

# An Ecological and Longitudinal Analysis of the African Continent (2010–2024): Modeling AIDS-Related Mortality, ART Coverage, and HIV Incidence Across 54 Countries

Uthman Al-ameen

2025-07-27

## Load Required Libraries

## Load Raw Datasets

```
#load the data set to filter only arica country
africa <- read_csv("C:/Users/THIS PC/Downloads/africa.csv")
```

```
## Rows: 376 Columns: 34
## -- Column specification -----
## Delimiter: ","
## chr (11): IndicatorCode, Indicator, ValueType, ParentLocationCode, ParentLo...
## dbl (3): Period, FactValueNumeric, FactValueTranslationID
## lgl (19): IsLatestYear, Dim1 type, Dim1, Dim1ValueCode, Dim2 type, Dim2, Di...
## dtm (1): DateModified
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
library(readr)
hiv_incidence_xlsx <- read_csv("C:/Users/THIS PC/Downloads/hiv_incidence_xlsx.csv")
```

```
## Rows: 181 Columns: 141
## -- Column specification -----
## Delimiter: ","
## chr (141): Country, 1990, 1990_lower, 1990_upper, 1990_Footnote, 1991, 1991_...
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.
```

```
library(readr)
art_coverage_xlsx <- read_csv("C:/Users/THIS PC/Downloads/art_coverage_xlsx.csv")
```

```
## Rows: 181 Columns: 61
## -- Column specification -----
## Delimiter: ","
```

```
## chr (61): Country, 2010, 2010_lower, 2010_upper, 2010_Footnote, 2011, 2011_l...
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

library(readr)
aids_mortality_xlsx <- read_csv("C:/Users/THIS PC/Downloads/aids_mortality_xlsx.csv")

## New names:
## Rows: 181 Columns: 63
## -- Column specification
## ----- Delimiter: "," chr
## (63): Country, 2010, 2010_lower, 2010_upper, 2010_Footnote, 2011, 2011_l...
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * '2024' -> '2024...58'
## * '2024_Footnote' -> '2024_Footnote...61'
## * '2024' -> '2024...62'
## * '2024_Footnote' -> '2024_Footnote...63'
```

## Filtering of only the African countries

```
# Step 1: Read the WHO Africa dataset
africa <- read_csv("C:/Users/THIS PC/Downloads/africa.csv")

## Rows: 376 Columns: 34
## -- Column specification -----
## Delimiter: ","
## chr (11): IndicatorCode, Indicator, ValueType, ParentLocationCode, ParentLo...
## dbl (3): Period, FactValueNumeric, FactValueTranslationID
## lgl (19): IsLatestYear, Dim1 type, Dim1, Dim1ValueCode, Dim2 type, Dim2, Di...
## dtm (1): DateModified
##
## i Use 'spec()' to retrieve the full column specification for this data.
## i Specify the column types or set 'show_col_types = FALSE' to quiet this message.

# Step 2: Extract the 47 unique African countries using the 'Location' column
african_countries <- africa %>%
  filter(`Location type` == "Country") %>% # Ensures we're not picking Africa or regions
  distinct(Location) %>%
  pull(Location)

# Optional: Check how many unique countries were extracted
length(african_countries) # Should return 47

## [1] 47
```

## View the 54 African countries

```

flextable::set_flextable_defaults(fonts_ignore = TRUE)

# Step 1: Convert existing vector to data frame
african_countries <- data.frame(Country = african_countries)

# Step 2: Create missing countries as data frame
missing_countries <- data.frame(
  Country = c("Egypt", "Libya", "Morocco", "Sudan", "Tunisia", "Somalia", "Djibouti")
)

# Step 3: Combine using bind_rows and override african_countries
african_countries <- bind_rows(african_countries, missing_countries)

# Step 4: Sort alphabetically and add numbering
african_countries <- african_countries %>%
  arrange(Country) %>%
  mutate(No = row_number()) %>%
  select(No, Country)

# Step 5: View the 54 African Countries (prevent wrapping)
flextable(african_countries) %>%
  autofit() %>%
  align(align = "left", j = "Country") %>%
  set_table_properties(layout = "autofit")

```

No	Country
1	Algeria
2	Angola
3	Benin
4	Botswana
5	Burkina Faso
6	Burundi
7	Cabo Verde
8	Cameroon
9	Central African Republic
10	Chad
11	Comoros
12	Congo
13	Cote d'Ivoire
14	Democratic Republic of the Congo
15	Djibouti
16	Egypt
17	Equatorial Guinea

No	Country
18	Eritrea
19	Eswatini
20	Ethiopia
21	Gabon
22	Gambia
23	Ghana
24	Guinea
25	Guinea-Bissau
26	Kenya
27	Lesotho
28	Liberia
29	Libya
30	Madagascar
31	Malawi
32	Mali
33	Mauritania
34	Mauritius
35	Morocco
36	Mozambique
37	Namibia
38	Niger
39	Nigeria
40	Rwanda
41	Sao Tome and Principe
42	Senegal
43	Seychelles
44	Sierra Leone
45	Somalia
46	South Africa
47	South Sudan
48	Sudan
49	Togo
50	Tunisia
51	Uganda
52	United Republic of Tanzania

