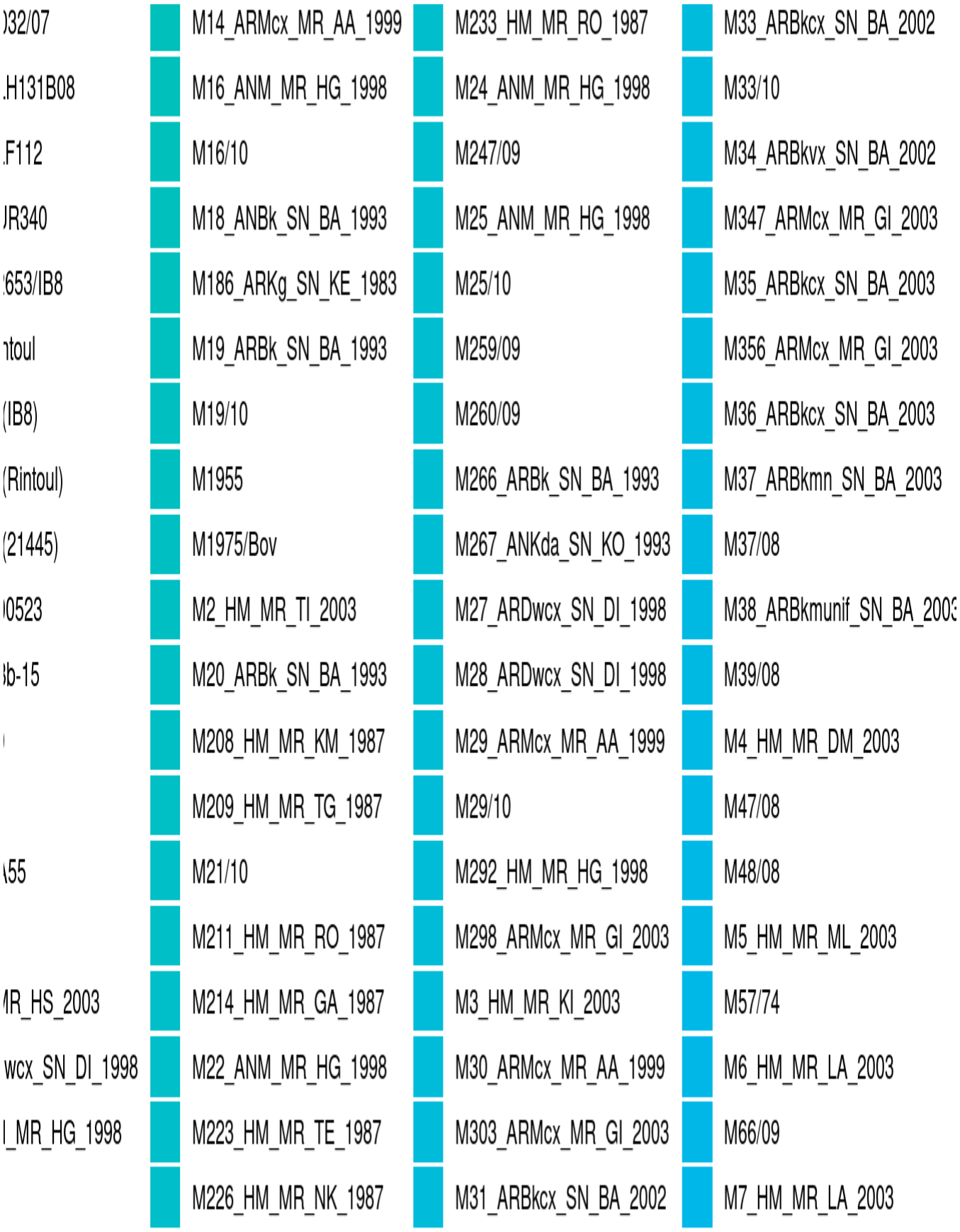
```
print(resume_pays_annee)
```

```
## # A tibble: 113 × 3
##    IsolationCountry CollectionYear Nb_isolats
##    <chr>                    <int>      <int>
##  1 South Africa              2010        279
##  2 Madagascar                2008        171
##  3 Kenya                     2007        152
##  4 Uganda                    2018         49
##  5 South Africa              2008         35
##  6 Kenya                     2006         27
##  7 Mauritania                1987         27
##  8 South Africa              2011         27
##  9 Zimbabwe                  1978         27
## 10 South Africa              2009         26
## #      103 more rows
```

```
write_tsv(resume_pays_annee, "resume_isolats_par_pays_annee.tsv")
```

Barplot : distribution des souches par pays

```
ggplot(donnees, aes(x = IsolationCountry, fill = Strain)) +
  geom_bar(position = "dodge") +
  labs(title = "Distribution des souches par pays",
       x = "Pays",
       y = "Nombre d’isolats",
       fill = "Souche (Strain)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```


)32/07
.H131B08
.F112
)R340
:653/IB8
ıtoul
(IB8)
(Rintoul)
(21445)
)0523
)b-15
\55
MR_HS_2003
wcx_SN_DI_1998
I_MR_HG_1998
M14_ARMcx_MR_AA_1999
M16_ANM_MR_HG_1998
M16/10
M18_ANBk_SN_BA_1993
M186_ARKg_SN_KE_1983
M19_ARBk_SN_BA_1993
M19/10
M1955
M1975/Bov
M2_HM_MR_TI_2003
M20_ARBk_SN_BA_1993
M208_HM_MR_KM_1987
M209_HM_MR_TG_1987
M21/10
M211_HM_MR_RO_1987
M214_HM_MR_GA_1987
M22_ANM_MR_HG_1998
M223_HM_MR_TE_1987
M226_HM_MR_NK_1987
M233_HM_MR_RO_1987
M24_ANM_MR_HG_1998
M247/09
M25_ANM_MR_HG_1998
M25/10
M259/09
M260/09
M266_ARBk_SN_BA_1993
M267_ANKda_SN_KO_1993
M27_ARDwcx_SN_DI_1998
M28_ARDwcx_SN_DI_1998
M29_ARMcx_MR_AA_1999
M29/10
M292_HM_MR_HG_1998
M298_ARMcx_MR_GI_2003
M3_HM_MR_KI_2003
M30_ARMcx_MR_AA_1999
M303_ARMcx_MR_GI_2003
M31_ARBkcx_SN_BA_2002
M33_ARBkcx_SN_BA_2002
M33/10
M34_ARBkvx_SN_BA_2002
M347_ARMcx_MR_GI_2003
M35_ARBkcx_SN_BA_2003
M356_ARMcx_MR_GI_2003
M36_ARBkcx_SN_BA_2003
M37_ARBkmn_SN_BA_2003
M37/08
M38_ARBkmunif_SN_BA_2003
M39/08
M4_HM_MR_DM_2003
M47/08
M48/08
M5_HM_MR_ML_2003
M57/74
M6_HM_MR_LA_2003
M66/09
M7_HM_MR_LA_2003

```r
ggsave("barplot_souches_par_pays.png", width = 8, height = 5)


cat("      Graphiques générés et sauvegardés dans : ", getwd(), "\n")
```

```
##       Graphiques générés et sauvegardés dans :
/home/lnsp/Bureau/training_ghana/Travaux_pratique/rvf_africa_projet/resultats
```