Project

Suicide Rates

We need to decide on grouping variables for carrying out Tests. Let's start by using Sex, AgeGroup, CountryName and Year as the group factors to see if different groups have the same mean vector for the continuous variables (SuicideCount, GDPPerCapita, InflationRate, EmploymentPopulationRatio).

Exploring the Data

```
library(readr)
library(dplyr)


Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

    filter, lag

The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union

library(ggplot2)

Warning: package 'ggplot2' was built under R version 4.3.3

library(gridExtra)


Attaching package: 'gridExtra'

The following object is masked from 'package:dplyr':

    combine

library(tidyverse)

Warning: package 'tidyverse' was built under R version 4.3.2

Warning: package 'forcats' was built under R version 4.3.2

── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0
──
✔ forcats   1.0.0     ✔ stringr   1.5.0
✔ lubridate 1.9.3     ✔ tibble    3.2.1
✔ purrr     1.0.2     ✔ tidyr     1.3.0
```

```
── Conflicts ──────────────────────────────────────────── tidyverse_conflicts()
──
✖ gridExtra::combine() masks dplyr::combine()
✖ dplyr::filter()      masks stats::filter()
✖ dplyr::lag()         masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors
```

```
library(car)
```

```
Warning: package 'car' was built under R version 4.3.3

Loading required package: carData

Warning: package 'carData' was built under R version 4.3.3


Attaching package: 'car'

The following object is masked from 'package:purrr':

    some

The following object is masked from 'package:dplyr':

    recode
```

```
library(MVN)
```

```
Warning: package 'MVN' was built under R version 4.3.3
```

```
library(readr)
data <- read_csv("Suicide_Rates (4).csv", show_col_types = FALSE)
head(data)
```

```
# A tibble: 6 × 8
  CountryName  Year Sex   AgeGroup    SuicideCount GDPPerCapita InflationRate
  <chr>       <dbl> <chr> <chr>              <dbl>        <dbl>         <dbl>
1 Albania      1995 Male  0-14 years             0         751.          7.79
2 Albania      1995 Male  0-14 years             6         751.          7.79
3 Albania      1995 Male  15-24 years            5         751.          7.79
4 Albania      1995 Male  15-24 years            6         751.          7.79
5 Albania      1995 Male  25-34 years            8         751.          7.79
6 Albania      1995 Male  25-34 years            5         751.          7.79
# ℹ 1 more variable: EmploymentPopulationRatio <dbl>
```

```
data$Sex <- as.factor(data$Sex)
data$AgeGroup <- as.factor(data$AgeGroup)
data$Year <- as.factor(data$Year)
data$CountryName <- as.factor(data$CountryName)
data$SuicideCount = as.numeric(data$SuicideCount)
```

```r
data$GDPPerCapita = as.numeric(data$GDPPerCapita)
data$InflationRate = as.numeric(data$InflationRate)
data$EmploymentPopulationRatio = as.numeric(data$EmploymentPopulationRatio)


categorical_vars <- names(data)[sapply(data, is.factor)]
numeric_vars <- names(data)[sapply(data, is.numeric)]

unique_values <- map(data[categorical_vars], unique)
print(unique_values)

$CountryName
 [1] Albania                          Armenia
 [3] Australia                        Austria
 [5] Azerbaijan                       Bahrain
 [7] Bahamas                          Barbados
 [9] Belarus                          Belgium
[11] Belize                           Brazil
[13] Brunei Darussalam                Bulgaria
[15] Cabo Verde                       Canada
[17] Chile                            Colombia
[19] Costa Rica                       Croatia
[21] Cyprus                           Czechia
[23] Denmark                          Dominican Republic
[25] Ecuador                          Egypt
[27] El Salvador                      Estonia
[29] Fiji                             Finland
[31] France                           Georgia
[33] Germany                          Greece
[35] Guatemala                        Guyana
[37] Hong Kong                        Hungary
[39] Iceland                          Iraq
[41] Ireland                          Israel
[43] Italy                            Jamaica
[45] Japan                            Kazakhstan
[47] Republic of Korea                Kuwait
[49] Kyrgyzstan                       Latvia
[51] Lithuania                        Luxembourg
[53] Maldives                         Malta
[55] Mauritius                        Mexico
[57] Republic of Moldova              Montenegro
[59] Netherlands                      New Zealand
[61] Nicaragua                        North Macedonia
[63] Norway                           Panama
[65] Paraguay                         Peru
[67] Philippines                      Poland
[69] Portugal                         Romania
[71] Russia                           Saint Lucia
[73] Saint Vincent and the Grenadines Serbia
[75] Singapore                        Slovakia
```

```
[77] Slovenia                             South Africa
[79] Spain                                Sri Lanka
[81] Suriname                             Sweden
[83] Switzerland                          Syrian Arab Republic
[85] Tajikistan                           Thailand
[87] Trinidad and Tobago                  Turkey
[89] Ukraine                              United Kingdom
[91] United States of America             Uruguay
[93] Uzbekistan                           Iran
[95] Lebanon                              Mongolia
[97] Malaysia                             Oman
98 Levels: Albania Armenia Australia Austria Azerbaijan Bahamas ...
Uzbekistan

$Year
 [1] 1995 1996 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010
1997
[16] 1998 2011 2012 2013 2014 1991 1992 1993 1994 2021 2020 2019 2018 2017
2016
[31] 2015 2022
32 Levels: 1991 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 ...
2022

$Sex
[1] Male   Female
Levels: Female Male

$AgeGroup
[1] 0-14 years  15-24 years 25-34 years 35-54 years 55-74 years 75+ years
6 Levels: 0-14 years 15-24 years 25-34 years 35-54 years ... 75+ years
```

Test Assumptions for conducting MANOVA

Assumptions: The variables are from a multivariate normal distribution, with consistent variance and independent samples.

1. **Normality**: The data in each group should be approximately normally distributed.

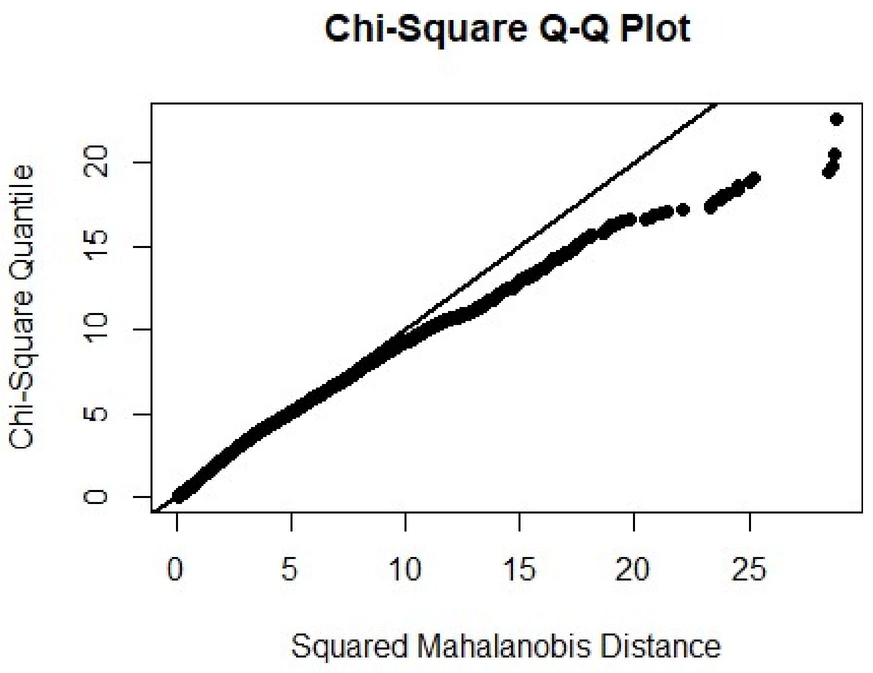
2.**Equal Covariance Matrices** (Homogeneity of Covariance): The covariance matrices of the groups should be equal.

3. **Independence**: Observations should be independent of each other.

Verifying if the data is from Multivariate normal distribution

```
# Converting grouping variables to factors
data$Sex <- as.factor(data$Sex)
data$AgeGroup <- as.factor(data$AgeGroup)
data$CountryName <- as.factor(data$CountryName)
data$Year <- as.factor(data$Year)
```

```
# Converting continuous variables to numeric, if not already
data$SuicideCount <- as.numeric(data$SuicideCount)
data$GDPPerCapita <- as.numeric(data$GDPPerCapita)
data$InflationRate <- as.numeric(data$InflationRate)
data$EmploymentPopulationRatio <- as.numeric(data$EmploymentPopulationRatio)

# Remove rows with any NA values
data <- na.omit(data)

library(MVN)
# Example of sampling the data
set.seed(123)  # for reproducibility
sampled_data <- data[sample(nrow(data), 10000), 5:8]  # adjust sample size
based on your available memory

# Run mvn on the sampled data
mvn_result <- mvn(data=sampled_data, multivariatePlot = "qq")
```




```
print(mvn_result)

$multivariateNormality
           Test        HZ p value MVN
1 Henze-Zirkler 64.97598       0  NO

$univariateNormality
              Test                  Variable Statistic   p value Normality
1 Anderson-Darling       SuicideCount          794.1707  <0.001       NO
```

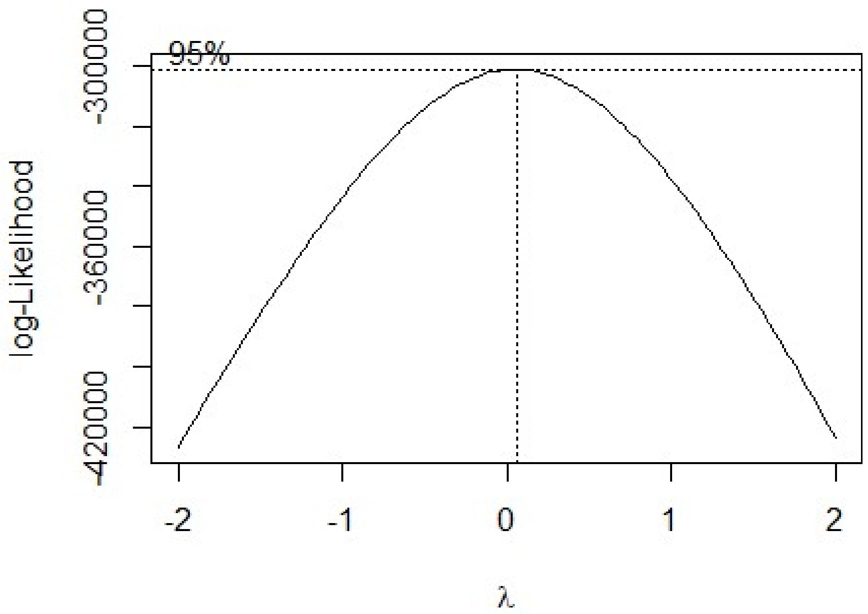
```
2 Anderson-Darling       GDPPerCapita        456.9050  <0.001      NO
3 Anderson-Darling       InflationRate       254.1447  <0.001      NO
4 Anderson-Darling EmploymentPopulationRatio  14.1925  <0.001      NO

$Descriptives
                              n         Mean      Std.Dev       Median
SuicideCount              10000    22.574200    29.042700    10.000000
GDPPerCapita              10000 17051.313723 15400.329653 11497.710765
InflationRate             10000     3.834346     3.325184     2.952301
EmploymentPopulationRatio 10000    55.917391     7.106136    56.337000
                                 Min          Max      25th         75th
SuicideCount                0.000000    131.00000    2.0000    33.000000
GDPPerCapita              186.663376  60020.36046 4609.8973 25808.860990
InflationRate              -4.478103     14.71492    1.5102     5.590259
EmploymentPopulationRatio  36.665000     74.74500   51.2140    60.433000
                                Skew    Kurtosis
SuicideCount               1.6264219  2.03117435
GDPPerCapita               1.0034203 -0.06842661
InflationRate              0.9847433  0.61753191
EmploymentPopulationRatio -0.0896565 -0.02933336
```

