Impact of Environmental Challenges on Pollution

1. INTRODUCTION

In the modern era, environmental pollution causes a serious impact on living orgranisms, including human beings. It also becomes one of the most serious global challenges. The pollution is caused by either purpose or accidental pollution such as wastewater from cities. Pollution affects air and water; air pollution is present on substances in the atmosphere that affects human health or other living orgranisms. It also has a detrimental effect of climate to Earth, thus the effect to Earth will be drought, flood etc. Alternatively, air quality is measures with Air Quality Index or Particulate Matter[1]. Next, water pollution will affect negative effects to aquatic ecosystems, humans and other living orgranisms that rely on water. Water quality can be measured using the most common called Coloured Dissolved Organic matter, Conductivity etc[2].

To let users, know the facts and statistics on water and air pollutions in every country, visualisation should be made and act as an important role using the data collected from the measurements mentioned. This role is responsible and gives a warning to users that how serious impact that water and air pollution has affected in the whole world.

The problem statement is to show the impact of environmental challenges on pollution in the world where shows the air and water quality by scaling from 0 to 100.

2. RELATED WORK

Based on the author[3] who provides a past solution which used R to show data visualisation using water and air pollution dataset[4].
At first, 3 libraries used for the visualisation, which is readr, dplyr and ggplot2. After that, csv file is read through and input as a dataframe. The checking of dataframe proceeds and the author founds that the values of Region and Country column in the dataframe have a space in front of the values. The author thinks that this might cause problems, so he erased every value's first space in those columns in Figure 2.1.

In [4]:
```
data$Region <- substring(data$Region, 2)
data$Country <- substring(data$Country, 2)
```

Figure 2.1: Code for removing spaces from every value on those columns

The next step author will take is to check if there is any NA value in the dataframe using the code from Figure 2.2 aand result from Figure 2.3.

In [6]:
```
for (column in colnames(data)){
  print(paste('In the', column, 'column of our dataset, there are', sum(is.na(data
$column)), 'NAs'))
if (sum(is.na(data$column)) != 0) {
print('There are some NAs here so please, take a look!')
} else {
}
}
```

Figure 2.2: Code to check NA values in columns

```
[1] "In the City column of our dataset, there are 0 NAs"
[1] "In the Region column of our dataset, there are 0 NAs"
[1] "In the Country column of our dataset, there are 0 NAs"
[1] "In the AirQuality column of our dataset, there are 0 NAs"
[1] "In the WaterPollution column of our dataset, there are 0 NAs"
```

Figure 2.3: result for NA values

The next step of data cleaning includes name and value changing from this author. To make it clearer, the WaterPollution will be changed to AirQuality to fit and match easily with WaterQuality. The average of WaterQualirt and Air Quality will be calculated. But before that, the value for WaterQuality will be changed to 100 – WaterPollution, the values will also be rounded to integer for a simple and clearer display for visulisation as shown in Figure 2.4

In [7]:

```
avg_data <- data %>%
select(Country, AirQuality, WaterPollution) %>%
mutate(WaterQuality = 100 - WaterPollution) %>%
select(Country, AirQuality, WaterQuality) %>%
group_by(Country) %>%
summarise(WaterQuality = mean(WaterQuality), AirQuality = mean(AirQuality)) %>%
mutate(WaterQuality = round(WaterQuality), AirQuality = round(AirQuality))

head(avg_data, n = 5)
```

Figure 2.4: changing WaterQuality name and value and finding the mean

And finally, the author made the visualisation below in Figure 2.5.

