West Africa

Bbosa Robert

6/17/2020

Table of Contents

# Install the necessary packages

library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.0 v purrr 0.3.4  
## v tibble 3.0.1 v dplyr 0.8.5  
## v tidyr 1.1.0 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(plotly)

##   
## Attaching package: 'plotly'

## The following object is masked from 'package:ggplot2':  
##   
## last\_plot

## The following object is masked from 'package:stats':  
##   
## filter

## The following object is masked from 'package:graphics':  
##   
## layout

library(ggplot2)  
library(dplyr)  
library(viridis)

## Loading required package: viridisLite

library(patchwork)  
library(ggpubr)  
library(hrbrthemes)

## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.

## Please use hrbrthemes::import\_roboto\_condensed() to install Roboto Condensed and

## if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow

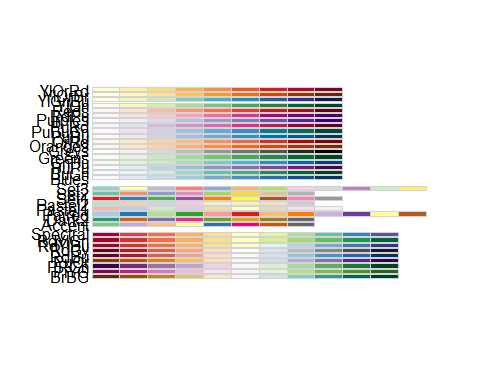
library(directlabels)  
library(knitr)  
library(pastecs)

##   
## Attaching package: 'pastecs'

## The following objects are masked from 'package:dplyr':  
##   
## first, last

## The following object is masked from 'package:tidyr':  
##   
## extract

library (RColorBrewer)  
display.brewer.all()



# Read the file

WA\_global <- read.csv("~/COVID-19/COVID-19\_Data/coronavirus.csv",   
 header = TRUE, stringsAsFactors = FALSE)  
attach(WA\_global)  
summary(WA\_global)

## Province.State Country.Region Lat Long   
## Length:115395 Length:115395 Min. :-51.796 Min. :-135.00   
## Class :character Class :character 1st Qu.: 6.611 1st Qu.: -14.45   
## Mode :character Mode :character Median : 22.300 Median : 21.76   
## Mean : 20.714 Mean : 24.35   
## 3rd Qu.: 40.143 3rd Qu.: 81.00   
## Max. : 71.707 Max. : 178.06   
## date cases type   
## Length:115395 Min. :-10034 Length:115395   
## Class :character 1st Qu.: 0 Class :character   
## Mode :character Median : 0 Mode :character   
## Mean : 109   
## 3rd Qu.: 3   
## Max. : 94305

names(WA\_global)

## [1] "Province.State" "Country.Region" "Lat" "Long"   
## [5] "date" "cases" "type"

str(WA\_global)

## 'data.frame': 115395 obs. of 7 variables:  
## $ Province.State: chr "" "" "" "" ...  
## $ Country.Region: chr "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...  
## $ Lat : num 33 33 33 33 33 33 33 33 33 33 ...  
## $ Long : num 65 65 65 65 65 65 65 65 65 65 ...  
## $ date : chr "2020-01-22" "2020-01-23" "2020-01-24" "2020-01-25" ...  
## $ cases : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ type : chr "confirmed" "confirmed" "confirmed" "confirmed" ...

# Eliminate the unnecessary columns

WA\_global <- select(WA\_global, -Province.State, -Lat, -Long)  
class(WA\_global)

## [1] "data.frame"

head(WA\_global)

## Country.Region date cases type  
## 1 Afghanistan 2020-01-22 0 confirmed  
## 2 Afghanistan 2020-01-23 0 confirmed  
## 3 Afghanistan 2020-01-24 0 confirmed  
## 4 Afghanistan 2020-01-25 0 confirmed  
## 5 Afghanistan 2020-01-26 0 confirmed  
## 6 Afghanistan 2020-01-27 0 confirmed

# Transfor date from “Character format to date format”

WA\_global <- rename(WA\_global, country = Country.Region)  
myformat <- "%Y-%m-%d"  
WA\_global$date <- as.Date(WA\_global$date, myformat)  
str(WA\_global)

## 'data.frame': 115395 obs. of 4 variables:  
## $ country: chr "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...  
## $ date : Date, format: "2020-01-22" "2020-01-23" ...  
## $ cases : int 0 0 0 0 0 0 0 0 0 0 ...  
## $ type : chr "confirmed" "confirmed" "confirmed" "confirmed" ...

# Check if the date has changed

WA\_global <- tbl\_df(WA\_global)  
class(WA\_global)

## [1] "tbl\_df" "tbl" "data.frame"

str(WA\_global)

## tibble [115,395 x 4] (S3: tbl\_df/tbl/data.frame)  
## $ country: chr [1:115395] "Afghanistan" "Afghanistan" "Afghanistan" "Afghanistan" ...  
## $ date : Date[1:115395], format: "2020-01-22" "2020-01-23" ...  
## $ cases : int [1:115395] 0 0 0 0 0 0 0 0 0 0 ...  
## $ type : chr [1:115395] "confirmed" "confirmed" "confirmed" "confirmed" ...

# Include a column for totals

WA\_global\_total <- WA\_global %>%  
 group\_by(country, type ) %>%  
 mutate(total\_cases = cumsum(cases))  
head(WA\_global\_total)

## # A tibble: 6 x 5  
## # Groups: country, type [1]  
## country date cases type total\_cases  
## <chr> <date> <int> <chr> <int>  
## 1 Afghanistan 2020-01-22 0 confirmed 0  
## 2 Afghanistan 2020-01-23 0 confirmed 0  
## 3 Afghanistan 2020-01-24 0 confirmed 0  
## 4 Afghanistan 2020-01-25 0 confirmed 0  
## 5 Afghanistan 2020-01-26 0 confirmed 0  
## 6 Afghanistan 2020-01-27 0 confirmed 0

tail(WA\_global\_total)

## # A tibble: 6 x 5  
## # Groups: country, type [1]  
## country date cases type total\_cases  
## <chr> <date> <int> <chr> <int>  
## 1 China 2020-06-11 0 recovered 79493  
## 2 China 2020-06-12 0 recovered 79493  
## 3 China 2020-06-13 0 recovered 79493  
## 4 China 2020-06-14 0 recovered 79493  
## 5 China 2020-06-15 0 recovered 79493  
## 6 China 2020-06-16 0 recovered 79493