The results from the multivariate normality tests suggest that the dataset does not follow a multivariate normal distribution:

- **Multivariate Test (Henze-Zirkler)**: The Henze-Zirkler test yields a very high statistic (64.56467) with a p-value of 0, indicating strong evidence against multivariate normality. The summary explicitly states “NO” for multivariate normality.
- **Univariate Normality Tests**: Each variable individually also fails to conform to normality as evidenced by the Anderson-Darling tests, which all return significant results (p-values < 0.001), indicating that none of the variables are normally distributed.
- **Descriptive Statistics**: The skewness and kurtosis values for the variables further affirm the deviation from normality, as ideally, for normal distribution, skewness should be around 0 and kurtosis around 3.

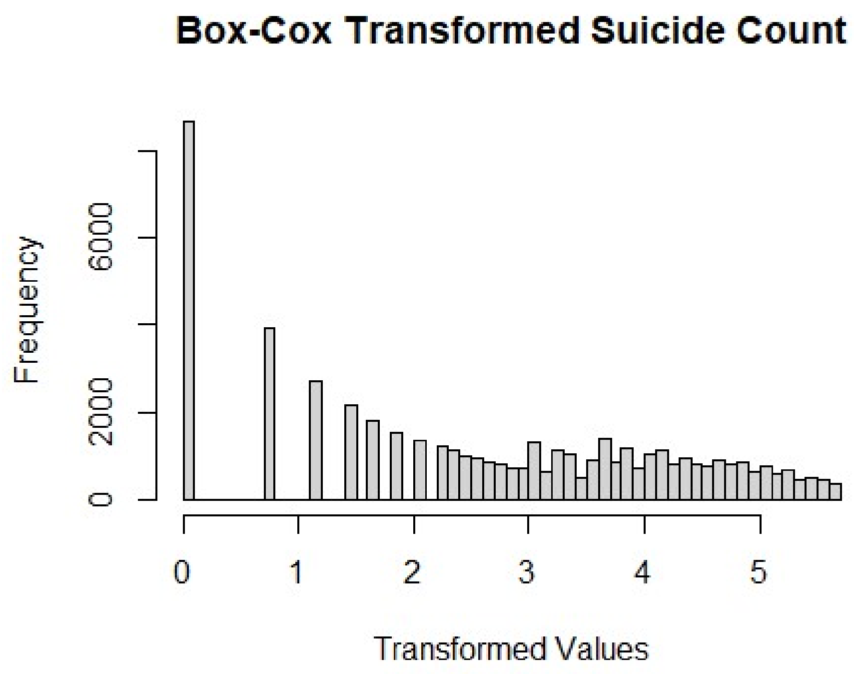
```r
# Check the minimum value in the SuicideCount column
min_suicide_count <- min(data$SuicideCount, na.rm = TRUE)

# Calculate the necessary constant to make all values positive
constant <- if(min_suicide_count <= 0) { abs(min_suicide_count) + 1 } else { 
0 }

# Apply the constant to adjust the data
data$adjusted_SuicideCount <- data$SuicideCount + constant
```

```
# Now check again to ensure all values are positive
min(data$adjusted_SuicideCount, na.rm = TRUE)

[1] 1

# Apply the Box-Cox transformation using the MASS package
library(MASS)


Attaching package: 'MASS'

The following object is masked from 'package:dplyr':

    select

# Model fitting with adjusted SuicideCount
bc <- boxcox(lm(adjusted_SuicideCount ~ 1, data = data))
```




```
# Find the lambda that maximizes the log-likelihood
optimal_lambda <- bc$x[which.max(bc$y)]
data$bc_SuicideCount <- (data$adjusted_SuicideCount^optimal_lambda - 1) /
optimal_lambda

# Check the transformed data
summary(data$bc_SuicideCount)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  0.000   1.136   2.581   2.502   3.932   5.682
```

```
hist(data$bc_SuicideCount, main="Box-Cox Transformed Suicide Count",
xlab="Transformed Values", breaks=50)
```



Even after the Transformation [Box-Cox] the Data does not appear to be Normal.

This can also shown by the Normality Tests and QQ Plots:

Anderson-Darling normality test (Size >5000)

Since Size of the greater than 5000, we need to carry out Anderson-Darling normality test as Shapiro-Wilk test does not work when size>5000 and we need sampling incase we need to use Shapiro-Wilk test.

```
library(nortest)  # This package includes alternatives like Anderson-Darling

# Applying Anderson-Darling test which is suitable for larger samples
results_ad <- lapply(data[c("SuicideCount", "GDPPerCapita", "InflationRate",
"EmploymentPopulationRatio")], ad.test)

print(results_ad)

$SuicideCount

    Anderson-Darling normality test

data:  X[[i]]
A = 4043.6, p-value < 2.2e-16
```

```
$GDPPerCapita

	Anderson-Darling normality test

data:  X[[i]]
A = 2376.6, p-value < 2.2e-16


$InflationRate

	Anderson-Darling normality test

data:  X[[i]]
A = 1322.9, p-value < 2.2e-16


$EmploymentPopulationRatio

	Anderson-Darling normality test

data:  X[[i]]
A = 73.729, p-value < 2.2e-16

# Kolmogorov-Smirnov test as another alternative (Note: This requires
empirical distribution comparison)
results_ks <- lapply(data[c("SuicideCount", "GDPPerCapita", "InflationRate",
"EmploymentPopulationRatio")], function(x) {
  ks.test(x, "pnorm", mean=mean(x, na.rm=TRUE), sd=sd(x, na.rm=TRUE))
})

Warning in ks.test.default(x, "pnorm", mean = mean(x, na.rm = TRUE), sd =
sd(x,
: ties should not be present for the Kolmogorov-Smirnov test

Warning in ks.test.default(x, "pnorm", mean = mean(x, na.rm = TRUE), sd =
sd(x,
: ties should not be present for the Kolmogorov-Smirnov test

Warning in ks.test.default(x, "pnorm", mean = mean(x, na.rm = TRUE), sd =
sd(x,
: ties should not be present for the Kolmogorov-Smirnov test

Warning in ks.test.default(x, "pnorm", mean = mean(x, na.rm = TRUE), sd =
sd(x,
: ties should not be present for the Kolmogorov-Smirnov test

print(results_ks)
```

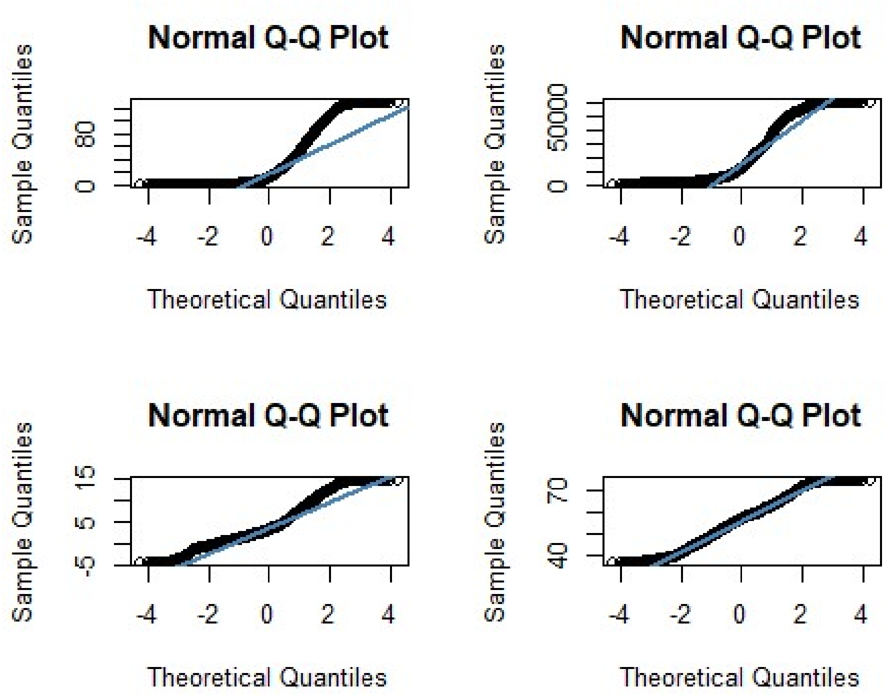
```
$SuicideCount

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  x
D = 0.21735, p-value < 2.2e-16
alternative hypothesis: two-sided


$GDPPerCapita

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  x
D = 0.1464, p-value < 2.2e-16
alternative hypothesis: two-sided


$InflationRate

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  x
D = 0.1214, p-value < 2.2e-16
alternative hypothesis: two-sided


$EmploymentPopulationRatio

    Asymptotic one-sample Kolmogorov-Smirnov test

data:  x
D = 0.031181, p-value < 2.2e-16
alternative hypothesis: two-sided
```

The results from the Anderson-Darling normality tests for the variables indicate that none of the distributions conform to normality, as evidenced by the extremely high test statistics and the very low p-values for all tested variables.