No Country
53 Zambia
54 Zimbabwe

## Clean and Filter Country Names in Datasets

```
african_countries_clean <- african_countries$Country %>%
  stri_trans_general("Latin-ASCII") %>%
  tolower() %>%
  trimws()

filter_and_clean <- function(data, var = "Country") {
  data %>%
    mutate(Country_clean = stri_trans_general(!sym(var), "Latin-ASCII") %>% tolower() %>% trimws()) %>%
    filter(Country_clean %in% african_countries_clean)
}

filtered_art_coverage <- filter_and_clean(art_coverage_xlsx)
filtered_hiv_incidence <- filter_and_clean(hiv_incidence_xlsx)
filtered_aids_mortality <- filter_and_clean(aids_mortality_xlsx)
```

## Save Filtered Datasets

```
write_csv(filtered_art_coverage, "filtered_art_coverage.csv")
write_csv(filtered_hiv_incidence, "filtered_hiv_incidence.csv")
write_csv(filtered_aids_mortality, "filtered_aids_mortality.csv")
```

## Data clean

```
# Step 1: Drop the unwanted duplicate "2024" and "2024_Footnote" columns
filtered_aids_mortality_cleaned <- filtered_aids_mortality[, -c(62, 63)] # Drop second 2024 and 2024_F

# Step 2: Clean malformed column names (remove ...58 etc.)
filtered_aids_mortality_cleaned <- filtered_aids_mortality_cleaned %>%
  rename_with(~str_replace_all(., "\\..\\.\\.\\.\\.\\.*", ""), everything())
```

## Convert to Long Format

```
# Convert ART Coverage
art_long <- filtered_art_coverage %>%
  pivot_longer(cols = starts_with("20"), names_to = "Year", values_to = "ART_Coverage")

# Convert HIV Incidence
hiv_long <- filtered_hiv_incidence %>%
```

```

pivot_longer(cols = starts_with("20"), names_to = "Year", values_to = "HIV_Incidence")

# Convert AIDS-Related Mortality
mortality_long <- filtered_aids_mortality_cleaned %>%
  pivot_longer(cols = starts_with("20"), names_to = "Year", values_to = "AIDS_Mortality")

```

## Clean the Datasets

```

clean_data <- function(data, value_col) {
  data %>%
    filter(grepl("[0-9]{4}$", Year)) %>%
    select(Country, Year, !!sym(value_col)) %>%
    mutate(Year = as.integer(Year))}

art_long_clean <- clean_data(art_long, "ART_Coverage")
hiv_long_clean <- clean_data(hiv_long, "HIV_Incidence")
mortality_long_clean <- clean_data(mortality_long, "AIDS_Mortality")

```

## Merge and Clean

```

common_data <- art_long_clean %>%
  inner_join(hiv_long_clean, by = c("Country", "Year")) %>%
  inner_join(mortality_long_clean, by = c("Country", "Year"))

# Remove countries with missing key indicators
countries_to_remove <- c("Equatorial Guinea",
                        "Sao Tome and Principe")

common_data_filtered <- common_data %>%
  filter(!Country %in% countries_to_remove)
view(common_data)

```

## Adjust for “<” Scenario

```

common_data_adjusted <- common_data_filtered %>%
  mutate(across(c(ART_Coverage, HIV_Incidence, AIDS_Mortality), ~ {
    num <- parse_number(., locale = locale(grouping_mark = " "))
    ifelse(str_detect(., "<"), num / 2, num)
  })))

```

```

## Warning: There were 3 warnings in 'mutate()'.
## The first warning was:
## i In argument: 'across(...)'
## Caused by warning:
## ! 16 parsing failures.
## row col expected actual

```

```
## 420 -- a number ...
## 481 -- a number ...
## 482 -- a number ...
## 483 -- a number ...
## 484 -- a number ...
## ... ..
## See problems(...) for more details.
## i Run 'dplyr::last_dplyr_warnings()' to see the 2 remaining warnings.
```

## Save out the combine data

```
write.csv(common_data_adjusted, "C:/Users/THIS PC/Downloads/common_data_adjusted.csv", row.names = FALSE)
```

## R CODE FOR FULL ANALYSIS

```
summary(common_data_adjusted)
```

```
##      Country              Year      ART_Coverage      HIV_Incidence
## Length:765      Min.   :2010      Min.   : 1.00      Min.   :   100
## Class :character 1st Qu.:2013      1st Qu.:25.00      1st Qu.: 29000
## Mode  :character Median :2017      Median :43.00      Median :100000
##              Mean   :2017      Mean   :46.01      Mean   : 483010
##              3rd Qu.:2021      3rd Qu.:67.00      3rd Qu.: 420000
##              Max.   :2024      Max.   :96.00      Max.   :7800000
##              NA's   :1
## AIDS_Mortality
## Min.   :   50
## 1st Qu.: 1300
## Median : 3700
## Mean   :11826
## 3rd Qu.:16000
## Max.   :160000
## NA's   :2
```