# Select Southern Africa countries

WA\_countries <- WA\_global\_total %>%   
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin", "Gambia", "Mauritania", "Senegal","Cote d'Ivoire","Sierra Leone","Mali","Liberia"))  
   
head(WA\_countries)

## # A tibble: 6 x 5  
## # Groups: country, type [1]  
## country date cases type total\_cases  
## <chr> <date> <int> <chr> <int>  
## 1 Benin 2020-01-22 0 confirmed 0  
## 2 Benin 2020-01-23 0 confirmed 0  
## 3 Benin 2020-01-24 0 confirmed 0  
## 4 Benin 2020-01-25 0 confirmed 0  
## 5 Benin 2020-01-26 0 confirmed 0  
## 6 Benin 2020-01-27 0 confirmed 0

tail(WA\_countries)

## # A tibble: 6 x 5  
## # Groups: country, type [1]  
## country date cases type total\_cases  
## <chr> <date> <int> <chr> <int>  
## 1 Togo 2020-06-11 6 recovered 271  
## 2 Togo 2020-06-12 8 recovered 279  
## 3 Togo 2020-06-13 12 recovered 291  
## 4 Togo 2020-06-14 0 recovered 291  
## 5 Togo 2020-06-15 8 recovered 299  
## 6 Togo 2020-06-16 45 recovered 344

# Compute the infection rate

confirmed\_cases\_WA <- WA\_countries %>%   
 filter(type=="confirmed") %>%  
 group\_by(country,type) %>%  
 mutate(infection\_rate=ifelse(country=="Ghana",  
 total\_cases\*(1000000/31072940),  
 ifelse(country=="Nigeria",  
 total\_cases\*(1000000/206139589),  
 ifelse(country=="Niger",   
 total\_cases\*(1000000/24206644),  
 ifelse(country=="Cabo Verde",  
 total\_cases\*(1000000/555987),  
 ifelse(country=="Guinea-Bissau",  
 total\_cases\*(1000000/1968001),  
 ifelse(country=="Guinea",  
 total\_cases\*(1000000/13132795),  
 ifelse(country=="Burkina Faso",  
 total\_cases\*(1000000/20903273),  
 ifelse(country=="Togo",  
 total\_cases\*(1000000/8278724),  
 ifelse(country=="Benin",  
 total\_cases\*(1000000/12123200),  
 ifelse(country=="Gambia",  
 total\_cases\*(1000000/2416668),  
 ifelse(country=="Mauritania",   
 total\_cases\*(1000000/4649658),  
 ifelse(country=="Senegal",  
 total\_cases\*(1000000/16743927),  
 ifelse(country=="Cote d'Ivoire",  
 total\_cases\*(1000000/26378274),  
 ifelse(country=="Mali",  
 total\_cases\*(1000000/20250833),  
 ifelse(country=="Sierra Leona",  
 total\_cases\*(1000000/7976983),   
 total\_cases\*(1000000/5057681)))))))))))))))))   
head(confirmed\_cases\_WA)

## # A tibble: 6 x 6  
## # Groups: country, type [1]  
## country date cases type total\_cases infection\_rate  
## <chr> <date> <int> <chr> <int> <dbl>  
## 1 Benin 2020-01-22 0 confirmed 0 0  
## 2 Benin 2020-01-23 0 confirmed 0 0  
## 3 Benin 2020-01-24 0 confirmed 0 0  
## 4 Benin 2020-01-25 0 confirmed 0 0  
## 5 Benin 2020-01-26 0 confirmed 0 0  
## 6 Benin 2020-01-27 0 confirmed 0 0

tail(confirmed\_cases\_WA)

## # A tibble: 6 x 6  
## # Groups: country, type [1]  
## country date cases type total\_cases infection\_rate  
## <chr> <date> <int> <chr> <int> <dbl>  
## 1 Togo 2020-06-11 2 confirmed 524 63.3  
## 2 Togo 2020-06-12 1 confirmed 525 63.4  
## 3 Togo 2020-06-13 5 confirmed 530 64.0  
## 4 Togo 2020-06-14 0 confirmed 530 64.0  
## 5 Togo 2020-06-15 1 confirmed 531 64.1  
## 6 Togo 2020-06-16 6 confirmed 537 64.9

str(confirmed\_cases\_WA)

## tibble [2,352 x 6] (S3: grouped\_df/tbl\_df/tbl/data.frame)  
## $ country : chr [1:2352] "Benin" "Benin" "Benin" "Benin" ...  
## $ date : Date[1:2352], format: "2020-01-22" "2020-01-23" ...  
## $ cases : int [1:2352] 0 0 0 0 0 0 0 0 0 0 ...  
## $ type : chr [1:2352] "confirmed" "confirmed" "confirmed" "confirmed" ...  
## $ total\_cases : int [1:2352] 0 0 0 0 0 0 0 0 0 0 ...  
## $ infection\_rate: num [1:2352] 0 0 0 0 0 0 0 0 0 0 ...  
## - attr(\*, "groups")= tibble [16 x 3] (S3: tbl\_df/tbl/data.frame)  
## ..$ country: chr [1:16] "Benin" "Burkina Faso" "Cabo Verde" "Cote d'Ivoire" ...  
## ..$ type : chr [1:16] "confirmed" "confirmed" "confirmed" "confirmed" ...  
## ..$ .rows :List of 16  
## .. ..$ : int [1:147] 1 2 3 4 5 6 7 8 9 10 ...  
## .. ..$ : int [1:147] 148 149 150 151 152 153 154 155 156 157 ...  
## .. ..$ : int [1:147] 295 296 297 298 299 300 301 302 303 304 ...  
## .. ..$ : int [1:147] 442 443 444 445 446 447 448 449 450 451 ...  
## .. ..$ : int [1:147] 589 590 591 592 593 594 595 596 597 598 ...  
## .. ..$ : int [1:147] 736 737 738 739 740 741 742 743 744 745 ...  
## .. ..$ : int [1:147] 883 884 885 886 887 888 889 890 891 892 ...  
## .. ..$ : int [1:147] 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 ...  
## .. ..$ : int [1:147] 1177 1178 1179 1180 1181 1182 1183 1184 1185 1186 ...  
## .. ..$ : int [1:147] 1324 1325 1326 1327 1328 1329 1330 1331 1332 1333 ...  
## .. ..$ : int [1:147] 1471 1472 1473 1474 1475 1476 1477 1478 1479 1480 ...  
## .. ..$ : int [1:147] 1618 1619 1620 1621 1622 1623 1624 1625 1626 1627 ...  
## .. ..$ : int [1:147] 1765 1766 1767 1768 1769 1770 1771 1772 1773 1774 ...  
## .. ..$ : int [1:147] 1912 1913 1914 1915 1916 1917 1918 1919 1920 1921 ...  
## .. ..$ : int [1:147] 2059 2060 2061 2062 2063 2064 2065 2066 2067 2068 ...  
## .. ..$ : int [1:147] 2206 2207 2208 2209 2210 2211 2212 2213 2214 2215 ...  
## ..- attr(\*, ".drop")= logi TRUE