QQ PLots

```
par(mfrow = c(2, 2))

# Generating Q-Q plots for each variable
qqnorm(data$SuicideCount); qqline(data$SuicideCount, col = "steelblue", lwd =
2)
qqnorm(data$GDPPerCapita); qqline(data$GDPPerCapita, col = "steelblue", lwd =
2)
qqnorm(data$InflationRate); qqline(data$InflationRate, col = "steelblue", lwd
```

```
= 2)
qqnorm(data$EmploymentPopulationRatio);
qqline(data$EmploymentPopulationRatio, col = "steelblue", lwd = 2)
```



Homogenity Of Covariances Test:

```
# Box's M test for homogeneity of covariance matrices
library(car)

library(biotools)

Warning: package 'biotools' was built under R version 4.3.3

---
biotools version 4.2

boxM(data[, c("SuicideCount", "GDPPerCapita", "InflationRate",
"EmploymentPopulationRatio")], data$Sex)


    Box's M-test for Homogeneity of Covariance Matrices

data:  data[, c("SuicideCount", "GDPPerCapita", "InflationRate",
"EmploymentPopulationRatio")]
Chi-Sq (approx.) = 1344.6, df = 10, p-value < 2.2e-16

boxM(data[, c("SuicideCount", "GDPPerCapita", "InflationRate",
"EmploymentPopulationRatio")], data$AgeGroup)
```

```
	Box's M-test for Homogeneity of Covariance Matrices

data:  data[, c("SuicideCount", "GDPPerCapita", "InflationRate",
"EmploymentPopulationRatio")]
Chi-Sq (approx.) = 7523.4, df = 50, p-value < 2.2e-16
```

```r
boxM(data[, c("SuicideCount", "GDPPerCapita", "InflationRate",
"EmploymentPopulationRatio")], data$Year)
```

```
	Box's M-test for Homogeneity of Covariance Matrices

data:  data[, c("SuicideCount", "GDPPerCapita", "InflationRate",
"EmploymentPopulationRatio")]
Chi-Sq (approx.) = 9728, df = 310, p-value < 2.2e-16
```

The results from Box's M-test for Homogeneity of Covariance Matrices indicate significant differences in covariance matrices across different groups.

The test suggests that these differences are statistically significant when grouping by AgeGroup (Chi-Square = 7523.4, df = 50, p-value < 2.2e-16), by Year (Chi-Square = 9728, df = 310, p-value < 2.2e-16), and in a more general analysis without specific grouping (Chi-Square = 1344.6, df = 10, p-value < 2.2e-16).

This implies a lack of homogeneity in variances across the specified groups, suggesting that the data may require different analytical approaches or transformations depending on the subgroup being analyzed.

So we will use Non Parametric Method for Analysis.

Non Parametric Method

```r
# Non-parametric test for differences based on 'Sex'
kruskal.test(SuicideCount ~ Sex, data = data)
```

```
	Kruskal-Wallis rank sum test

data:  SuicideCount by Sex
Kruskal-Wallis chi-squared = 1099.8, df = 1, p-value < 2.2e-16
```

```r
kruskal.test(GDPPerCapita ~ Sex, data = data)
```

```
	Kruskal-Wallis rank sum test

data:  GDPPerCapita by Sex
Kruskal-Wallis chi-squared = 41.977, df = 1, p-value = 9.237e-11
```

```r
kruskal.test(InflationRate ~ Sex, data = data)
```

```
	Kruskal-Wallis rank sum test

data:  InflationRate by Sex
Kruskal-Wallis chi-squared = 0.16646, df = 1, p-value = 0.6833

kruskal.test(EmploymentPopulationRatio ~ Sex, data = data)


	Kruskal-Wallis rank sum test

data:  EmploymentPopulationRatio by Sex
Kruskal-Wallis chi-squared = 10.052, df = 1, p-value = 0.001522

# Non-parametric test for differences based on 'AgeGroup'
kruskal.test(SuicideCount ~ AgeGroup, data = data)


	Kruskal-Wallis rank sum test

data:  SuicideCount by AgeGroup
Kruskal-Wallis chi-squared = 8817.3, df = 5, p-value < 2.2e-16

kruskal.test(GDPPerCapita ~ AgeGroup, data = data)


	Kruskal-Wallis rank sum test

data:  GDPPerCapita by AgeGroup
Kruskal-Wallis chi-squared = 14.162, df = 5, p-value = 0.01461

kruskal.test(InflationRate ~ AgeGroup, data = data)


	Kruskal-Wallis rank sum test

data:  InflationRate by AgeGroup
Kruskal-Wallis chi-squared = 28.366, df = 5, p-value = 3.086e-05

kruskal.test(EmploymentPopulationRatio ~ AgeGroup, data = data)


	Kruskal-Wallis rank sum test

data:  EmploymentPopulationRatio by AgeGroup
Kruskal-Wallis chi-squared = 18.4, df = 5, p-value = 0.002484

# Non-parametric test for differences based on 'CountryName'
kruskal.test(SuicideCount ~ CountryName, data = data)
```

```
    Kruskal-Wallis rank sum test

data:  SuicideCount by CountryName
Kruskal-Wallis chi-squared = 22030, df = 97, p-value < 2.2e-16

kruskal.test(GDPPerCapita ~ CountryName, data = data)


    Kruskal-Wallis rank sum test

data:  GDPPerCapita by CountryName
Kruskal-Wallis chi-squared = 43928, df = 97, p-value < 2.2e-16

kruskal.test(InflationRate ~ CountryName, data = data)


    Kruskal-Wallis rank sum test

data:  InflationRate by CountryName
Kruskal-Wallis chi-squared = 18854, df = 97, p-value < 2.2e-16

kruskal.test(EmploymentPopulationRatio ~ CountryName, data = data)


    Kruskal-Wallis rank sum test

data:  EmploymentPopulationRatio by CountryName
Kruskal-Wallis chi-squared = 44280, df = 97, p-value < 2.2e-16

# Non-parametric test for differences based on 'Year'
kruskal.test(SuicideCount ~ Year, data = data)


    Kruskal-Wallis rank sum test

data:  SuicideCount by Year
Kruskal-Wallis chi-squared = 145.62, df = 31, p-value < 2.2e-16

kruskal.test(GDPPerCapita ~ Year, data = data)


    Kruskal-Wallis rank sum test

data:  GDPPerCapita by Year
Kruskal-Wallis chi-squared = 2677.2, df = 31, p-value < 2.2e-16

kruskal.test(InflationRate ~ Year, data = data)


    Kruskal-Wallis rank sum test
```

```
data:  InflationRate by Year
Kruskal-Wallis chi-squared = 8671.9, df = 31, p-value < 2.2e-16

kruskal.test(EmploymentPopulationRatio ~ Year, data = data)


    Kruskal-Wallis rank sum test

data:  EmploymentPopulationRatio by Year
Kruskal-Wallis chi-squared = 364.16, df = 31, p-value < 2.2e-16
```

The Kruskal-Wallis rank sum test was conducted to assess differences in various continuous variables across different groupings. Here's a summary of the results:

1. **SuicideCount by Sex**:
   - The Kruskal-Wallis chi-squared statistic is 1099.8 with 1 degree of freedom.
   - The p-value is < 2.2e-16, indicating a significant difference in SuicideCount across different sexes.
2. **GDPPerCapita by Sex**:
   - The Kruskal-Wallis chi-squared statistic is 41.977 with 1 degree of freedom.
   - The p-value is 9.237e-11, indicating a significant difference in GDPPerCapita across different sexes.
3. **InflationRate by Sex**:
   - The Kruskal-Wallis chi-squared statistic is 0.16646 with 1 degree of freedom.
   - The p-value is 0.6833, indicating no significant difference in InflationRate across different sexes.
4. **EmploymentPopulationRatio by Sex**:
   - The Kruskal-Wallis chi-squared statistic is 10.052 with 1 degree of freedom.
   - The p-value is 0.001522, indicating a significant difference in EmploymentPopulationRatio across different sexes.
5. **SuicideCount by AgeGroup**:
   - The Kruskal-Wallis chi-squared statistic is 8817.3 with 5 degrees of freedom.
   - The p-value is < 2.2e-16, indicating a significant difference in SuicideCount across different age groups.
6. **GDPPerCapita by AgeGroup**:

- The Kruskal-Wallis chi-squared statistic is 14.162 with 5 degrees of freedom.
- The p-value is 0.01461, indicating a significant difference in GDPPerCapita across different age groups.

7. **InflationRate by AgeGroup**:
    - The Kruskal-Wallis chi-squared statistic is 28.366 with 5 degrees of freedom.
    - The p-value is 3.086e-05, indicating a significant difference in InflationRate across different age groups.

8. **EmploymentPopulationRatio by AgeGroup**:
    - The Kruskal-Wallis chi-squared statistic is 18.4 with 5 degrees of freedom.
    - The p-value is 0.002484, indicating a significant difference in EmploymentPopulationRatio across different age groups.

9. **SuicideCount by CountryName**:
    - The Kruskal-Wallis chi-squared statistic is 22030 with 97 degrees of freedom.
    - The p-value is < 2.2e-16, indicating a significant difference in SuicideCount across different countries.

10. **GDPPerCapita by CountryName**:

- The Kruskal-Wallis chi-squared statistic is 43928 with 97 degrees of freedom.
- The p-value is < 2.2e-16, indicating a significant difference in GDPPerCapita across different countries.

11. **InflationRate by CountryName**:

- The Kruskal-Wallis chi-squared statistic is 18854 with 97 degrees of freedom.
- The p-value is < 2.2e-16, indicating a significant difference in InflationRate across different countries.

12. **EmploymentPopulationRatio by CountryName**:

- The Kruskal-Wallis chi-squared statistic is 44280 with 97 degrees of freedom.
- The p-value is < 2.2e-16, indicating a significant difference in EmploymentPopulationRatio across different countries.

13. **SuicideCount by Year:**

- The Kruskal-Wallis chi-squared statistic is 145.62with 31 degrees of freedom.
- The p-value is < 2.2e-16, indicating a significant difference in SuicideCount across different years.

14. **GDPPerCapita by Year:**

- The Kruskal-Wallis chi-squared statistic is 145.62with 31 degrees of freedom.
- The p-value is < 2.2e-16, indicating a significant difference in GDPPerCapita across different years.

15. **InflationRate by Year:**

- The Kruskal-Wallis chi-squared statistic is 8671.9 with 31 degrees of freedom.
- The p-value is < 2.2e-16, indicating a significant difference in InflationRate across different years.

16. **EmploymentPopulationRatio by Year:**

- The Kruskal-Wallis chi-squared statistic is 364.16 with 31 degrees of freedom.
- The p-value is < 2.2e-16, indicating a significant difference in EmploymentPopulationRatio across different years.

These results suggest that there are significant differences in certain variables across different groupings, while for others, the differences are not statistically significant.

Overall, the results suggest that factors like Sex, Age Group, Country Name, and Year have a significant impact on variables like Suicide Count, GDP Per Capita, and Employment Population Ratio.