#Summary of Extreme Values for Key Indicators Across African Countries

```
summary_table <- data.frame(
  Indicator = c("Minimum ART Coverage", "Maximum ART Coverage",
               "Minimum HIV Incidence", "Maximum HIV Incidence",
               "Minimum AIDS Mortality", "Maximum AIDS Mortality"),
  Country = c(
    common_data_adjusted$Country[which.min(common_data_adjusted$ART_Coverage)],
    common_data_adjusted$Country[which.max(common_data_adjusted$ART_Coverage)],
    common_data_adjusted$Country[which.min(common_data_adjusted$HIV_Incidence)],
    common_data_adjusted$Country[which.max(common_data_adjusted$HIV_Incidence)],
    common_data_adjusted$Country[which.min(common_data_adjusted$AIDS_Mortality)],
    common_data_adjusted$Country[which.max(common_data_adjusted$AIDS_Mortality)]
  ),
```

```

Value = c(
  min(common_data_adjusted$ART_Coverage, na.rm = TRUE),
  max(common_data_adjusted$ART_Coverage, na.rm = TRUE),
  min(common_data_adjusted$HIV_Incidence, na.rm = TRUE),
  max(common_data_adjusted$HIV_Incidence, na.rm = TRUE),
  min(common_data_adjusted$AIDS_Mortality, na.rm = TRUE),
  max(common_data_adjusted$AIDS_Mortality, na.rm = TRUE)
)
)

# Now render as flextable
flextable(summary_table) %>%
  set_caption("Summary of Extreme Values for Key Indicators Across African Countries") %>%
  align(align = "left", part = "all")

```

Table 2: Summary of Extreme Values for Key Indicators Across African Countries

Indicator	Country	Value
Minimum ART Coverage	Madagascar	1
Maximum ART Coverage	Rwanda	96
Minimum HIV Incidence	Comoros	100
Maximum HIV Incidence	South Africa	7,800,000
Minimum AIDS Mortality	Cabo Verde	50
Maximum AIDS Mortality	South Africa	160,000

## Shapiro-wilk for testing normality

```

# Shapiro-Wilk
shapiro.test(common_data_adjusted$ART_Coverage)

##
## Shapiro-Wilk normality test
##
## data: common_data_adjusted$ART_Coverage
## W = 0.96116, p-value = 2.367e-13

```



```
shapiro.test(common_data_adjusted$HIV_Incidence)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: common_data_adjusted$HIV_Incidence  
## W = 0.45484, p-value < 2.2e-16
```

```
shapiro.test(common_data_adjusted$AIDS_Mortality)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: common_data_adjusted$AIDS_Mortality  
## W = 0.63439, p-value < 2.2e-16
```

## Correlation Tests

```
# Spearman
```

```
cor.test(common_data_adjusted$ART_Coverage, common_data_adjusted$HIV_Incidence, method = "spearman")
```

```
## Warning in cor.test.default(common_data_adjusted$ART_Coverage,  
## common_data_adjusted$HIV_Incidence, : Cannot compute exact p-value with ties
```

```
##  
## Spearman's rank correlation rho  
##  
## data: common_data_adjusted$ART_Coverage and common_data_adjusted$HIV_Incidence  
## S = 51548182, p-value < 2.2e-16  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho  
## 0.306438
```

```
cor.test(common_data_adjusted$ART_Coverage, common_data_adjusted$AIDS_Mortality, method = "spearman")
```

```
## Warning in cor.test.default(common_data_adjusted$ART_Coverage,  
## common_data_adjusted$AIDS_Mortality, : Cannot compute exact p-value with ties
```

```
##  
## Spearman's rank correlation rho  
##  
## data: common_data_adjusted$ART_Coverage and common_data_adjusted$AIDS_Mortality  
## S = 67318776, p-value = 0.01617  
## alternative hypothesis: true rho is not equal to 0  
## sample estimates:  
## rho  
## 0.08709982
```

```
cor.test(common_data_adjusted$HIV_Incidence, common_data_adjusted$AIDS_Mortality, method = "spearman")
```

```
## Warning in cor.test.default(common_data_adjusted$HIV_Incidence,
## common_data_adjusted$AIDS_Mortality, : Cannot compute exact p-value with ties

##
## Spearman's rank correlation rho
##
## data: common_data_adjusted$HIV_Incidence and common_data_adjusted$AIDS_Mortality
## S = 2709412, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.9634023
```