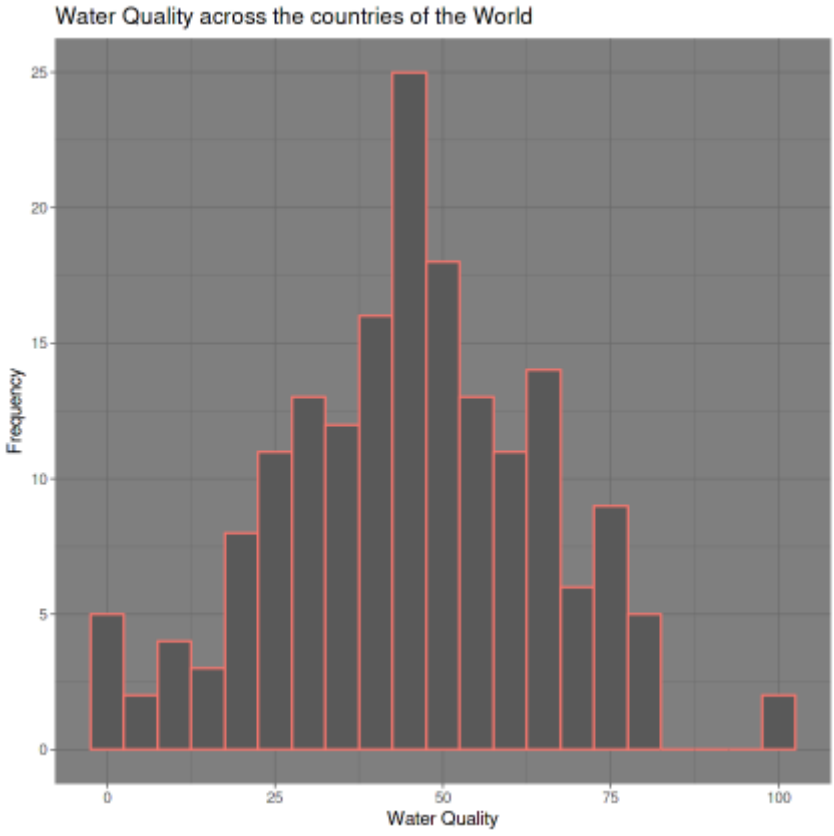

Figure 2.4: Histogram of Water Quality across the countries of the World

3. METHODS/DESIGN

For this data visualisation, the first thing that need to do is data cleaning. R simple library and dplyr library can do the cleaning and some useful statistic datas. After loading data, the dataframe is created. Select() function will be used to select columns from the dataframe, by adding minus sign into select() parameters will remove the column and show the rest of the columns. View() function gets the object or class and preview the details of it. Some other left_join() methods to join map table and dataframe to achieve joined table with matched values.

ggplot() function is to make a chart by adding several plotting conditions such as geom_polygon(), theme(), scale_fill_steps()_. These are to solve the UI and map data mapping problems. By showing world map, geom_polygon() is used. For bar charts, geom_col() is used, for solving the horizontal issue, coord_flip() is also used to solve it.

By designing a map, the map will be a world map that will be plotted within each region area and scaled from two different colors with a gradient style. Title and tooltip will be used up using ggtitle() and tooltips=''. The titles for the map will be indicating both air and water polluted across the world, that are "Air Pollution in Every Country" and "Water Pollution in Every Country".Removing the axis gridlines and axis titles are used using axis.text.x = element_blank(),axis.text.y = element_blank(),axis.ticks = element_blank(), axis.title.y = element_blank(), axis.title.x = element_blank(), rect = element_blank(). For the bar charts, the bars will be shown in a horizontal way and will be indicated by using a scale of 0 to 100 with a gradient style of two colors again.The title will be also added for the 4 bar charts named as "Top 10 Countries with the Cleanest Air", "Top 10 Countries with the Cleanest Water", "Top 10 Countries with the Most Polluted Air", "Top 10 Countries with the Most Polluted Air".

4. IMPLEMENTATION

4.1. R

Data cleaning will be the first step before doing visualisation. R library for data cleaning can be used in simple cleaning such that the visualisation can be presented by a more simple and clear way.

```
data = read.csv("C:/cities_air_quality_water_pollution.18-10-2021.csv")
```

This reads the csv file that input the file directory in the `read.csv()` function.

```
data1 = data %>% select(-Region)
```

This function is to select columns from the dataframe but with the minus sign in the parameter, it becomes to select all except "Region" column will be shown.

```
colnames(data1)[2] <- 'region'
```

This function is to rename columns from data1 dataframe where specifying the index for the column number and rename it to "region"

```
view(data1)
```

This function is to view variable declared and determine any errors or maintenance usage.

```
wpdata = aggregate(x=data1$WaterPollution, by=list(region=data1$region), FUN=mean)
```

This aggregrate function is to compute the summary statistics for subsets of the data. After using this function. The output will be class data.frame. For the above line shown, this function is to get the mean from "WaterPollution" column group by "region".