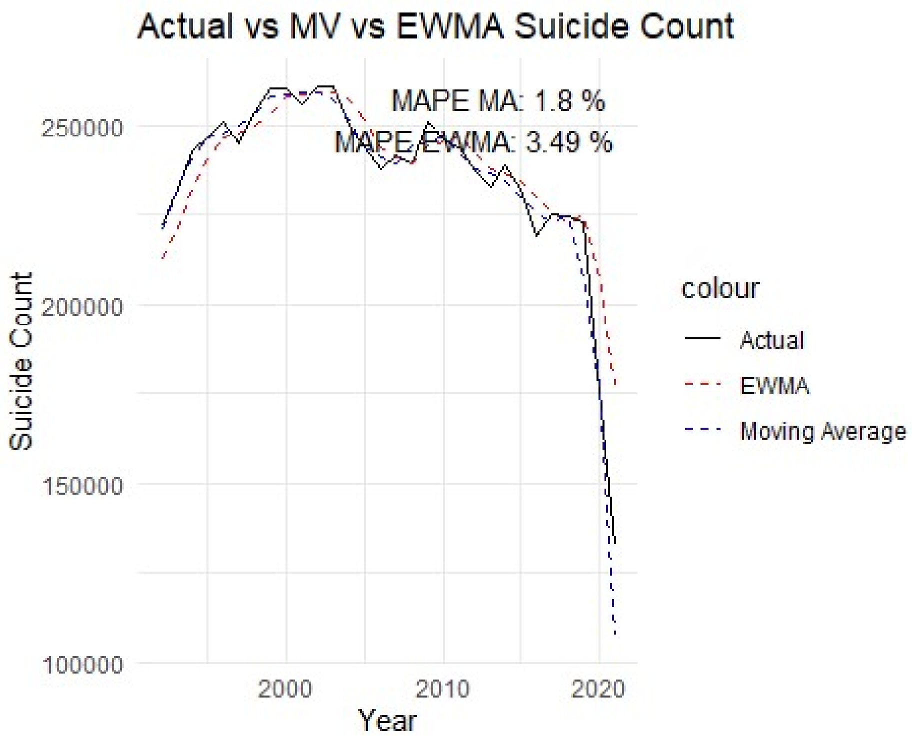
Time Series Analysis

```
library(tidyverse)
library(dplyr)
library(rstatix)
# Read the CSV file
data <- read.csv("age_std_suicide_rates_1990-2022.csv")
```

Let's begin by looking at the patterns present in the global data.

```
# Group by Year and sum the SuicideCount
result <- data %>% group_by(Year) %>% summarize(SuicideCount =
sum(SuicideCount))

# Sort the result by Year
df <- result %>% arrange(Year) %>% ungroup()

df <- as.data.frame(df)

df$MovingAverage <- zoo::rollmean(df$SuicideCount, k = 3, fill = NA)
df$ewma <- stats::filter(df$SuicideCount, filter = rep(1/3, 3), sides = 1)

df$forecast_error_MA = df$SuicideCount - df$MovingAverage
df$forecast_error_EWMA = df$SuicideCount - df$ewma
```

```
df <- df[-1, ]
df <- df[-nrow(df), ]
MAPE_MA = mean(abs(df$forecast_error_MA) / df$SuicideCount) * 100
df <- df[-1, ]
MAPE_EWMA = mean(abs(df$forecast_error_EWMA) / df$SuicideCount) * 100

ggplot(df, aes(x = Year)) +
  geom_line(aes(y = SuicideCount, color = "Actual")) +
  geom_line(aes(y = ewma, color = "EWMA"), linetype = "dashed") +
  geom_line(aes(y = MovingAverage, color = "Moving Average"), linetype =
"dashed") +
  labs(x = "Year", y = "Suicide Count", title = paste("Actual vs MV vs EWMA
Suicide Count")) +
  scale_color_manual(values = c("Actual" = "black", "EWMA" = "red", "Moving
Average" = "blue")) +
  theme_minimal()+
  annotate("text", x = max(df$Year), y = max(df$SuicideCount),
           label = paste("MAPE MA:", round(MAPE_MA, 2), "%", "\n",
                         "MAPE EWMA:", round(MAPE_EWMA, 2), "%"),
           hjust = 1, vjust = 1)
```

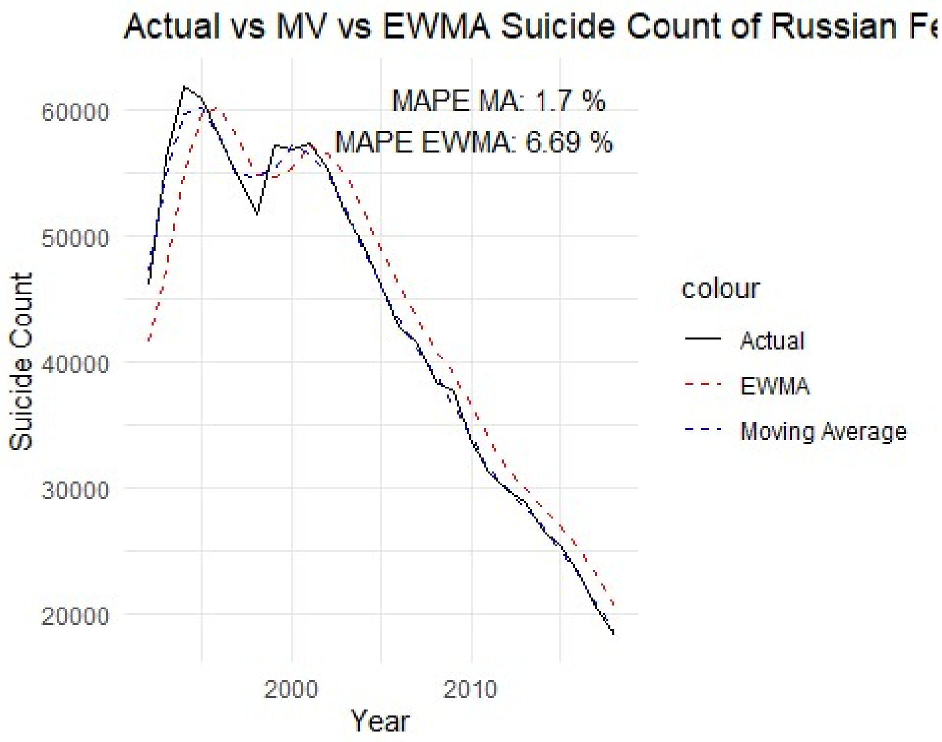


From the graph we can see there is not a noticeable trend in global suicide rates until 2020, where the number of suicides drops drastically. It is possible there would be a more noticeable seasonal pattern if our data was collected monthly rather than yearly. This may explain why our moving average and exponentially weighted moving average performed so well, as the data have already been smoothed by aggregating by year. Let’s find if there is a similar pattern among all countries by analyzing data for the five countries with the highest suicide rates.

```
library(ggplot2)

plot_ma_ewma <- function(data, num) {
  data <- data %>% select(CountryName, Sex, Year, SuicideCount)
  result <- data %>% group_by(CountryName) %>% summarize(SumSuicideCount =
sum(SuicideCount))
  result_sorted <- result %>% arrange(desc(SumSuicideCount)) %>% ungroup()
  names <- result_sorted$CountryName

  country <- names[num]
  df <- filter(data, CountryName == country) %>%
    group_by(Year) %>%
    summarize(SuicideCount = sum(SuicideCount)) %>%
    mutate(MovingAverage = zoo::rollmean(SuicideCount, k = 3, fill = NA),
           ewma = stats::filter(SuicideCount, filter = rep(1/3, 3), sides =
1),
           forecast_error_MA = SuicideCount - MovingAverage,
           forecast_error_EWMA = SuicideCount - ewma,
           )
  # Exclude first and last rows from MAPE calculations
  df <- df[-1, ]
  df <- df[-nrow(df), ]
  MAPE_MA = mean(abs(df$forecast_error_MA) / df$SuicideCount) * 100
  df <- df[-1, ]
  MAPE_EWMA = mean(abs(df$forecast_error_EWMA) / df$SuicideCount) * 100

  ggplot(df, aes(x = Year)) +
    geom_line(aes(y = SuicideCount, color = "Actual")) +
    geom_line(aes(y = ewma, color = "EWMA"), linetype = "dashed") +
    geom_line(aes(y = MovingAverage, color = "Moving Average"), linetype =
"dashed") +
    labs(x = "Year", y = "Suicide Count", title = paste("Actual vs MV vs EWMA
Suicide Count of", country)) +
    scale_color_manual(values = c("Actual" = "black", "EWMA" = "red", "Moving
Average" = "blue")) +
    theme_minimal()+
    annotate("text", x = max(df$Year), y = max(df$SuicideCount),
             label = paste("MAPE MA:", round(MAPE_MA, 2), "%", "\n",
                           "MAPE EWMA:", round(MAPE_EWMA, 2), "%"),
             hjust = 1, vjust = 1)
}
```

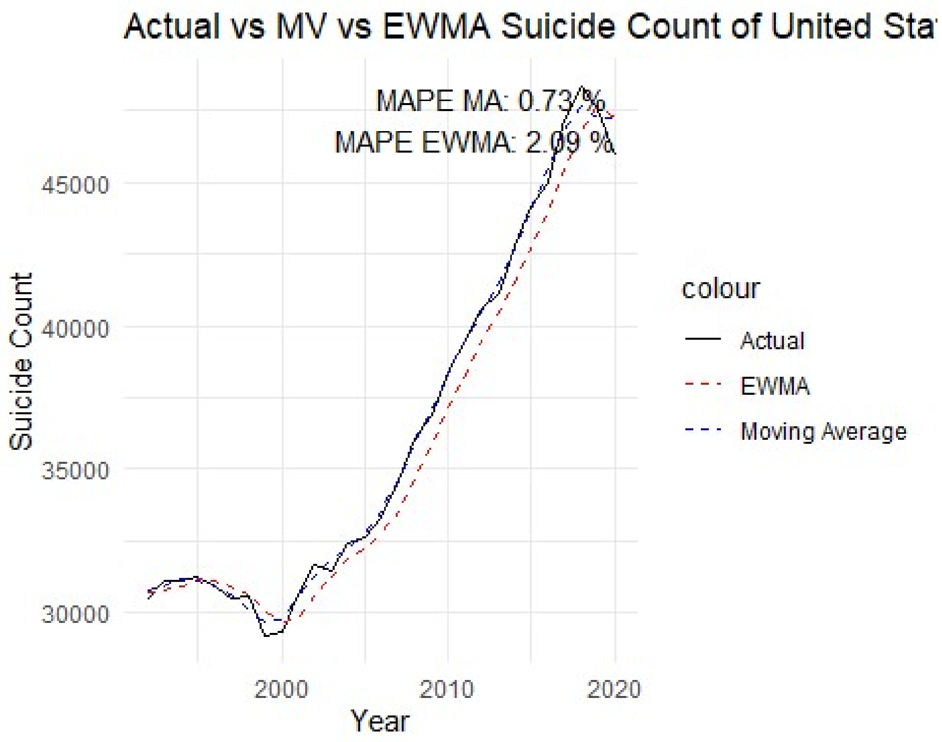
The above code uses the original data frame and creates subsets of data for each country and then groups the countries data by year, and sort the subsets by suicide count by each country in decreasing order.

```r
rank_of_country=1
plot_ma_ewma(data, rank_of_country)
```