## Regressions

```
# Basic Regression
summary(lm(AIDS_Mortality ~ ART_Coverage, data = common_data_adjusted))
```

```
##
## Call:
## lm(formula = AIDS_Mortality ~ ART_Coverage, data = common_data_adjusted)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12359 -10520  -8286   3539 148472
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11155.16    1464.00   7.620 7.58e-14 ***
## ART_Coverage    14.92      28.04   0.532  0.595
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 19140 on 760 degrees of freedom
## (3 observations deleted due to missingness)
## Multiple R-squared:  0.0003724, Adjusted R-squared: -0.0009429
## F-statistic: 0.2831 on 1 and 760 DF, p-value: 0.5948
```

```
summary(lm(AIDS_Mortality ~ HIV_Incidence, data = common_data_adjusted))
```

```
##
## Call:
## lm(formula = AIDS_Mortality ~ HIV_Incidence, data = common_data_adjusted)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -62825  -4538  -3511    98   74028
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.942e+03  4.686e+02  10.54  <2e-16 ***
## HIV_Incidence 1.422e-02  4.022e-04  35.34  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11770 on 761 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared:  0.6214, Adjusted R-squared:  0.6209
## F-statistic: 1249 on 1 and 761 DF, p-value: < 2.2e-16
```

```
summary(lm(AIDS_Mortality ~ ART_Coverage + HIV_Incidence, data = common_data_adjusted))
```

```
##
## Call:
## lm(formula = AIDS_Mortality ~ ART_Coverage + HIV_Incidence, data = common_data_adjusted)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -63039  -4948  -1844   1335   68242
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.029e+04  8.730e+02  11.782  < 2e-16 ***
## ART_Coverage -1.226e+02  1.712e+01  -7.159 1.92e-12 ***
## HIV_Incidence  1.483e-02  3.993e-04  37.145  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11410 on 759 degrees of freedom
## (3 observations deleted due to missingness)
## Multiple R-squared:  0.6453, Adjusted R-squared:  0.6443
## F-statistic: 690.3 on 2 and 759 DF, p-value: < 2.2e-16
```

#### *# Panel Regression*

```
panel_data <- pdata.frame(common_data_adjusted, index = c("Country", "Year"))
model_fe <- plm(AIDS_Mortality ~ ART_Coverage + HIV_Incidence, data = panel_data, model = "within")
summary(model_fe)
```

```
## Oneway (individual) effect Within Model
##
## Call:
## plm(formula = AIDS_Mortality ~ ART_Coverage + HIV_Incidence,
##      data = panel_data, model = "within")
##
## Unbalanced Panel: n = 51, T = 13-15, N = 762
##
## Residuals:
##      Min.    1st Qu.    Median    3rd Qu.    Max.
## -24890.02  -2100.61     61.37   1827.40  36755.60
##
```

```
## Coefficients:
##           Estimate Std. Error t-value Pr(>|t|)
## ART_Coverage -1.5843e+02  9.9772e+00 -15.879 < 2.2e-16 ***
## HIV_Incidence -3.2087e-02  1.6169e-03 -19.845 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Total Sum of Squares:    3.4651e+10
## Residual Sum of Squares: 1.5554e+10
## R-Squared:    0.55113
## Adj. R-Squared: 0.5182
## F-statistic: 435.254 on 2 and 709 DF, p-value: < 2.22e-16
```

#### # Growth Curve Modeling

```
model_gcml <- lmer(AIDS_Mortality ~ ART_Coverage + Year + (1 | Country), data = common_data_adjusted)
summary(model_gcml)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: AIDS_Mortality ~ ART_Coverage + Year + (1 | Country)
## Data: common_data_adjusted
##
## REML criterion at convergence: 15593.4
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -4.0226 -0.3831 -0.0450  0.3219 12.5007
##
## Random effects:
## Groups Name Variance Std.Dev.
## Country (Intercept) 390075053 19750
## Residual 33653605 5801
## Number of obs: 762, groups: Country, 51
##
## Fixed effects:
##           Estimate Std. Error t value
## (Intercept) -632205.1 244563.1 -2.585
## ART_Coverage -280.4 29.7 -9.443
## Year 325.7 121.9 2.672
##
## Correlation of Fixed Effects:
##           (Intr) ART_Cv
## ART_Coverag 0.915
## Year -1.000 -0.916
```