4.2. dplyr

The next library to clean datasets is dplyr package. It is a grammar of data manipulation and provde consistent set of verbs that solve common data manipulation challenges. As in the code show below:

```
wpdata = wpdata %>% drop_na(x); #line 38
```

In this case, `drop_na()` function in dplyr library drops the rows from any columns in the dataset that contains a missing value. Another explanation is this function only keeps the "complete" row where no missing values.

```
wpdata = wpdata %>%
  mutate_all(trimws) %>%
  mutate(x=round(wpdata$x, digits=0))
```

The function used in these lines is `mutate_all()` & `mutate()`. `mutate_all()` affects all columns in the dataframe and `trimws` in this function is used to remove leading and/or whitespace from any string values. While `mutate()` is just to affect a single row chosen from dataframe specified and `round()` used to make the numbers / decimals to any digit number of decimals we want. In here, we are using 0 digits of demicals to ensure more clear and easy way to implement visulisations.

```
wpdata1 <- left_join(mapdata, wpdata)
```

To combine all the vlaues into world map dataframe from ggplot2 mapdata, `left_join()` function is used to combine includes all rows in dataframe and map to the mapdata.

```
top10_ac = aqdata[order(aqdata$x),] %>% slice(1:10)
```

Function of `[order(),]` is used to change the order according to the parameter in the function. `slice(1:10)` gets the first 10 rows from the dataframe selected in the parameter mentioned at the head of right-hand assignment.

4.3. ggplot2

After cleaning data, the visulisation can be easily implemented by using library provided by ggplot2.

```
mapdata <- map_data('world')
```

`map_data('world')` is used to turn data from maps package provided into a dataframe that can be plotted with ggplot2.

```
wpmap <- ggplot(wpdata1, aes( x = long, y = lat, group=group)) +
  geom_polygon(aes(data = region, fill = x, text = paste(region, ': ', x),), color = "black")
+
  scale_fill_steps(name='Water Polluted Range', low='white',high='red') +
  theme(
    axis.text.x = element_blank(),
    axis.text.y = element_blank(),
    axis.ticks = element_blank(),
    axis.title.y = element_blank(),
    axis.title.x = element_blank(),
    rect = element_blank()
  )
```

In the above codes provided, there are some functions that are from ggplot2 library. `ggplot()` intializes the ggplot object and delcare input dataframe into a spcified set of plot aesthethics. Next, `geom_polygon(aes())` can point all the values from dataframe to the map generated and turns scatterplot into a map, which will draw each country into a distinct polygon. `scale_fill_steps()` transform scale beside map generated into column of boxeswith colour gradient. Title for the scale can also be manually input to get a clearer guidelin e. `theme()` can modify the map image either to show gridlines or x or y-axis, title of x or y-axis etc.

```
ggplotly(apbar + coord_flip() + ggtitle("Top 10 Countries with the Most Polluted Air"), tooltip='text')
```

`coord_flip()` flips the barchart so that the bars data will be shown in horizontal form. `ggtitle()` function adds a custom title string on top of the header for the map. Tooltip in the code above shown is to cuustomize hover data value shown in the visualisation。

4.4. ggplotly

Lastly, the library used is ggplotly. `ggplotly()` converts ggplot2's `ggplot()` to a plotly object.