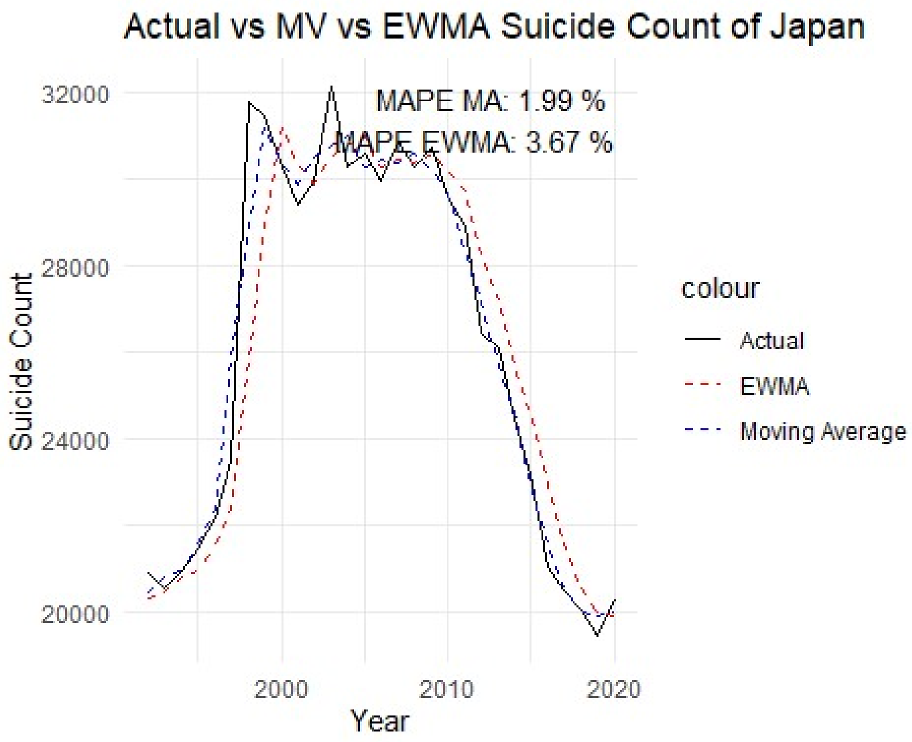
Russia has the highest suicide rate of the countries in our data set, but since 2000 suicide rates have been steadily decreasing. Moving average still tracks the data well, but EWMA has dropped noticeably.

```r
rank_of_country=2
plot_ma_ewma(data, rank_of_country)
```



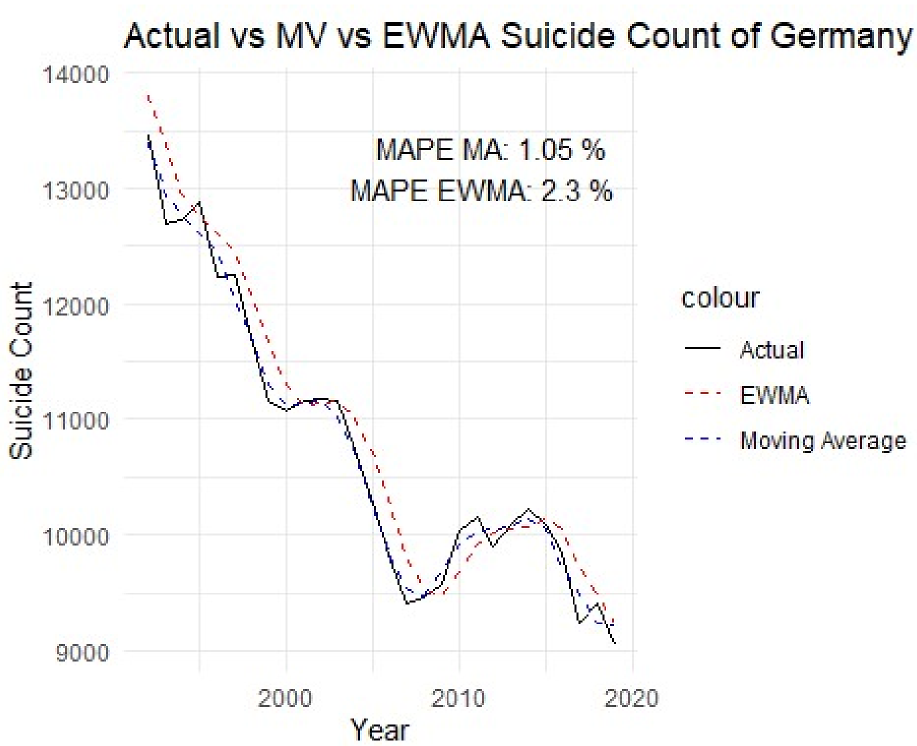
The United States has the second highest suicide rate and unfortunately has the opposite trend that Russia has. Suicide rates have been rapidly increasing since 2000. Both moving average and EWMA perform better on this data than on the global set.

```
rank_of_country=3
plot_ma_ewma(data, rank_of_country)
```



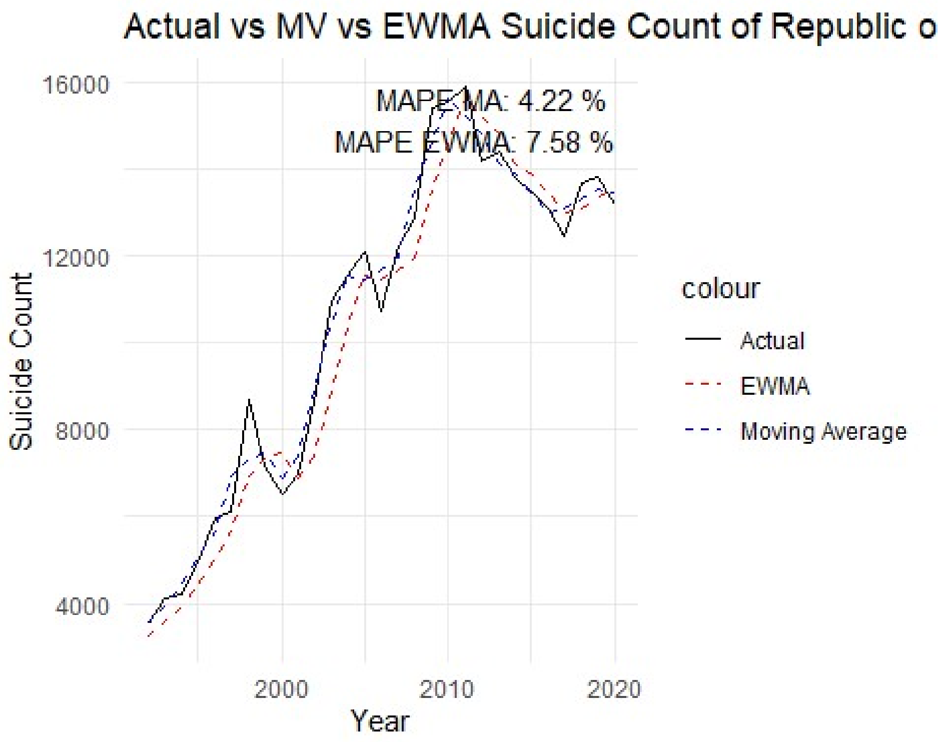
Japan has an interesting pattern, a steep increase in 1995 followed by a noisy but consistent rate until 2010 where it experienced a steep decrease in suicides. MA and EWMA both track the pattern well and perform similarly to the global data.

```
rank_of_country=4
plot_ma_ewma(data, rank_of_country)
```



Germany has a noisy but generally downward trend, with an uptick between 2010 and 2015. MA and EWMA both perform well despite the noise.

```
rank_of_country=5
plot_ma_ewma(data, rank_of_country)
```



Korea displays a very noisy pattern. There is a general upward trend that tends to spike up or down randomly. This leads to MA and EWMA performing significantly worse on this data set than on the global data.

In conclusion, using moving averages to forecast suicide rates works well globally and nationally, at least at an annual level.

Investigating the relationships between suicide rate, generation, and economic conditions.

We will begin by loading the necessary packages for manipulating the data, then separate data into groups for analysis.

```
df <- read.csv('suicide_rates_1990-2022.csv')
df_distinct <- df %>%
  distinct()
df_distinct <- drop_na(df_distinct)

df_econ <- df_distinct %>%
  group_by(CountryName, Year) %>%
  summarise(suicides = sum(SuicideCount),
            GDP = mean(GDP))

`summarise()` has grouped output by 'CountryName'. You can override using the
`.groups` argument.
```

```
df_gen <- df_distinct %>%
  group_by(Year, Generation) %>%
  summarise(suicides = sum(SuicideCount))

`summarise()` has grouped output by 'Year'. You can override using the
`.groups` argument.
```

Our first data frame will be used to calculate the economic status of each country in each year. The technical definition of a recession is two quarters of a decrease in GDP, however; our data is not that precise. An overall decrease in GDP for a year clearly indicates recession, but there will be some years which had a recession during part of the year but overall have positive GDP growth. This will cause our model to only detect the most severe economic conditions, but if there's no clear relationship between severe economic down turns and suicide rates then there is unlikely to be a relationship between moderate economic down turns and suicide rates.

```
library(ICSNP)
df_econ <- df_econ %>%
  mutate(diff_gdp <- GDP - lag(GDP))
df_econ$recession <- ifelse(df_econ$`diff_gdp <- GDP -
lag(GDP)`<=0,'recession', 'normal')
head(select(df_econ,CountryName,Year,recession))

# A tibble: 6 × 3
# Groups:   CountryName [1]
  CountryName  Year recession
  <chr>       <int> <chr>
1 Albania      1992 <NA>
2 Albania      1993 normal
3 Albania      1994 normal
4 Albania      1995 normal
5 Albania      1996 normal
6 Albania      1997 recession
```

We have created a dataframe which determines if a given country experienced a recession in a give year. We can see the first 10 years of our data for Albania. Notice the first row is labeled NA since we do not have a year in the dataset to compare 1992 to for Albania

```
df_sample <- select(df_econ, CountryName,Year,recession)
df_sample[sample(nrow(df_sample),10),]

# A tibble: 10 × 3
# Groups:   CountryName [10]
   CountryName                         Year recession
   <chr>                              <int> <chr>
 1 North Macedonia                     2000 recession
 2 Australia                           2002 normal
 3 Romania                             2011 normal
 4 Maldives                            2005 recession
 5 Singapore                           2020 recession
 6 France                              2008 normal
```

```
 7 Saint Vincent and the Grenadines  1997 normal
 8 Czechia                           2015 recession
 9 Egypt                             2019 normal
10 Costa Rica                        1996 normal
```