## Visualization – Trend Lines

```
# ART
art_plot <- ggplot(common_data_adjusted, aes(x = Year, y = ART_Coverage)) +
  geom_line(color = "blue") +
  facet_wrap(~ Country) +
  labs(title = "ART Coverage by Country", y = "ART Coverage (%)") +
```

```

theme_minimal() +
scale_x_continuous(breaks = seq(2010, 2023, by = 10)) +
scale_y_continuous(breaks = seq(0, 100, by = 100)) # Adjust the 20 to suit your data range

# HIV Incidence
hiv_plot <- ggplot(common_data_adjusted, aes(x = Year, y = HIV_Incidence)) +
  geom_line(color = "red") +
  facet_wrap(~ Country) +
  labs(title = "HIV Incidence by Country", y = "HIV Incidence") +
  theme_minimal() +
  scale_x_continuous(breaks = seq(2010, 2023, by = 10)) +
  scale_y_continuous(
    breaks = seq(0, 8000000, by = 8000000), # Adjust by as needed
    labels = scales::comma_format())

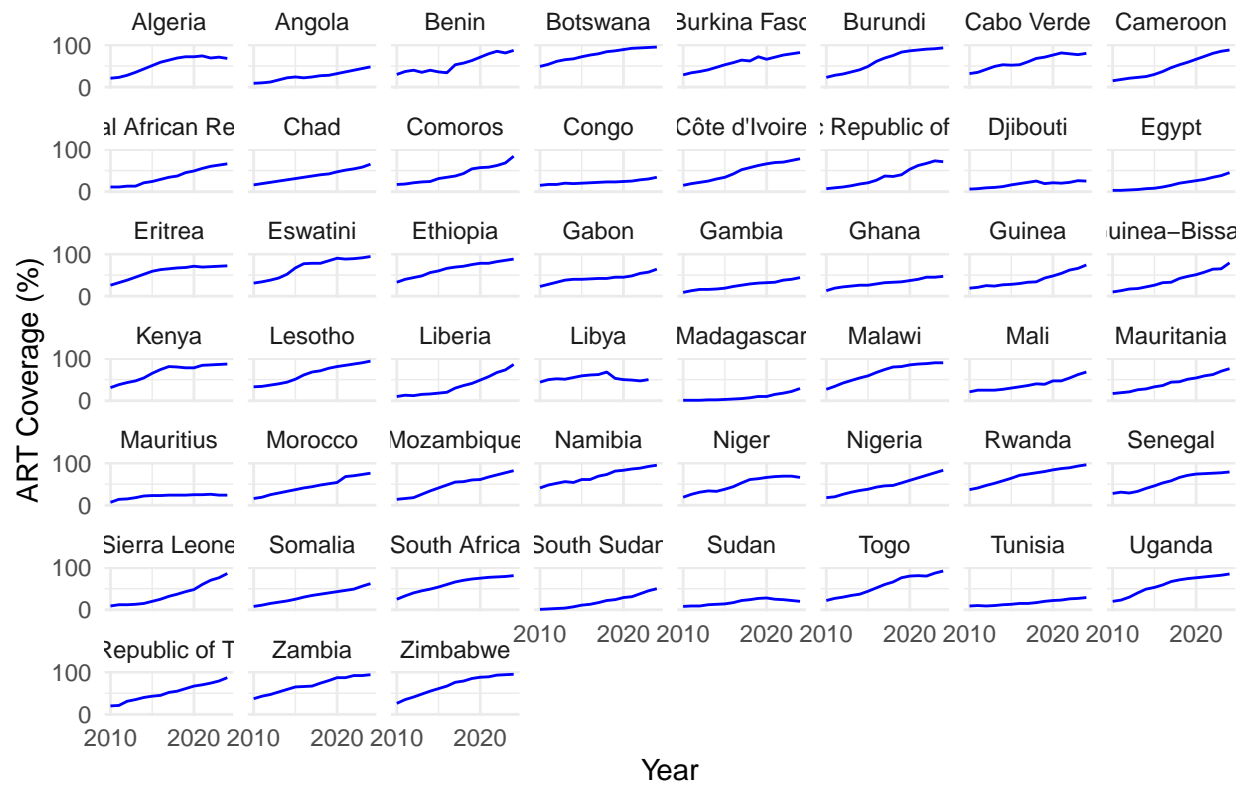
# AIDS Mortality
aids_plot <- ggplot(common_data_adjusted, aes(x = Year, y = AIDS_Mortality)) +
  geom_line(color = "darkgreen") +
  facet_wrap(~ Country) +
  labs(title = "AIDS Mortality by Country", y = "AIDS Mortality") +
  theme_minimal() +
  scale_x_continuous(breaks = seq(2010, 2023, by = 10)) +
  scale_y_continuous(breaks = seq(0, 150000, by = 150000))