summary(confirmed\_cases\_WA$infection\_rate)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.000 0.000 2.847 55.279 38.066 1404.709

stat.desc(confirmed\_cases\_WA)

## country date cases type total\_cases infection\_rate  
## nbr.val NA 2.352000e+03 2352.000000 NA 2.352000e+03 2.352000e+03  
## nbr.null NA 0.000000e+00 1301.000000 NA 8.470000e+02 8.470000e+02  
## nbr.na NA 0.000000e+00 0.000000 NA 0.000000e+00 0.000000e+00  
## min NA 1.828300e+04 -209.000000 NA 0.000000e+00 0.000000e+00  
## max NA 1.842900e+04 921.000000 NA 1.714800e+04 1.404709e+03  
## range NA 1.460000e+02 1130.000000 NA 1.714800e+04 1.404709e+03  
## sum NA 4.317331e+07 55869.000000 NA 1.587934e+06 1.300164e+05  
## median NA 1.835600e+04 0.000000 NA 2.000000e+01 2.847007e+00  
## mean NA 1.835600e+04 23.753827 NA 6.751420e+02 5.527908e+01  
## SE.mean NA 8.751658e-01 1.441429 NA 3.699862e+01 2.973054e+00  
## CI.mean NA 1.716177e+00 2.826604 NA 7.255331e+01 5.830081e+00  
## var NA 1.801433e+03 4886.791351 NA 3.219648e+06 2.078945e+04  
## std.dev NA 4.244329e+01 69.905589 NA 1.794338e+03 1.441855e+02  
## coef.var NA 2.312230e-03 2.942919 NA 2.657719e+00 2.608319e+00

# Fatality rate

death\_cases\_WA <- WA\_countries %>%   
 filter(type %in% c("death", "confirmed")) %>%   
 select(-total\_cases) %>%   
 spread(type, cases) %>%   
 mutate(fatality\_rate=ifelse(country=="Ghana",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Nigeria",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Niger",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Cabo Verde",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Guinea-Bissau",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Guinea",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Bukina Faso",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Togo",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Benin",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Gambia",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Mauritania",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Senegal",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Cote d'Ivoire",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Mali",   
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Sierra Leona",  
 round((cumsum(death)/cumsum(confirmed))\*100,digits=2),  
 round((cumsum(death)/cumsum(confirmed))\*100,  
 digits=2)))))))))))))))))  
  
head(death\_cases\_WA)

## # A tibble: 6 x 5  
## # Groups: country [1]  
## country date confirmed death fatality\_rate  
## <chr> <date> <int> <int> <dbl>  
## 1 Benin 2020-01-22 0 0 NaN  
## 2 Benin 2020-01-23 0 0 NaN  
## 3 Benin 2020-01-24 0 0 NaN  
## 4 Benin 2020-01-25 0 0 NaN  
## 5 Benin 2020-01-26 0 0 NaN  
## 6 Benin 2020-01-27 0 0 NaN

tail(death\_cases\_WA)

## # A tibble: 6 x 5  
## # Groups: country [1]  
## country date confirmed death fatality\_rate  
## <chr> <date> <int> <int> <dbl>  
## 1 Togo 2020-06-11 2 0 2.48  
## 2 Togo 2020-06-12 1 0 2.48  
## 3 Togo 2020-06-13 5 0 2.45  
## 4 Togo 2020-06-14 0 0 2.45  
## 5 Togo 2020-06-15 1 0 2.45  
## 6 Togo 2020-06-16 6 0 2.42

str(death\_cases\_WA)

## tibble [2,352 x 5] (S3: grouped\_df/tbl\_df/tbl/data.frame)  
## $ country : chr [1:2352] "Benin" "Benin" "Benin" "Benin" ...  
## $ date : Date[1:2352], format: "2020-01-22" "2020-01-23" ...  
## $ confirmed : int [1:2352] 0 0 0 0 0 0 0 0 0 0 ...  
## $ death : int [1:2352] 0 0 0 0 0 0 0 0 0 0 ...  
## $ fatality\_rate: num [1:2352] NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...  
## - attr(\*, "groups")= tibble [16 x 2] (S3: tbl\_df/tbl/data.frame)  
## ..$ country: chr [1:16] "Benin" "Burkina Faso" "Cabo Verde" "Cote d'Ivoire" ...  
## ..$ .rows :List of 16  
## .. ..$ : int [1:147] 1 2 3 4 5 6 7 8 9 10 ...  
## .. ..$ : int [1:147] 148 149 150 151 152 153 154 155 156 157 ...  
## .. ..$ : int [1:147] 295 296 297 298 299 300 301 302 303 304 ...  
## .. ..$ : int [1:147] 442 443 444 445 446 447 448 449 450 451 ...  
## .. ..$ : int [1:147] 589 590 591 592 593 594 595 596 597 598 ...  
## .. ..$ : int [1:147] 736 737 738 739 740 741 742 743 744 745 ...  
## .. ..$ : int [1:147] 883 884 885 886 887 888 889 890 891 892 ...  
## .. ..$ : int [1:147] 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 ...  
## .. ..$ : int [1:147] 1177 1178 1179 1180 1181 1182 1183 1184 1185 1186 ...  
## .. ..$ : int [1:147] 1324 1325 1326 1327 1328 1329 1330 1331 1332 1333 ...  
## .. ..$ : int [1:147] 1471 1472 1473 1474 1475 1476 1477 1478 1479 1480 ...  
## .. ..$ : int [1:147] 1618 1619 1620 1621 1622 1623 1624 1625 1626 1627 ...  
## .. ..$ : int [1:147] 1765 1766 1767 1768 1769 1770 1771 1772 1773 1774 ...  
## .. ..$ : int [1:147] 1912 1913 1914 1915 1916 1917 1918 1919 1920 1921 ...  
## .. ..$ : int [1:147] 2059 2060 2061 2062 2063 2064 2065 2066 2067 2068 ...  
## .. ..$ : int [1:147] 2206 2207 2208 2209 2210 2211 2212 2213 2214 2215 ...  
## ..- attr(\*, ".drop")= logi TRUE

stat.desc(death\_cases\_WA)