Here is a sample of the data set showing the same information for different countries. Next let's divide our data set in two, one for recessions and one for periods of normal economic activity. This will allow us to test if there is a difference in the mean vector of suicide rates for the two groups.

```
df_econ_recession <- subset(df_econ,recession == 'recession')
nrow(df_econ_recession)

[1] 575

df_econ_normal <- subset(df_econ,recession != 'recession')
nrow(df_econ_normal)

[1] 1673
```

We can see that we have far more samples of countries in normal economic conditions, but still enough recession responses to test our data. Let's use a t test for independent samples to see if there's a difference in mean suicide rates.

```
t.test(df_econ_normal$suicides,df_econ_recession$suicides)


	Welch Two Sample t-test

data:  df_econ_normal$suicides and df_econ_recession$suicides
t = -1.8038, df = 970.38, p-value = 0.07157
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -1317.66533    55.49603
sample estimates:
mean of x mean of y 
 2796.865  3427.950 
```

We find that there is not enough evidence to suggest a difference in suicide rates at a 5% significance level, but the p-value is still relatively low. There may be other variables influencing this score. Let's continue our investigation by looking at the anova table.

```
econ.lm <- lm(suicides~.-CountryName-`diff_gdp <- GDP - lag(GDP)`,df_econ)
summary(econ.lm)


Call:
lm(formula = suicides ~ . - CountryName - `diff_gdp <- GDP - lag(GDP)`, 
    data = df_econ)

Residuals:
```

```
   Min      1Q Median      3Q     Max 
-17113  -1609   -969   -136  57303 

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)    
(Intercept)         1.740e+05  2.652e+04   6.564 6.50e-11 ***
Year               -8.611e+01  1.321e+01  -6.517 8.81e-11 ***
GDP                 2.800e-09  5.809e-11  48.199  < 2e-16 ***
recessionrecession  6.758e+02  2.419e+02   2.793  0.00526 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4995 on 2244 degrees of freedom
  (101 observations deleted due to missingness)
Multiple R-squared:  0.5104,	Adjusted R-squared:  0.5097 
F-statistic: 779.6 on 3 and 2244 DF,  p-value: < 2.2e-16
```

According to the anova table the recession variable is now significant at a 5% level. We can see that once we accounted for overall GDP (an indication of the overall wealth of a country) that changes in condition had a clearer impact. Interestingly GDP had a positive impact on suicide rates, indicating that mo money does mean mo problems. Year had a significant negative impact on overall suicide rates, this reflects the decreasing global trend in suicide rate. Our adjusted R^2 is approximately 51%, this indicates these variables explain 51% of variation in suicide rate which is a strong effect.

Let's find out if different age groups commit suicide at different rates. We begin with a pairwise t-test with p-values adjusted by the bonferonni method. This will tell us if the mean suicide rate for each generation differ significantly when compared to each other generation. We will not be using pooled variance because we have an equal number of samples for each generation and we do not assume they have equal variance.

```
pairwise.t.test(x = df_gen$suicides,g = df_gen$Generation, p.adjust.method = 
'bonf',pool.sd = FALSE)


    Pairwise comparisons using t tests with non-pooled SD 

data:  df_gen$suicides and df_gen$Generation 

                  Baby Boomers Generation Alpha Generation X Generation Z
Generation Alpha  < 2e-16      -                -            -           
Generation X      3.2e-06      < 2e-16          -            -           
Generation Z      2.4e-14      < 2e-16          3.2e-15      -           
Millennials       3.3e-08      < 2e-16          7.7e-13      9.6e-06     
Silent Generation 5.6e-16      < 2e-16          6.7e-16      0.04        
                  Millennials
Generation Alpha  -          
Generation X      -          
Generation Z      -          
```

```
Millennials       -
Silent Generation 4.3e-10

P value adjustment method: bonferroni
```

The pairwise t test tells us that there is a significant difference in the mean vector for each pair of generations. Let's look at the ANOVA and confidence intervals to see which generations have the highest suicide rate.

```r
gen.lm <- lm(suicides~., data = df_gen)
summary(gen.lm)


Call:
lm(formula = suicides ~ ., data = df_gen)

Residuals:
   Min     1Q Median     3Q    Max 
-70461  -1438   2304   4017  18356 

Coefficients:
                             Estimate Std. Error t value Pr(>|t|)    
(Intercept)                 495480.79  163512.70   3.030  0.00279 ** 
Year                          -220.30      81.49  -2.704  0.00750 ** 
GenerationGeneration Alpha  -51708.28    2606.29 -19.840  < 2e-16 ***
GenerationGeneration X       24979.97    2606.29   9.584  < 2e-16 ***
GenerationGeneration Z      -28144.84    2606.29 -10.799  < 2e-16 ***
GenerationMillennials       -18293.41    2606.29  -7.019 4.11e-11 ***
GenerationSilent Generation -32240.06    2606.29 -12.370  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10430 on 185 degrees of freedom
Multiple R-squared:  0.8527,    Adjusted R-squared:  0.8479 
F-statistic: 178.5 on 6 and 185 DF,  p-value: < 2.2e-16

confint(gen.lm, level = 1-.05/5)

                                  0.5 %        99.5 %
(Intercept)                  69911.9518 921049.629297
Year                          -432.3805     -8.217176
GenerationGeneration Alpha  -58491.5791 -44924.983405
GenerationGeneration X       18196.6709  31763.266595
GenerationGeneration Z      -34928.1416 -21361.545905
GenerationMillennials       -25076.7041 -11510.108405
GenerationSilent Generation -39023.3603 -25456.764655
```

Our model considers Baby Boomers to be the base case, so each value indicates a suicide rate relative to Baby Boomer suicide rates. From the ANOVA table we can see the effect of belonging to each generation has a significant effect, which lets us be confident in our confidence interval estimates. From the bonferroni adjusted confidence intervals we can gauge the magnitude effect. We can see that Generation Alpha has a the lowest suicide rate compared to baby boomers, we would hope this is the case since they are the youngest generation. The only generation which has a higher suicide rate is generation X. We can see that the rates change among generation and that this explains a 85% of variation in suicide rate, but we can't explain why. Let's investigate the combined effect of generation and economic condition to see if certain generations are impacted by economic conditions more heavily than others.

```
df_com <- merge(df_distinct,df_econ)
df_ar <- drop_na(df_com) %>%
  group_by(Year, Generation,recession) %>%
  summarise(suicides = sum(SuicideCount))
```

```
`summarise()` has grouped output by 'Year', 'Generation'. You can override
using the `.groups` argument.
```

```
ar.lm <- lm(suicides~.+Generation*recession,data = df_ar)
summary(ar.lm)
```

```

Call:
lm(formula = suicides ~ . + Generation * recession, data = df_ar)

Residuals:
   Min     1Q Median     3Q    Max 
-49907  -6021   -453   6102  55349 

Coefficients:
                                              Estimate Std. Error t value
(Intercept)                                  281550.86  144040.06   1.955
Year                                           -122.07      71.76  -1.701
GenerationGeneration Alpha                   -35296.55    3144.38 -11.225
GenerationGeneration X                        18769.45    3144.38   5.969
GenerationGeneration Z                       -18234.45    3144.38  -5.799
GenerationMillennials                        -11719.39    3144.38  -3.727
GenerationSilent Generation                  -21975.16    3144.38  -6.989
recessionrecession                           -19815.94    3144.38  -6.302
GenerationGeneration Alpha:recessionrecession  19015.68    4446.82   4.276
GenerationGeneration X:recessionrecession    -12643.61    4446.82  -2.843
GenerationGeneration Z:recessionrecession      8327.10    4446.82   1.873
GenerationMillennials:recessionrecession       5023.90    4446.82   1.130
GenerationSilent Generation:recessionrecession 11815.00    4446.82   2.657
                                             Pr(>|t|)    
(Intercept)                                  0.051398 .  
Year                                         0.089807 .  
GenerationGeneration Alpha                    < 2e-16 ***
GenerationGeneration X                       5.73e-09 ***
GenerationGeneration Z                       1.46e-08 ***
GenerationMillennials                        0.000225 ***
```

```
GenerationSilent Generation                         1.36e-11 ***
recessionrecession                                  8.61e-10 ***
GenerationGeneration Alpha:recessionrecession       2.44e-05 ***
GenerationGeneration X:recessionrecession           0.004721 **
GenerationGeneration Z:recessionrecession           0.061937 .
GenerationMillennials:recessionrecession            0.259326
GenerationSilent Generation:recessionrecession 0.008237 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 12380 on 359 degrees of freedom
Multiple R-squared:  0.6071,    Adjusted R-squared:  0.5939
F-statistic: 46.22 on 12 and 359 DF,  p-value: < 2.2e-16
```

The results are surprising. Our assumption was that Millenials, Gen X, Baby Boomers, and the Silent Generation would have strong positive responses to recession, but Gen X had a negative response, Millenials had an insignificant response, and Gen Alpha had the most significant and highest positive response. The Silent Generation is the only one which matched our assumptions. Let's look at the confidence intervals to get an idea of the magnitude of the effects.

```
confint(ar.lm, level = 1-(.05/11))

                                                     0.227 %     99.773 %
(Intercept)                                    -129768.3566 692870.08063
Year                                              -326.9830     82.85274
GenerationGeneration Alpha                      -44275.6006 -26317.49616
GenerationGeneration X                            9790.3994  27748.50384
GenerationGeneration Z                          -27213.5038  -9255.39938
GenerationMillennials                           -20698.4393  -2740.33487
GenerationSilent Generation                     -30954.2135 -12996.10906
recessionrecession                              -28794.9877 -10836.88326
GenerationGeneration Alpha:recessionrecession     6317.3800  31713.97486
GenerationGeneration X:recessionrecession       -25341.9103     54.68454
GenerationGeneration Z:recessionrecession        -4371.2007  21025.39421
GenerationMillennials:recessionrecession         -7674.3942  17722.20066
GenerationSilent Generation:recessionrecession    -883.2974  24513.29744
```

Implementing the bonferonni p value adjustment we can see our interaction effect may not be significant for most generations, but it is still strongly significant for generation alpha. This suggests that generation alpha is more strongly affected by changes in economic conditions than one may assume.

In conclusion we saw a significant effect on the mean value of suicide rates from both economic conditions and generation. There was also a significant difference between the effect of economic conditions on the mean suicide rates for generation alpha.