```

View line graph

```
print(art_plot)
```

## ART Coverage by Country

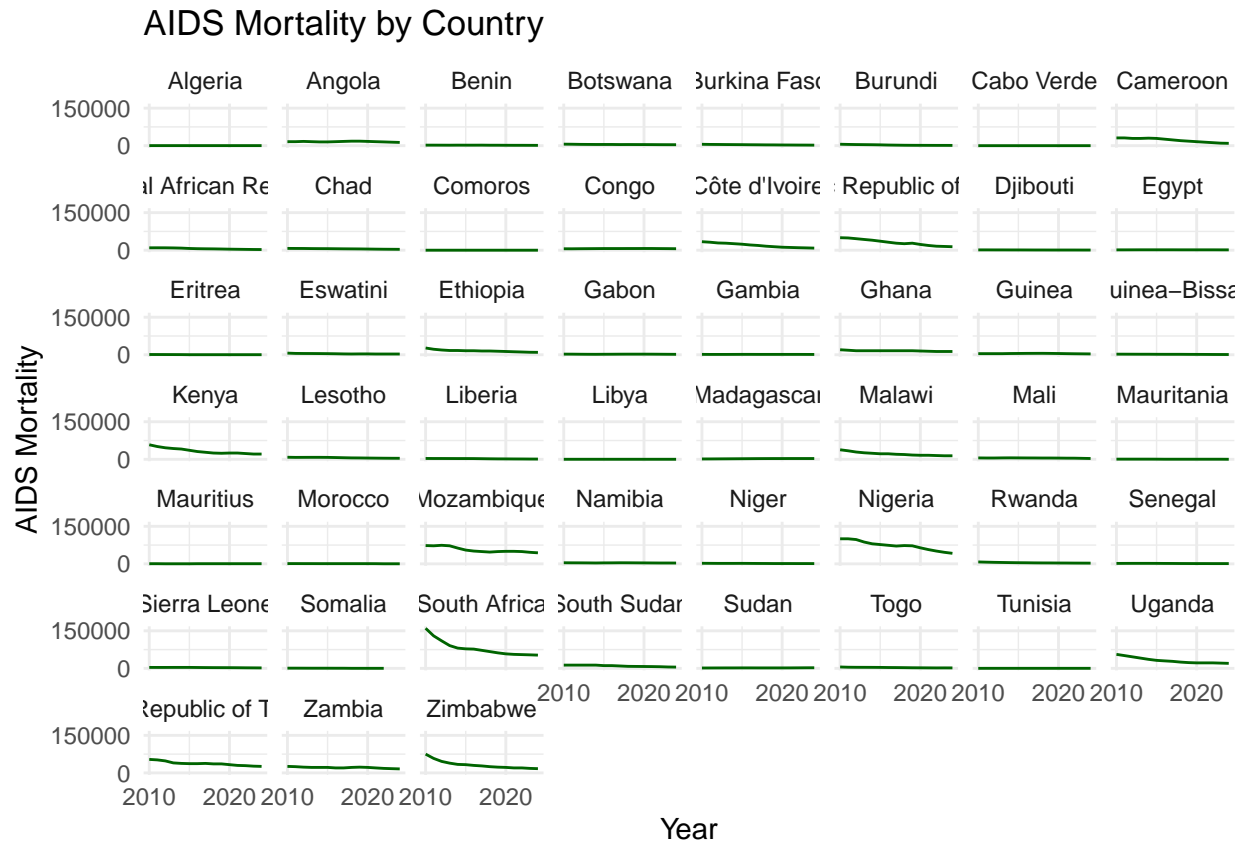


```
print(hiv_plot)
```

## HIV Incidence by Country



```
print(aids_plot)
```



## Map Creation (2024)

```
# Prepare the data
data_2024 <- common_data_adjusted %>%
  filter(Year == 2024) %>%
  mutate(Country = case_when(
    Country == "South Sudan" ~ "S. Sudan",
    Country == "United Republic of Tanzania" ~ "Tanzania",
    Country == "Democratic Republic of the Congo" ~ "Dem. Rep. Congo",
    Country == "Eswatini" ~ "eSwatini",
    Country == "Central African Republic" ~ "Central African Rep.",
    TRUE ~ Country
  ))

# Load shapefile and prepare map data
sf_use_s2(FALSE)

## Spherical geometry (s2) switched off

africa <- ne_countries(continent = "Africa", returnclass = "sf") %>%
  st_make_valid()

map_data <- left_join(africa, data_2024, by = c("name" = "Country")) %>%
  st_transform(crs = 102022)
```



```
## Warning in CPL_crs_from_input(x): GDAL Message 1: EPSG:102022 is not a valid
## CRS code, but ESRI:102022 is. Assuming ESRI:102022 was meant
```

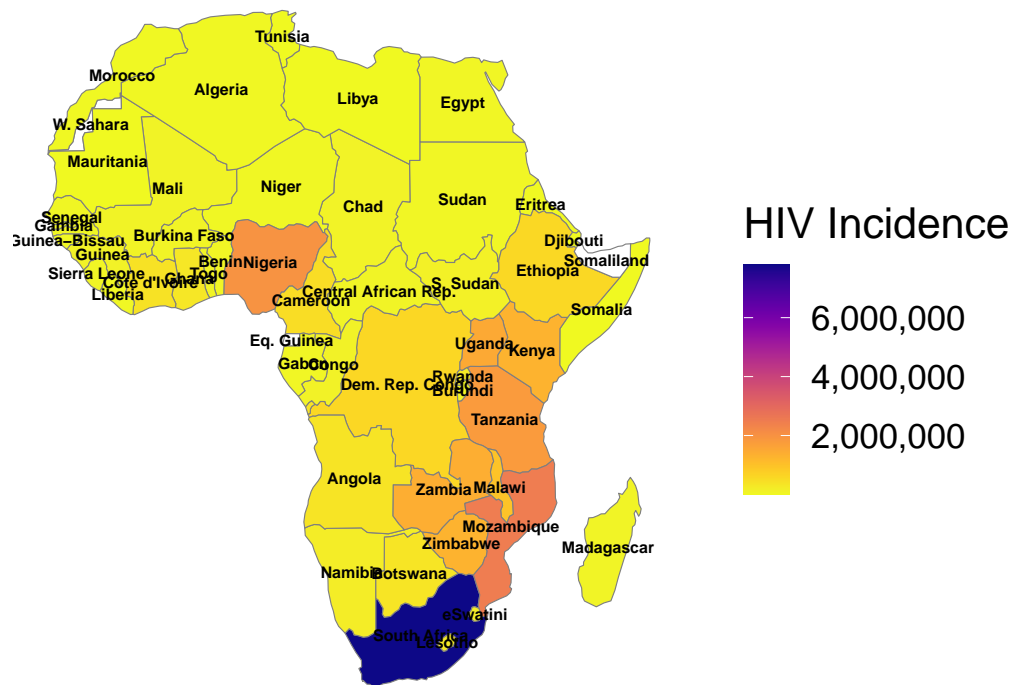
```
# Compute centroids for labeling
map_data$centroid <- st_centroid(map_data$geometry)
coords <- st_coordinates(map_data$centroid)
map_data$X <- coords[, 1]
map_data$Y <- coords[, 2]
```