5. RESULTS

5.1. Air Pollution in World Map

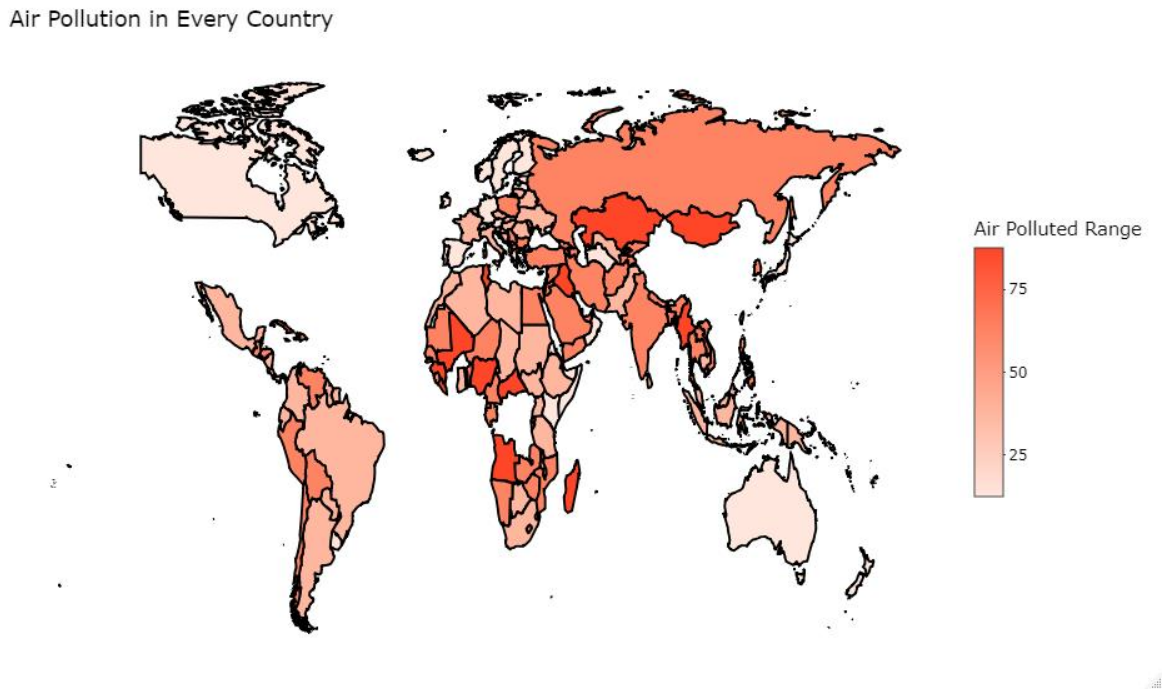

Figure 5.1: Data Visualisation of Air Pollution in Every Country (map_air_pollution.png)

In Figure 5.1, this visualisation shows the air polluted varies from 0 (good) to 100 (extremely bad) which the scale indicates from white to red which increasingly becoming red as the value is increasing. The interactions can be made when you hover on every part in the map, it will show region name and the air pollution value.