## country date confirmed death fatality\_rate  
## nbr.val NA 2.352000e+03 2352.000000 2.352000e+03 1505.0000000  
## nbr.null NA 0.000000e+00 1301.000000 1.934000e+03 271.0000000  
## nbr.na NA 0.000000e+00 0.000000 0.000000e+00 847.0000000  
## min NA 1.828300e+04 -209.000000 0.000000e+00 0.0000000  
## max NA 1.842900e+04 921.000000 3.100000e+01 50.0000000  
## range NA 1.460000e+02 1130.000000 3.100000e+01 50.0000000  
## sum NA 4.317331e+07 55869.000000 1.100000e+03 5834.8700000  
## median NA 1.835600e+04 0.000000 0.000000e+00 2.2700000  
## mean NA 1.835600e+04 23.753827 4.676871e-01 3.8769900  
## SE.mean NA 8.751658e-01 1.441429 3.361857e-02 0.1271401  
## CI.mean NA 1.716177e+00 2.826604 6.592512e-02 0.2493906  
## var NA 1.801433e+03 4886.791351 2.658249e+00 24.3277161  
## std.dev NA 4.244329e+01 69.905589 1.630414e+00 4.9323135  
## coef.var NA 2.312230e-03 2.942919 3.486121e+00 1.2722017

# Recovery rate

recovered\_cases\_WA <- WA\_countries %>%   
 filter(type %in% c("recovered","confirmed")) %>%   
 select(-total\_cases) %>%   
 spread(type, cases) %>%  
 mutate(recovery\_rate=ifelse(country=="Angola",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Nigeria",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Niger",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Cabo Verde",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Guinea-Bissau",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Guinea",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Bukina Faso",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Togo",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Benin",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Gambia",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Mauritania",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Senegal",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Cote d'Ivoire",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Mali",   
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 ifelse(country=="Sierra Leona",  
 round((cumsum(recovered)/cumsum(confirmed))\*100,digits=2),  
 round((cumsum(recovered)/cumsum(confirmed))\*100,  
 digits=2)))))))))))))))))  
  
head(recovered\_cases\_WA)

## # A tibble: 6 x 5  
## # Groups: country [1]  
## country date confirmed recovered recovery\_rate  
## <chr> <date> <int> <int> <dbl>  
## 1 Benin 2020-01-22 0 0 NaN  
## 2 Benin 2020-01-23 0 0 NaN  
## 3 Benin 2020-01-24 0 0 NaN  
## 4 Benin 2020-01-25 0 0 NaN  
## 5 Benin 2020-01-26 0 0 NaN  
## 6 Benin 2020-01-27 0 0 NaN

tail(recovered\_cases\_WA)

## # A tibble: 6 x 5  
## # Groups: country [1]  
## country date confirmed recovered recovery\_rate  
## <chr> <date> <int> <int> <dbl>  
## 1 Togo 2020-06-11 2 6 51.7  
## 2 Togo 2020-06-12 1 8 53.1  
## 3 Togo 2020-06-13 5 12 54.9  
## 4 Togo 2020-06-14 0 0 54.9  
## 5 Togo 2020-06-15 1 8 56.3  
## 6 Togo 2020-06-16 6 45 64.1

str(recovered\_cases\_WA)

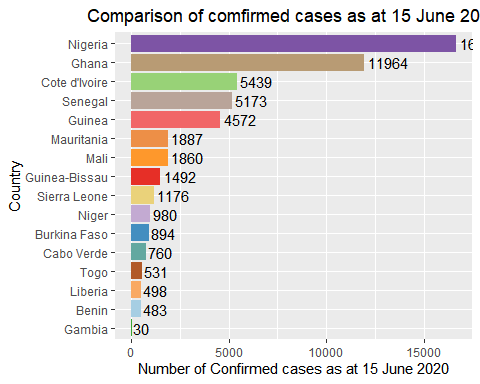
## tibble [2,352 x 5] (S3: grouped\_df/tbl\_df/tbl/data.frame)  
## $ country : chr [1:2352] "Benin" "Benin" "Benin" "Benin" ...  
## $ date : Date[1:2352], format: "2020-01-22" "2020-01-23" ...  
## $ confirmed : int [1:2352] 0 0 0 0 0 0 0 0 0 0 ...  
## $ recovered : int [1:2352] 0 0 0 0 0 0 0 0 0 0 ...  
## $ recovery\_rate: num [1:2352] NaN NaN NaN NaN NaN NaN NaN NaN NaN NaN ...  
## - attr(\*, "groups")= tibble [16 x 2] (S3: tbl\_df/tbl/data.frame)  
## ..$ country: chr [1:16] "Benin" "Burkina Faso" "Cabo Verde" "Cote d'Ivoire" ...  
## ..$ .rows :List of 16  
## .. ..$ : int [1:147] 1 2 3 4 5 6 7 8 9 10 ...  
## .. ..$ : int [1:147] 148 149 150 151 152 153 154 155 156 157 ...  
## .. ..$ : int [1:147] 295 296 297 298 299 300 301 302 303 304 ...  
## .. ..$ : int [1:147] 442 443 444 445 446 447 448 449 450 451 ...  
## .. ..$ : int [1:147] 589 590 591 592 593 594 595 596 597 598 ...  
## .. ..$ : int [1:147] 736 737 738 739 740 741 742 743 744 745 ...  
## .. ..$ : int [1:147] 883 884 885 886 887 888 889 890 891 892 ...  
## .. ..$ : int [1:147] 1030 1031 1032 1033 1034 1035 1036 1037 1038 1039 ...  
## .. ..$ : int [1:147] 1177 1178 1179 1180 1181 1182 1183 1184 1185 1186 ...  
## .. ..$ : int [1:147] 1324 1325 1326 1327 1328 1329 1330 1331 1332 1333 ...  
## .. ..$ : int [1:147] 1471 1472 1473 1474 1475 1476 1477 1478 1479 1480 ...  
## .. ..$ : int [1:147] 1618 1619 1620 1621 1622 1623 1624 1625 1626 1627 ...  
## .. ..$ : int [1:147] 1765 1766 1767 1768 1769 1770 1771 1772 1773 1774 ...  
## .. ..$ : int [1:147] 1912 1913 1914 1915 1916 1917 1918 1919 1920 1921 ...  
## .. ..$ : int [1:147] 2059 2060 2061 2062 2063 2064 2065 2066 2067 2068 ...  
## .. ..$ : int [1:147] 2206 2207 2208 2209 2210 2211 2212 2213 2214 2215 ...  
## ..- attr(\*, ".drop")= logi TRUE