## HIV Incidence Map

```
# Step 5: Plot the HIV Incidence Map (Publication-ready)
hiv_incidence_map <- ggplot(map_data) +
  geom_sf(aes(fill = HIV_Incidence), color = "grey50", size = 0.3) +
  geom_text(aes(x = X, y = Y, label = name),
            size = 2, color = "black", fontface = "bold") +
  scale_fill_viridis(option = "plasma", direction = -1, na.value = "white",
                    name = "HIV Incidence",
                    labels = comma) + # Full numbers with commas
  theme_minimal(base_size = 16) +
  ggtitle("HIV Incidence in Africa (2024)") +
  theme(
    legend.position = "right",
    plot.title = element_text(hjust = 0.5, size = 18, face = "bold"),
    panel.grid = element_blank(),
    axis.text = element_blank(),
    axis.title = element_blank(),
    axis.ticks = element_blank()
  )

# **Display the map**
print(hiv_incidence_map)
```

# HIV Incidence in Africa (2024)

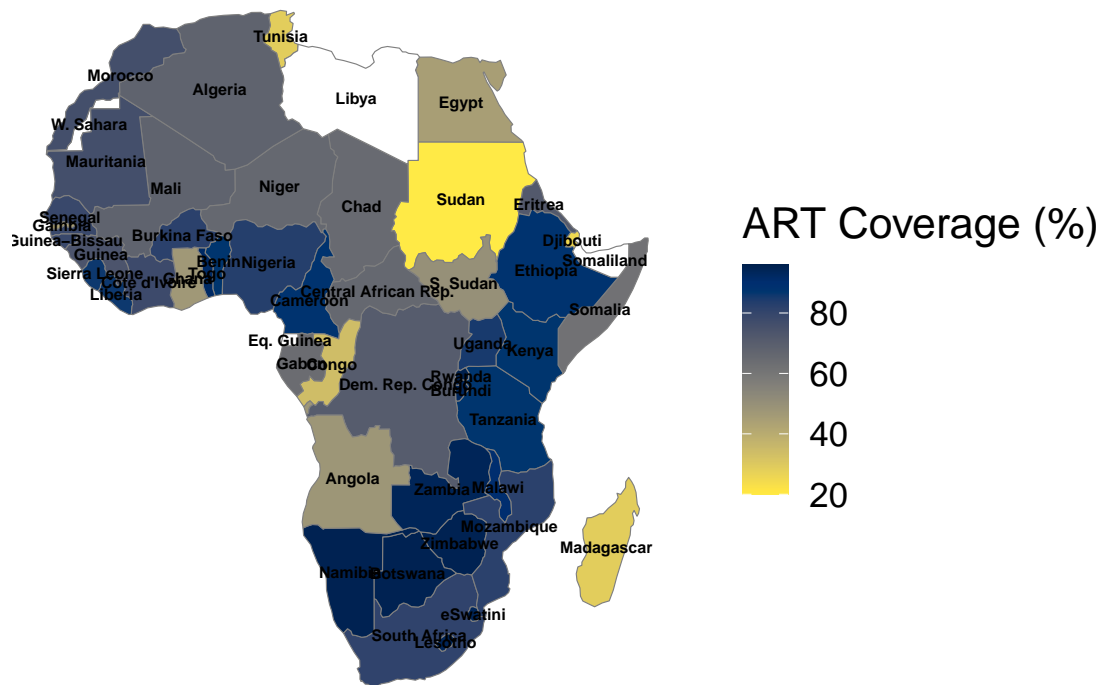


## ART Coverage Map

```
art_coverage_map <- ggplot(map_data) +
  geom_sf(aes(fill = ART_Coverage), color = "grey50", size = 0.3) +
  geom_text(aes(x = X, y = Y, label = name), size = 2, color = "black", fontface = "bold") +
  scale_fill_viridis(option = "cividis", direction = -1, na.value = "white",
    name = "ART Coverage (%)", labels = scales::comma) +
  ggtitle("ART Coverage in Africa (2024)") +
  theme_minimal(base_size = 16) +
  theme(
    legend.position = "right",
    plot.title = element_text(hjust = 0.5, size = 18, face = "bold"),
    panel.grid = element_blank(),
    axis.text = element_blank(),
    axis.title = element_blank(),
    axis.ticks = element_blank()
  )

# Display the map
print(art_coverage_map)
```