5.2. Water Pollution in World Map

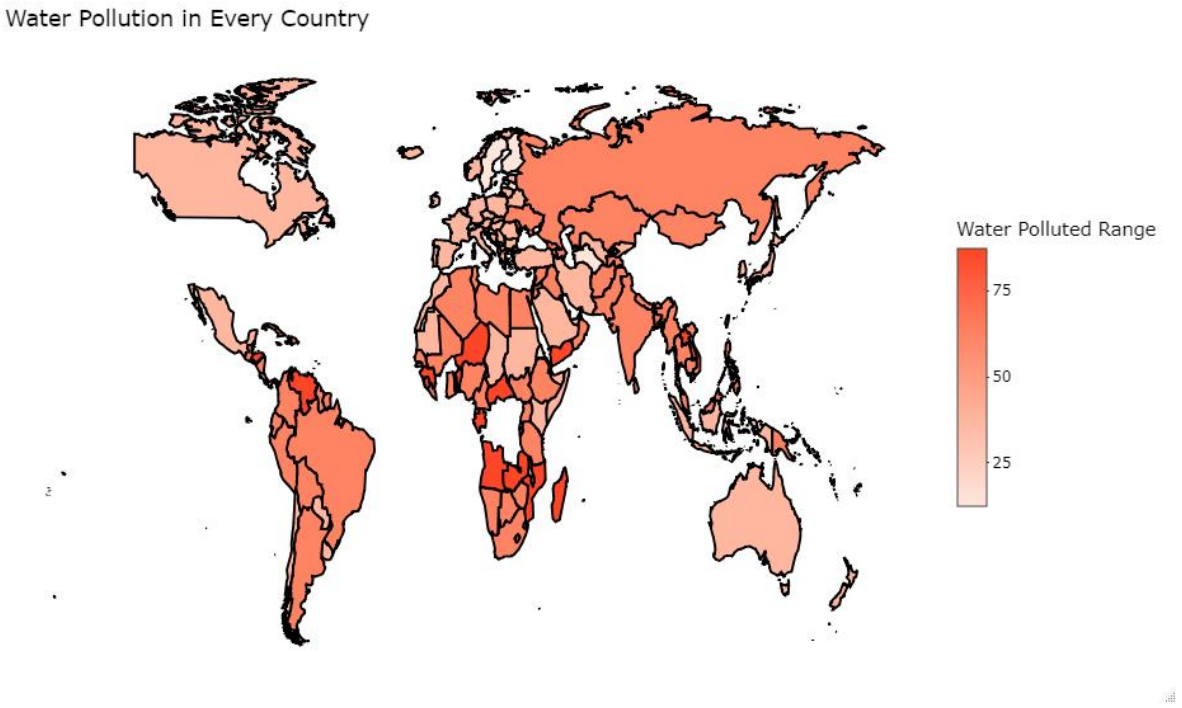

Figure 5.2: Data Visualisation of Water Pollution in Every Country (map_water_pollution.png)

In Figure 5.2, this visualisation shows the water polluted varies from 0 (good) to 100 (extremely bad) which the scale indicates from white to red which increasingly becoming red as the value is increasing. The interactions can be made when you hover on every part in the map, it will show region name and the water pollution value.

5.3. Top 10 Countries Hold the Cleanest Air

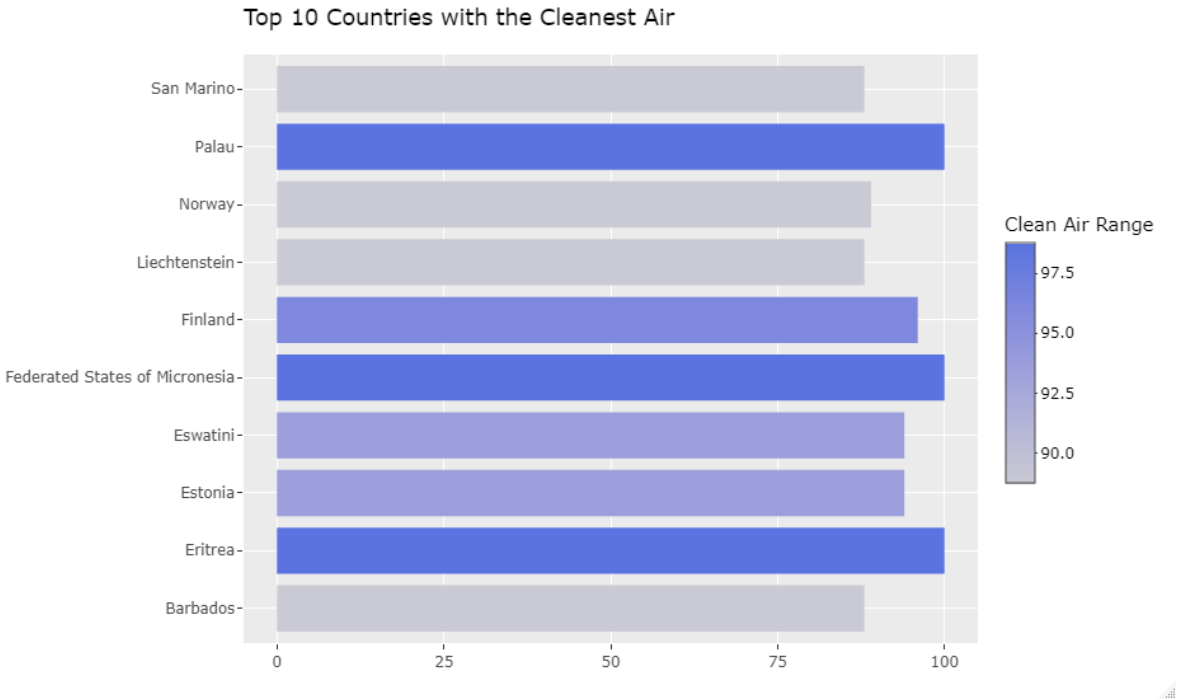

Figure 5.3: Data Visualisation of Top 10 Countries Hold the Cleanest Air (top10_cleanest_air.png)