stat.desc(recovered\_cases\_WA)

## country date confirmed recovered recovery\_rate  
## nbr.val NA 2.352000e+03 2352.000000 2352.000000 1.505000e+03  
## nbr.null NA 0.000000e+00 1301.000000 1570.000000 2.400000e+02  
## nbr.na NA 0.000000e+00 0.000000 0.000000 8.470000e+02  
## min NA 1.828300e+04 -209.000000 -31.000000 0.000000e+00  
## max NA 1.842900e+04 921.000000 786.000000 9.031000e+01  
## range NA 1.460000e+02 1130.000000 817.000000 9.031000e+01  
## sum NA 4.317331e+07 55869.000000 24667.000000 4.744249e+04  
## median NA 1.835600e+04 0.000000 0.000000 2.857000e+01  
## mean NA 1.835600e+04 23.753827 10.487670 3.152325e+01  
## SE.mean NA 8.751658e-01 1.441429 0.721344 6.727476e-01  
## CI.mean NA 1.716177e+00 2.826604 1.414537 1.319623e+00  
## var NA 1.801433e+03 4886.791351 1223.833110 6.811469e+02  
## std.dev NA 4.244329e+01 69.905589 34.983326 2.609879e+01  
## coef.var NA 2.312230e-03 2.942919 3.335662 8.279220e-01

# Plot a bar chart

nb.colors <- 16  
mycolors <- colorRampPalette(brewer.pal(12, "Paired"))(nb.colors)  
  
WA\_countries %>%   
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"), type=="confirmed", date=="2020-06-15" ) %>%   
 ggplot( aes(country, total\_cases, fill=country)) +  
 geom\_bar(aes(reorder(country, total\_cases),total\_cases),  
 stat= "identity", show.legend = FALSE) +  
 geom\_text(aes(label= total\_cases), vjust=0.5, hjust=-0.1, colour="black") +  
 scale\_fill\_manual(values = mycolors) +  
 coord\_flip() +  
 ggtitle("Comparison of comfirmed cases as at 15 June 2020") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 xlab("Country")+  
 ylab("Number of Confirmed cases as at 15 June 2020")

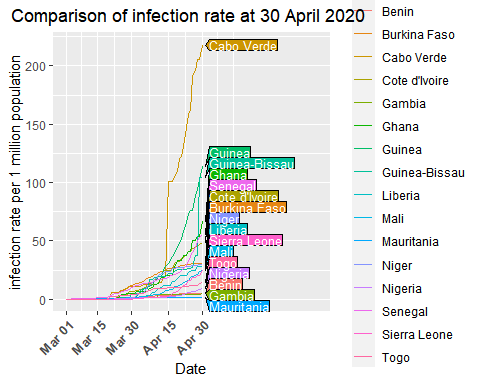


ggsave("confirmed\_WA1.png",  
 width =30, height = 15, units = "cm", dpi = 70)

# Plot a line graph as at 30 April

confirmed\_cases\_WA %>%  
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea","Guinea-Bissau",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"),date<"2020-05-01") %>%  
 ggplot( aes(x=date, y=infection\_rate, group=country, color=country)) +  
 geom\_line() +  
 geom\_dl(aes(label = country),   
 method = list(dl.trans(x = x + 0.1), "last.polygons")) +  
 scale\_fill\_manual(values = mycolors) +  
 ggtitle("Comparison of infection rate at 30 April 2020 ") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 ylab("infection rate per 1 million population")+  
 xlab("Date") +  
 scale\_x\_date(limits = as.Date(c("2020-03-01","2020-06-20")),  
 date\_labels = ("%b %d"),  
 breaks = as.Date(c("2020-03-01","2020-03-15","2020-03-30",  
 "2020-04-15","2020-04-30"))) +  
 theme(axis.text.x = element\_text(angle = 45, vjust = 1, hjust = 1,   
 size = 10, face = "bold"))

## Warning: Removed 624 row(s) containing missing values (geom\_path).



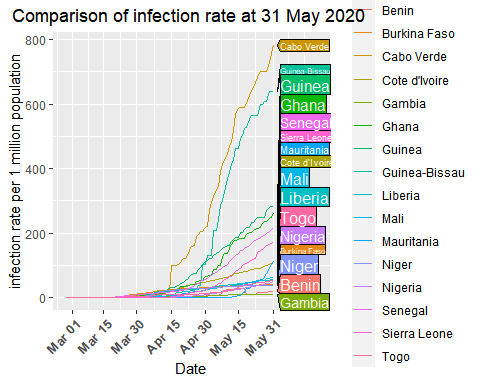
ggsave("confirmed\_WA2.png", #other supported files;- "eps", "ps", "tex" # (pictex),"pdf","jpeg", "tiff","bmp", "svg" or "wmf".  
width =30, height = 15,  
units = "cm", # other options c("in", "cm", "mm")  
dpi = 70)