# ART Coverage in Africa (2024)

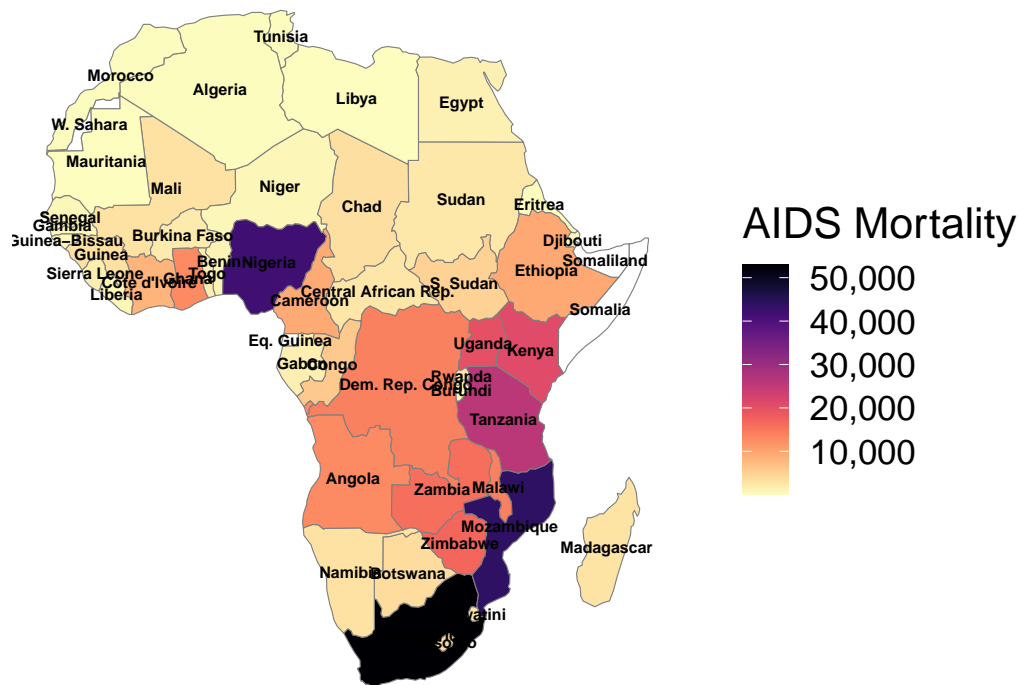


## AIDS-Related Mortality Map

```
AIDS_Mortality_Map <- ggplot(map_data) +
  geom_sf(aes(fill = AIDS_Mortality), color = "grey50", size = 0.3) +
  geom_text(aes(x = X, y = Y, label = name), size = 2, color = "black", fontface = "bold") +
  scale_fill_viridis(option = "magma", direction = -1, na.value = "white",
    name = "AIDS Mortality", labels = scales::comma) +
  ggtitle("AIDS-Related Mortality in Africa (2024)") +
  theme_minimal(base_size = 16) +
  theme(
    legend.position = "right",
    plot.title = element_text(hjust = 0.5, size = 18, face = "bold"),
    panel.grid = element_blank(),
    axis.text = element_blank(),
    axis.title = element_blank(),
    axis.ticks = element_blank()
  )

# Display the map
print(AIDS_Mortality_Map)
```

# AIDS-Related Mortality in Africa (2024)



## Saving of Maps

```
## Save AIDS Mortality Map
ggsave("AIDS_Mortality_Africa_2024.png", plot = AIDS_Mortality_Map,
       path = "C:/Users/THIS PC/Downloads", width = 10, height = 7, dpi = 300)
## Save ART Coverage Map
ggsave("ART_Coverage_Africa_2024.png", plot = art_coverage_map,
       path = "C:/Users/THIS PC/Downloads", width = 10, height = 7, dpi = 300)
## Save HIV Incidence Map
ggsave("HIV_Incidence_Africa_2024.png", plot = hiv_incidence_map, path = "C:/Users/THIS PC/Downloads",
```

## Saving of plots

```
# Define your path to Downloads
save_path <- "C:/Users/THIS PC/Downloads"

# Save ART Coverage plot
ggsave(filename = file.path(save_path, "ART_Coverage_by_Country.png"),
       plot = art_plot,
       width = 10, height = 6, dpi = 300)

# Save HIV Incidence plot
```

```
ggsave(filename = file.path(save_path, "HIV_Incidence_by_Country.png"),
        plot = hiv_plot,
        width = 10, height = 6, dpi = 300)

# Save AIDS Mortality plot
ggsave(filename = file.path(save_path, "AIDS_Mortality_by_Country.png"),
        plot = aids_plot,
        width = 10, height = 6, dpi = 300)
```