Does Simpson's Paradox occur? If so, for which variables and how?

```
{library(readr)} df = df <- read.csv('suicide_rates_1990-2022.csv')
```

Simpson's Paradox Function for the following confounding variables:

1. Sex

```
df$Sex = as.factor(df$Sex)
df$AgeGroup = as.factor(df$AgeGroup)
df$CountryName = as.factor(df$CountryName)


simpsons_paradox = function(var1, var2, df) {
  df_new = df
  df_new[is.na(df_new) | df_new == "Inf"] = NA

  if (class(var1) == "numeric"){
    summary1 = summary(glm(var1 ~ var2, data=df_new))
    summary2 = summary(glm(var1 ~ var2 + Sex, data=df_new))

    if (summary1$coefficients[2, "Estimate"] > 0 & summary2$coefficients[2,
"Estimate"] < 0) {
      return(TRUE)  # Simpson's Paradox exists
    } else if (summary1$coefficients[2, "Estimate"] < 0 &
summary2$coefficients[2, "Estimate"] > 0) {
      return(TRUE)  # Simpson's Paradox exists
    } else {
      return(FALSE) # No Simpson's Paradox
    }
  }
  else {return(FALSE)}
}

variables = c("Year", "SuicideCount", "GDPPerCapita", "InflationRate",
"EmploymentPopulationRatio", "CountryName", "Sex", "AgeGroup")

# Iterating through each pair of variables
cat("SEX:\n")

SEX:

for (i in 1:(length(variables)-1)) {
  for (j in (i+1):length(variables)) {
    paradox = simpsons_paradox(df[[variables[i]]], df[[variables[j]]], df)

    if (paradox) {
      cat("Simpson's Paradox exists between", variables[i], "and",
variables[j], "\n")
    } else {
      cat("No Simpson's Paradox between", variables[i], "and", variables[j],
"\n")
    }
  }
}
```

```
No Simpson's Paradox between Year and SuicideCount
No Simpson's Paradox between Year and GDPPerCapita
No Simpson's Paradox between Year and InflationRate
No Simpson's Paradox between Year and EmploymentPopulationRatio
No Simpson's Paradox between Year and CountryName
No Simpson's Paradox between Year and Sex
No Simpson's Paradox between Year and AgeGroup
No Simpson's Paradox between SuicideCount and GDPPerCapita
No Simpson's Paradox between SuicideCount and InflationRate
No Simpson's Paradox between SuicideCount and EmploymentPopulationRatio
No Simpson's Paradox between SuicideCount and CountryName
No Simpson's Paradox between SuicideCount and Sex
No Simpson's Paradox between SuicideCount and AgeGroup
No Simpson's Paradox between GDPPerCapita and InflationRate
No Simpson's Paradox between GDPPerCapita and EmploymentPopulationRatio
No Simpson's Paradox between GDPPerCapita and CountryName
No Simpson's Paradox between GDPPerCapita and Sex
No Simpson's Paradox between GDPPerCapita and AgeGroup
No Simpson's Paradox between InflationRate and EmploymentPopulationRatio
No Simpson's Paradox between InflationRate and CountryName
No Simpson's Paradox between InflationRate and Sex
No Simpson's Paradox between InflationRate and AgeGroup
No Simpson's Paradox between EmploymentPopulationRatio and CountryName
No Simpson's Paradox between EmploymentPopulationRatio and Sex
No Simpson's Paradox between EmploymentPopulationRatio and AgeGroup
No Simpson's Paradox between CountryName and Sex
No Simpson's Paradox between CountryName and AgeGroup
No Simpson's Paradox between Sex and AgeGroup
```

2. Age Group

```
simpsons_paradox = function(var1, var2, df) {
  df_new = df
  df_new[is.na(df_new) | df_new == "Inf"] = NA

  if (class(var1) == "numeric"){
    summary1 = summary(glm(var1 ~ var2, data=df_new))
    summary2 = summary(glm(var1 ~ var2 + AgeGroup, data=df_new))

    if (summary1$coefficients[2, "Estimate"] > 0 & summary2$coefficients[2,
"Estimate"] < 0) {
      return(TRUE)  # Simpson's Paradox exists
    } else if (summary1$coefficients[2, "Estimate"] < 0 &
summary2$coefficients[2, "Estimate"] > 0) {
      return(TRUE)  # Simpson's Paradox exists
    } else {
      return(FALSE) # No Simpson's Paradox
    }
  }
  else {return(FALSE)}
}
```

```
# Iterating through each pair of variables
cat("AgeGroup:\n")
```

```
AgeGroup:
```

```
for (i in 1:(length(variables)-1)) {
  for (j in (i+1):length(variables)) {
    paradox = simpsons_paradox(df[[variables[i]]], df[[variables[j]]], df)

    if (paradox) {
      cat("Simpson's Paradox exists between", variables[i], "and",
variables[j], "\n")
    } else {
      cat("No Simpson's Paradox between", variables[i], "and", variables[j],
"\n")
    }
  }
}
```

```
No Simpson's Paradox between Year and SuicideCount
No Simpson's Paradox between Year and GDPPerCapita
No Simpson's Paradox between Year and InflationRate
No Simpson's Paradox between Year and EmploymentPopulationRatio
No Simpson's Paradox between Year and CountryName
No Simpson's Paradox between Year and Sex
No Simpson's Paradox between Year and AgeGroup
No Simpson's Paradox between SuicideCount and GDPPerCapita
No Simpson's Paradox between SuicideCount and InflationRate
No Simpson's Paradox between SuicideCount and EmploymentPopulationRatio
No Simpson's Paradox between SuicideCount and CountryName
No Simpson's Paradox between SuicideCount and Sex
No Simpson's Paradox between SuicideCount and AgeGroup
No Simpson's Paradox between GDPPerCapita and InflationRate
No Simpson's Paradox between GDPPerCapita and EmploymentPopulationRatio
No Simpson's Paradox between GDPPerCapita and CountryName
No Simpson's Paradox between GDPPerCapita and Sex
No Simpson's Paradox between GDPPerCapita and AgeGroup
No Simpson's Paradox between InflationRate and EmploymentPopulationRatio
No Simpson's Paradox between InflationRate and CountryName
No Simpson's Paradox between InflationRate and Sex
No Simpson's Paradox between InflationRate and AgeGroup
No Simpson's Paradox between EmploymentPopulationRatio and CountryName
No Simpson's Paradox between EmploymentPopulationRatio and Sex
No Simpson's Paradox between EmploymentPopulationRatio and AgeGroup
No Simpson's Paradox between CountryName and Sex
No Simpson's Paradox between CountryName and AgeGroup
No Simpson's Paradox between Sex and AgeGroup
```

3. Country Name

```r
simpsons_paradox = function(var1, var2, df) {
  df_new = df
  df_new[is.na(df_new) | df_new == "Inf"] = NA

  if (class(var1) == "numeric"){
    summary1 = summary(glm(var1 ~ var2, data=df_new))
    summary2 = summary(glm(var1 ~ var2 + CountryName, data=df_new))

    if (summary1$coefficients[2, "Estimate"] > 0 & summary2$coefficients[2,
"Estimate"] < 0) {
      return(TRUE)  # Simpson's Paradox exists
    } else if (summary1$coefficients[2, "Estimate"] < 0 &
summary2$coefficients[2, "Estimate"] > 0) {
      return(TRUE)  # Simpson's Paradox exists
    } else {
      return(FALSE) # No Simpson's Paradox
    }
  }
  else {return(FALSE)}
}

# Iterating through each pair of variables
cat("CountryName:\n")
```

```
CountryName:
```

```r
for (i in 1:(length(variables)-1)) {
  for (j in (i+1):length(variables)) {
    paradox = simpsons_paradox(df[[variables[i]]], df[[variables[j]]], df)

    if (paradox) {
      cat("Simpson's Paradox exists between", variables[i], "and",
variables[j], "\n")
    } else {
      cat("No Simpson's Paradox between", variables[i], "and", variables[j],
"\n")
    }
  }
}
```

```
No Simpson's Paradox between Year and SuicideCount
No Simpson's Paradox between Year and GDPPerCapita
No Simpson's Paradox between Year and InflationRate
No Simpson's Paradox between Year and EmploymentPopulationRatio
No Simpson's Paradox between Year and CountryName
No Simpson's Paradox between Year and Sex
No Simpson's Paradox between Year and AgeGroup
No Simpson's Paradox between SuicideCount and GDPPerCapita
No Simpson's Paradox between SuicideCount and InflationRate
No Simpson's Paradox between SuicideCount and EmploymentPopulationRatio
```

```
No Simpson's Paradox between SuicideCount and CountryName
No Simpson's Paradox between SuicideCount and Sex
No Simpson's Paradox between SuicideCount and AgeGroup
No Simpson's Paradox between GDPPerCapita and InflationRate
No Simpson's Paradox between GDPPerCapita and EmploymentPopulationRatio
No Simpson's Paradox between GDPPerCapita and CountryName
No Simpson's Paradox between GDPPerCapita and Sex
Simpson's Paradox exists between GDPPerCapita and AgeGroup
No Simpson's Paradox between InflationRate and EmploymentPopulationRatio
No Simpson's Paradox between InflationRate and CountryName
Simpson's Paradox exists between InflationRate and Sex
No Simpson's Paradox between InflationRate and AgeGroup
No Simpson's Paradox between EmploymentPopulationRatio and CountryName
Simpson's Paradox exists between EmploymentPopulationRatio and Sex
Simpson's Paradox exists between EmploymentPopulationRatio and AgeGroup
No Simpson's Paradox between CountryName and Sex
No Simpson's Paradox between CountryName and AgeGroup
No Simpson's Paradox between Sex and AgeGroup
```

We see Simpson's paradox in the following variables when CountryName is the confounding variable causing the paradox :

Year and AgeGroup

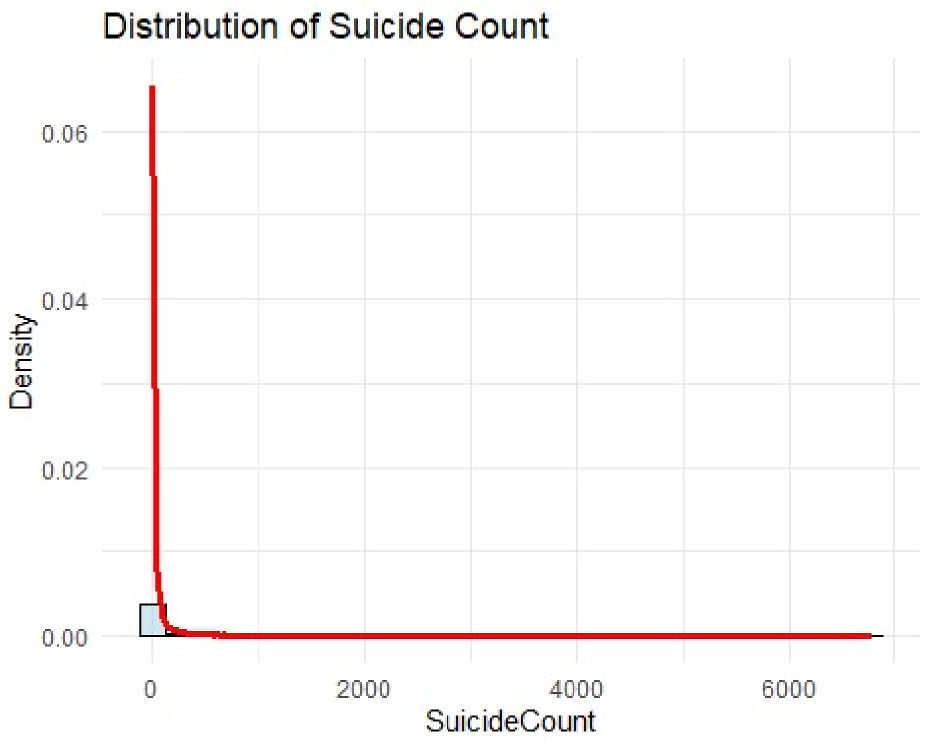
SuicideCount and GDPPerCapita

EmploymentPopulationRatio and AgeGroup

If not normally distributed, where are the distributions centered?

```
library(ggplot2)

# SuicideCount :

ggplot(df, aes(x = SuicideCount)) +
  geom_histogram(aes(y = ..density..), bins = 30, fill = "lightblue", color =
"black", alpha = 0.6) +
  geom_density(color = "red", size = 1.2) +
  labs(title = "Distribution of Suicide Count",
       x = "SuicideCount",
       y = "Density") +
  theme_minimal()

Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
ℹ Please use `linewidth` instead.

Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2
3.4.0.
ℹ Please use `after_stat(density)` instead.

Warning: Removed 464 rows containing non-finite outside the scale range
(`stat_bin()`).
```

```
Warning: Removed 464 rows containing non-finite outside the scale range
(`stat_density()`).
```