## Warning: Removed 624 row(s) containing missing values (geom\_path).

# Plot a line graph as at 31 May

confirmed\_cases\_WA %>%  
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea","Guinea-Bissau",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"),date<"2020-06-01") %>%  
 ggplot( aes(x=date, y=infection\_rate, group=country, color=country)) +  
 geom\_line() +  
 geom\_dl(aes(label = country),   
 method = list(dl.trans(x = x + 0.1), "last.polygons")) +  
 scale\_fill\_manual(values = mycolors) +  
 ggtitle("Comparison of infection rate at 31 May 2020 ") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 ylab("infection rate per 1 million population")+  
 xlab("Date") +  
 scale\_x\_date(limits = as.Date(c("2020-02-27","2020-06-20")),  
 date\_labels = ("%b %d"),  
 breaks = as.Date(c("2020-02-15","2020-03-01","2020-03-15","2020-03-30",  
 "2020-04-15","2020-04-30","2020-05-15",  
 "2020-05-31"))) +  
 theme(axis.text.x = element\_text(angle = 45, vjust = 1, hjust = 1,   
 size = 10, face = "bold"))

## Warning: Removed 576 row(s) containing missing values (geom\_path).



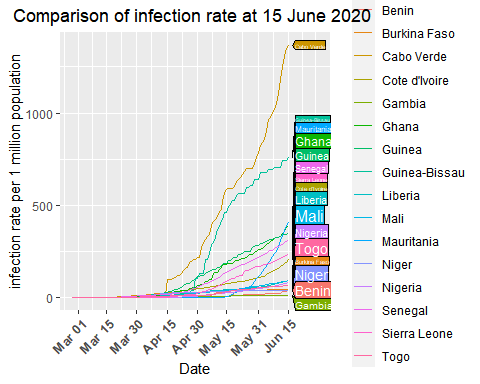
ggsave("confirmed\_WA3.png", #other supported files;- "eps", "ps", "tex" # (pictex),"pdf","jpeg", "tiff","bmp", "svg" or "wmf".  
width =30, height = 15,  
units = "cm", # other options c("in", "cm", "mm")  
dpi = 70)

## Warning: Removed 576 row(s) containing missing values (geom\_path).

# Plot a line graph as at 15 June

confirmed\_cases\_WA %>%  
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea","Guinea-Bissau",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"),date<"2020-06-16") %>%  
 ggplot( aes(x=date, y=infection\_rate, group=country, color=country)) +  
 geom\_line() +  
 geom\_dl(aes(label = country),   
 method = list(dl.trans(x = x + 0.1), "last.polygons")) +  
 scale\_fill\_manual(values = mycolors) +  
 ggtitle("Comparison of infection rate at 15 June 2020 ") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 ylab("infection rate per 1 million population")+  
 xlab("Date") +  
 scale\_x\_date(limits = as.Date(c("2020-02-27","2020-06-30")),  
 date\_labels = ("%b %d"),  
 breaks = as.Date(c("2020-02-15","2020-03-01","2020-03-15","2020-03-30",  
 "2020-04-15","2020-04-30","2020-05-15",  
 "2020-05-31","2020-06-15"))) +  
 theme(axis.text.x = element\_text(angle = 45, vjust = 1, hjust = 1,   
 size = 10, face = "bold"))

## Warning: Removed 576 row(s) containing missing values (geom\_path).

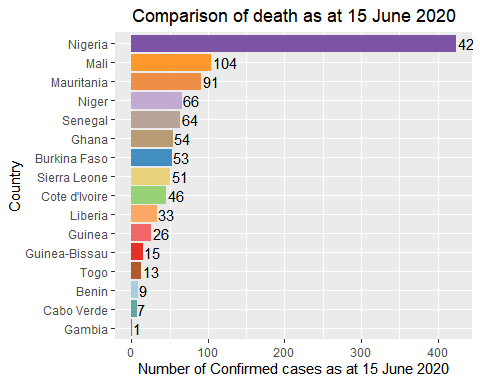


ggsave("confirmed\_WA4.png", #other supported files;- "eps", "ps", "tex" # (pictex),"pdf","jpeg", "tiff","bmp", "svg" or "wmf".  
width =30, height = 15,  
units = "cm", # other options c("in", "cm", "mm")  
dpi = 70)

## Warning: Removed 576 row(s) containing missing values (geom\_path).

# Plot a bar graph for the deaths

WA\_countries %>%   
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"), type=="death", date=="2020-06-15" ) %>%   
 ggplot( aes(country, total\_cases, fill=country)) +  
 geom\_bar(aes(reorder(country, total\_cases),total\_cases),  
 stat= "identity", show.legend = FALSE) +  
 geom\_text(aes(label= total\_cases), vjust=0.5, hjust=-0.1, colour="black") +  
 scale\_fill\_manual(values = mycolors) +  
 coord\_flip() +  
 ggtitle("Comparison of death as at 15 June 2020") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 xlab("Country")+  
 ylab("Number of Confirmed cases as at 15 June 2020")

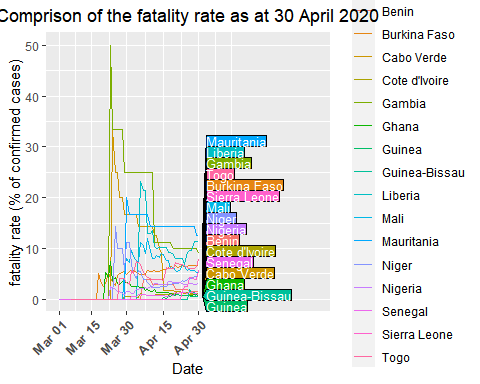


ggsave("death\_WA1.png",  
 width =30, height = 15, units = "cm", dpi = 70)

# Plot line graph for fatality rate as at 30 April

death\_cases\_WA %>%  
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"),date<"2020-05-01") %>%  
 ggplot( aes(x=date, y=fatality\_rate, group=country, color=country)) +  
 geom\_line() +  
 geom\_dl(aes(label = country),  
 method = list(dl.trans(x = x + 0.1), "last.polygons")) +  
 scale\_fill\_manual(values = mycolors) +  
 ggtitle("Comprison of the fatality rate as at 30 April 2020") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 ylab("fatality rate (% of confirmed cases)") +  
 xlab("Date") +  
 scale\_y\_continuous(limits = c(ymin= 0,   
 ymax= max(death\_cases\_WA$fatality\_rate)))+  
 scale\_x\_date(limits = as.Date(c("2020-03-01","2020-06-20")),  
 date\_labels = ("%b %d"),  
 breaks = as.Date(c("2020-03-01","2020-03-15","2020-03-30",  
 "2020-04-15","2020-04-30"))) +  
 theme(axis.text.x = element\_text(angle = 45, vjust = 1,   
 hjust = 1, size = 10, face = "bold"))

## Warning: Removed 849 row(s) containing missing values (geom\_path).



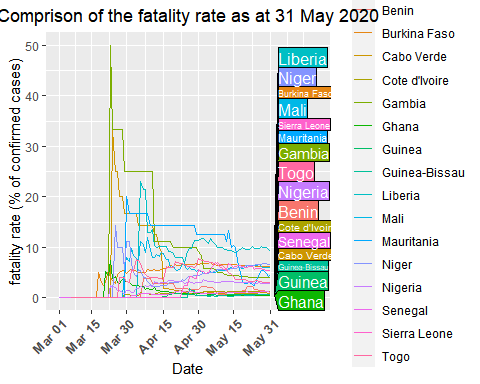
ggsave("death\_WA2.png",  
 width =30, height = 15,units = "cm",dpi = 70)

## Warning: Removed 849 row(s) containing missing values (geom\_path).

# Plot line graph for fatality rate as at 31 May

death\_cases\_WA %>%  
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"),date<"2020-06-01") %>%  
 ggplot( aes(x=date, y=fatality\_rate, group=country, color=country)) +  
 geom\_line() +  
 geom\_dl(aes(label = country),  
 method = list(dl.trans(x = x + 0.1), "last.polygons")) +  
 scale\_fill\_manual(values = mycolors) +  
 ggtitle("Comprison of the fatality rate as at 31 May 2020") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 ylab("fatality rate (% of confirmed cases)") +  
 xlab("Date") +  
 scale\_y\_continuous(limits = c(ymin= 0,   
 ymax= max(death\_cases\_WA$fatality\_rate)))+  
 scale\_x\_date(limits = as.Date(c("2020-03-01","2020-06-20")),  
 date\_labels = ("%b %d"),  
 breaks = as.Date(c("2020-03-01","2020-03-15","2020-03-30",  
 "2020-04-15","2020-04-30","2020-05-15",  
 "2020-05-31"))) +  
 theme(axis.text.x = element\_text(angle = 45, vjust = 1,   
 hjust = 1, size = 10, face = "bold"))

## Warning: Removed 849 row(s) containing missing values (geom\_path).



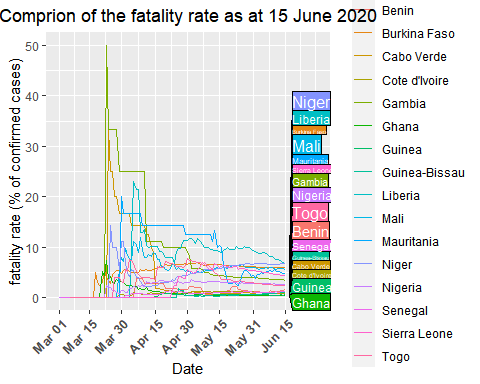
ggsave("death\_WA3.png",  
 width =30, height = 15,units = "cm",dpi = 70)

## Warning: Removed 849 row(s) containing missing values (geom\_path).

# Plot line graph for fatality rate as at 15 June

death\_cases\_WA %>%  
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"),date<"2020-06-16") %>%  
 ggplot( aes(x=date, y=fatality\_rate, group=country, color=country)) +  
 geom\_line() +  
 geom\_dl(aes(label = country),  
 method = list(dl.trans(x = x + 0.1), "last.polygons")) +  
 scale\_fill\_manual(values = mycolors) +  
 ggtitle("Comprion of the fatality rate as at 15 June 2020") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 ylab("fatality rate (% of confirmed cases)") +  
 xlab("Date") +  
 scale\_y\_continuous(limits = c(ymin= 0,   
 ymax= max(death\_cases\_WA$fatality\_rate)))+  
 scale\_x\_date(limits = as.Date(c("2020-03-01","2020-06-30")),  
 date\_labels = ("%b %d"),  
 breaks = as.Date(c("2020-03-01","2020-03-15","2020-03-30",  
 "2020-04-15","2020-04-30","2020-05-15",  
 "2020-05-31","2020-06-15"))) +  
 theme(axis.text.x = element\_text(angle = 45, vjust = 1,   
 hjust = 1, size = 10, face = "bold"))

## Warning: Removed 849 row(s) containing missing values (geom\_path).

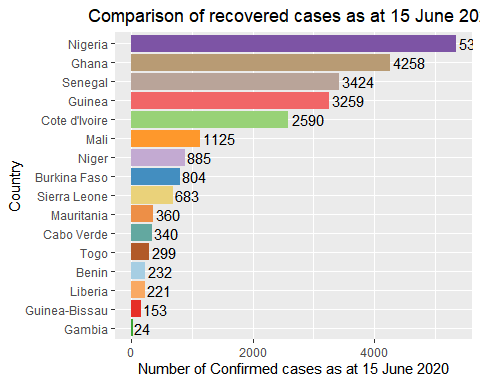


ggsave("death\_WA4.png",  
 width =30, height = 15,units = "cm",dpi = 70)

## Warning: Removed 849 row(s) containing missing values (geom\_path).

# Plot the bar chart for recoveries

WA\_countries %>%   
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"), type=="recovered", date=="2020-06-15" ) %>%   
 ggplot( aes(country, total\_cases, fill=country)) +  
 geom\_bar(aes(reorder(country, total\_cases),total\_cases),  
 stat= "identity", show.legend = FALSE) +  
 geom\_text(aes(label= total\_cases), vjust=0.5, hjust=-0.1, colour="black") +  
 scale\_fill\_manual(values = mycolors) +  
 coord\_flip() +  
 ggtitle("Comparison of recovered cases as at 15 June 2020") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 xlab("Country")+  
 ylab("Number of Confirmed cases as at 15 June 2020")