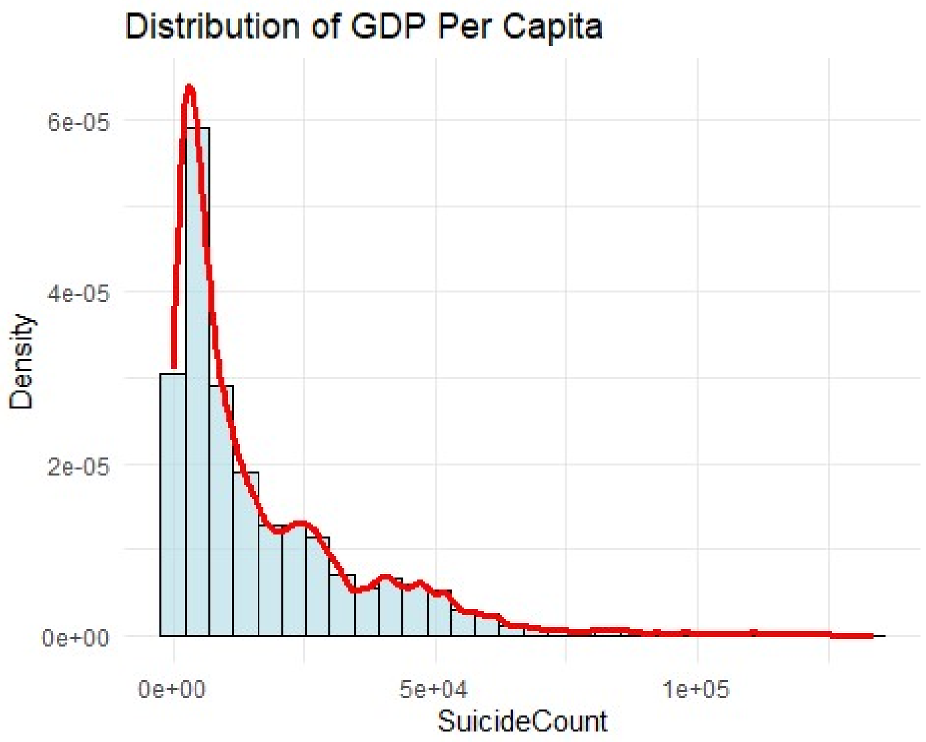

```
mean(df$SuicideCount)

[1] NA

median(df$SuicideCount)

[1] NA
```

We see the Suicide Count variable does not follow Normal Distribution but follows an Exponential distribution with mean = 22.63627 and median = 10

GDPPerCapita :

```
ggplot(df, aes(x = GDPPerCapita)) +
  geom_histogram(aes(y = ..density..), bins = 30, fill = "lightblue", color =
"black", alpha = 0.6) +
  geom_density(color = "red", size = 1.2) +
  labs(title = "Distribution of GDP Per Capita",
       x = "SuicideCount",
       y = "Density") +
  theme_minimal()

Warning: Removed 7240 rows containing non-finite outside the scale range
(`stat_bin()`).
```

```
Warning: Removed 7240 rows containing non-finite outside the scale range
(`stat_density()`).
```



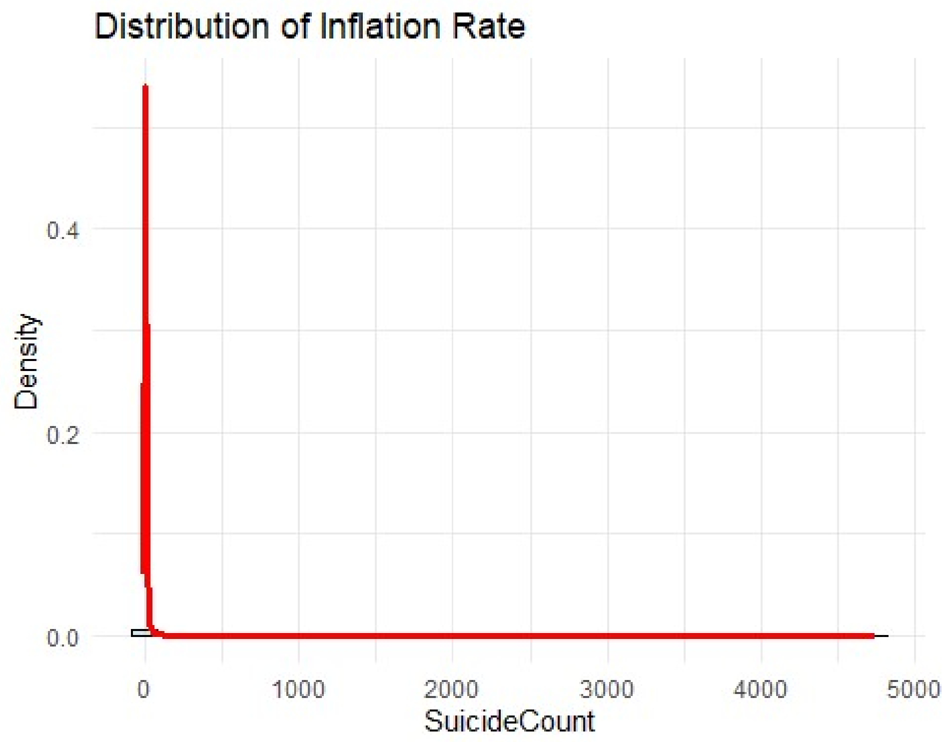

```
mean(df$GDPPerCapita)

[1] NA

median(df$GDPPerCapita)

[1] NA
```

We see the Suicide Count variable does not follow Normal Distribution but follows a non-symmetric right-skewed distribution mean = 17045.04 and median = 11452.78

InflationRate :

```
ggplot(df, aes(x = InflationRate)) +
  geom_histogram(aes(y = ..density..), bins = 30, fill = "lightblue", color =
"black", alpha = 0.6) +
  geom_density(color = "red", size = 1.2) +
  labs(title = "Distribution of Inflation Rate",
       x = "SuicideCount",
       y = "Density") +
  theme_minimal()

Warning: Removed 14460 rows containing non-finite outside the scale range
(`stat_bin()`).
```

```
Warning: Removed 14460 rows containing non-finite outside the scale range
(`stat_density()`).
```



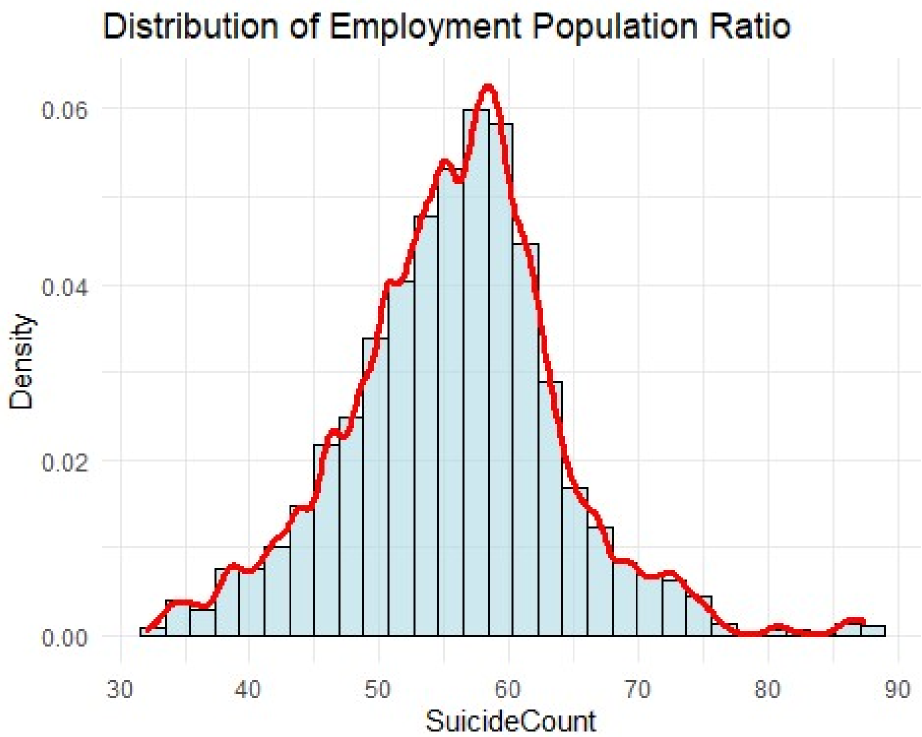

```
mean(df$InflationRate)

[1] NA

median(df$InflationRate)

[1] NA
```

Again, we see the Inflation Rate variable does not follow Normal Distribution but follows a non-symmetric right-skewed distribution mean = 3.825735 and median = 2.932363

EmploymentPopulationRatio :

```
ggplot(df, aes(x = EmploymentPopulationRatio)) +
  geom_histogram(aes(y = ..density..), bins = 30, fill = "lightblue", color =
"black", alpha = 0.6) +
  geom_density(color = "red", size = 1.2) +
  labs(title = "Distribution of Employment Population Ratio",
       x = "SuicideCount",
       y = "Density") +
  theme_minimal()

Warning: Removed 11120 rows containing non-finite outside the scale range
(`stat_bin()`).
```

```
Warning: Removed 11120 rows containing non-finite outside the scale range
(`stat_density()`).
```




```
mean(df$EmploymentPopulationRatio)

[1] NA

median(df$EmploymentPopulationRatio)

[1] NA
```

We see the Suicide Count variable appears to follow a Normal Distribution with mean = 55.80597 and median = 56.261