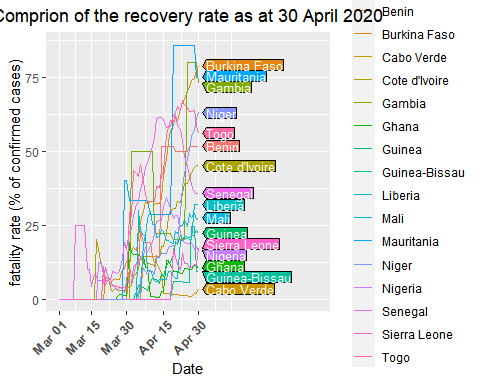


ggsave("recovered\_WA1.png",  
 width =30, height = 15, units = "cm", dpi = 70)

# Plot a line graph for recovery rate as at 30 April

recovered\_cases\_WA %>%  
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"),date<"2020-05-01") %>%  
 ggplot( aes(x=date, y=recovery\_rate, group=country, color=country)) +  
 geom\_line() +  
 geom\_dl(aes(label = country),  
 method = list(dl.trans(x = x + 0.1), "last.polygons")) +  
 scale\_fill\_manual(values = mycolors) +  
 ggtitle("Comprion of the recovery rate as at 30 April 2020") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 ylab("fatality rate (% of confirmed cases)") +  
 xlab("Date") +  
 scale\_y\_continuous(limits = c(ymin= 0,   
 ymax= max(recovered\_cases\_WA$recovery\_rate)))+  
 scale\_x\_date(limits = as.Date(c("2020-03-01","2020-06-20")),  
 date\_labels = ("%b %d"),  
 breaks = as.Date(c("2020-03-01","2020-03-15","2020-03-30",  
 "2020-04-15","2020-04-30"))) +  
 theme(axis.text.x = element\_text(angle = 45, vjust = 1,   
 hjust = 1, size = 10, face = "bold"))

## Warning: Removed 849 row(s) containing missing values (geom\_path).



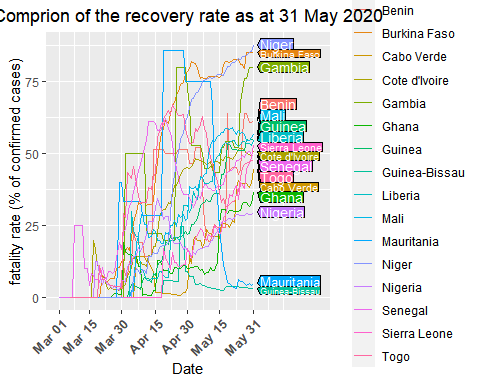
ggsave("recovered\_WA2.png",  
 width =30, height = 15,units = "cm",dpi = 70)

## Warning: Removed 849 row(s) containing missing values (geom\_path).

# Plot a line graph for recovery rate as at 31 May

recovered\_cases\_WA %>%  
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"),date<"2020-06-01") %>%  
 ggplot( aes(x=date, y=recovery\_rate, group=country, color=country)) +  
 geom\_line() +  
 geom\_dl(aes(label = country),  
 method = list(dl.trans(x = x + 0.1), "last.polygons")) +  
 scale\_fill\_manual(values = mycolors) +  
 ggtitle("Comprion of the recovery rate as at 31 May 2020") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 ylab("fatality rate (% of confirmed cases)") +  
 xlab("Date") +  
 scale\_y\_continuous(limits = c(ymin= 0,   
 ymax= max(recovered\_cases\_WA$recovery\_rate)))+  
 scale\_x\_date(limits = as.Date(c("2020-03-01","2020-06-30")),  
 date\_labels = ("%b %d"),  
 breaks = as.Date(c("2020-03-01","2020-03-15","2020-03-30",  
 "2020-04-15","2020-04-30","2020-05-15",  
 "2020-05-31"))) +  
 theme(axis.text.x = element\_text(angle = 45, vjust = 1,   
 hjust = 1, size = 10, face = "bold"))

## Warning: Removed 849 row(s) containing missing values (geom\_path).



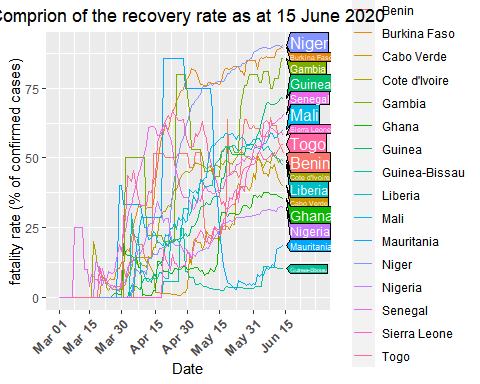
ggsave("recovered\_WA3.png",  
 width =30, height = 15,units = "cm",dpi = 70)

## Warning: Removed 849 row(s) containing missing values (geom\_path).

# Plot a line graph for recovery rate as at 15 June

recovered\_cases\_WA %>%  
 filter(country %in% c("Ghana","Nigeria","Niger","Guinea-Bissau","Guinea",  
 "Cabo Verde","Burkina Faso", "Togo", "Benin","Gambia",  
 "Mauritania","Senegal","Cote d'Ivoire",  
 "Sierra Leone","Mali","Liberia"),date<"2020-06-15") %>%  
 ggplot( aes(x=date, y=recovery\_rate, group=country, color=country)) +  
 geom\_line() +  
 geom\_dl(aes(label = country),  
 method = list(dl.trans(x = x + 0.1), "last.polygons")) +  
 scale\_fill\_manual(values = mycolors) +  
 ggtitle("Comprion of the recovery rate as at 15 June 2020") +  
 theme(plot.title = element\_text(hjust = 0.5)) +  
 ylab("fatality rate (% of confirmed cases)") +  
 xlab("Date") +  
 scale\_y\_continuous(limits = c(ymin= 0,   
 ymax= max(recovered\_cases\_WA$recovery\_rate)))+  
 scale\_x\_date(limits = as.Date(c("2020-03-01","2020-06-30")),  
 date\_labels = ("%b %d"),  
 breaks = as.Date(c("2020-03-01","2020-03-15","2020-03-30",  
 "2020-04-15","2020-04-30","2020-05-15",  
 "2020-05-31","2020-06-15"))) +  
 theme(axis.text.x = element\_text(angle = 45, vjust = 1,   
 hjust = 1, size = 10, face = "bold"))

## Warning: Removed 849 row(s) containing missing values (geom\_path).



ggsave("recovered\_WA4.png",  
 width =30, height = 15,units = "cm",dpi = 70)

## Warning: Removed 849 row(s) containing missing values (geom\_path).