In Figure 5.3, this visualisation shows top 10 countries with the cleanest air, the range is indicated from grey to blue. As the cleanest air varies from 0 (extremely bad) to 100 (good) which related to the scales shown. Value will be accurately shown from the dataframe to this visualisation by hovering it. We can know that Palau, Federated States of Micronesia has 100 points of clean value which means they have the highest clean air points (best air quality).

5.4. Top 10 Countries Hold the Cleanest Water

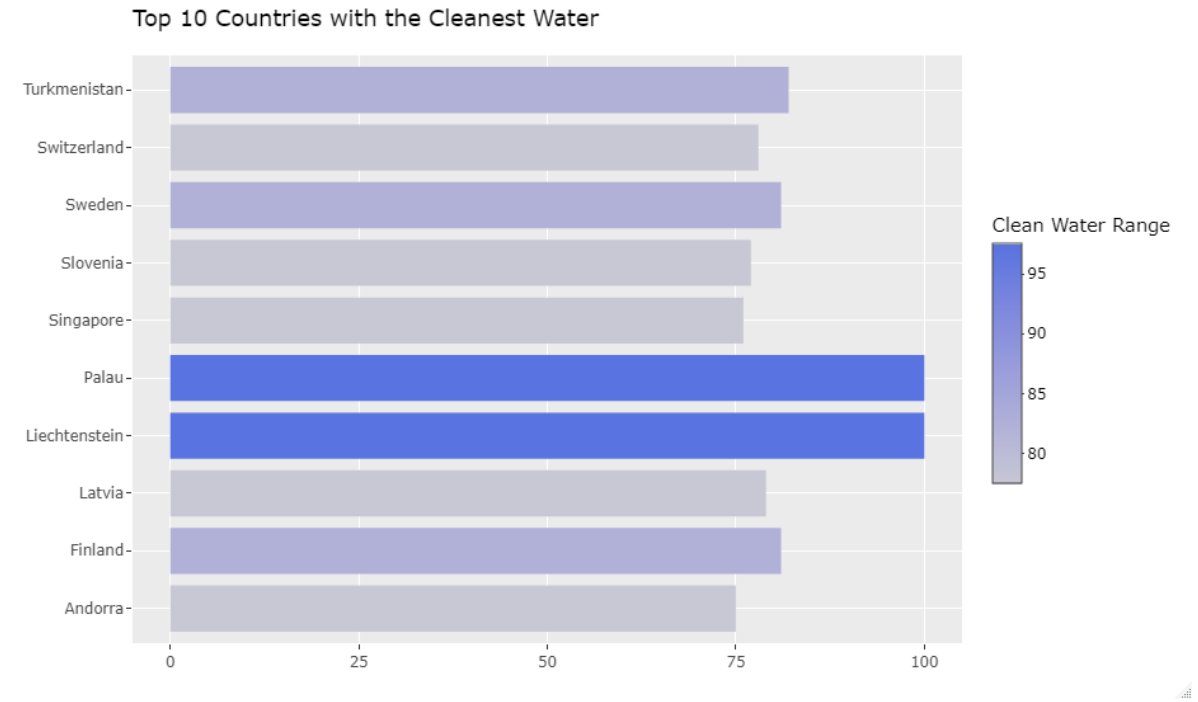

Figure 5.4: Data Visualisation of Top 10 Countries Hold the Cleanest Water (top10_cleanest_water.png)

In Figure 5.4, this visualisation shows top 10 countries with the cleanest water, the range is indicated from grey to blue. As the cleanest water varies from 0 (extremely bad) to 100 (good) which related to the scales shown. Value will be accurately shown from the dataframe to this visualisation by hovering it. We can know that Palau, Liechtenstein has 100 points of clean value which means they have the highest clean water points (best water quality).

5.5. Top 10 Countries Hold the Most Polluted Air

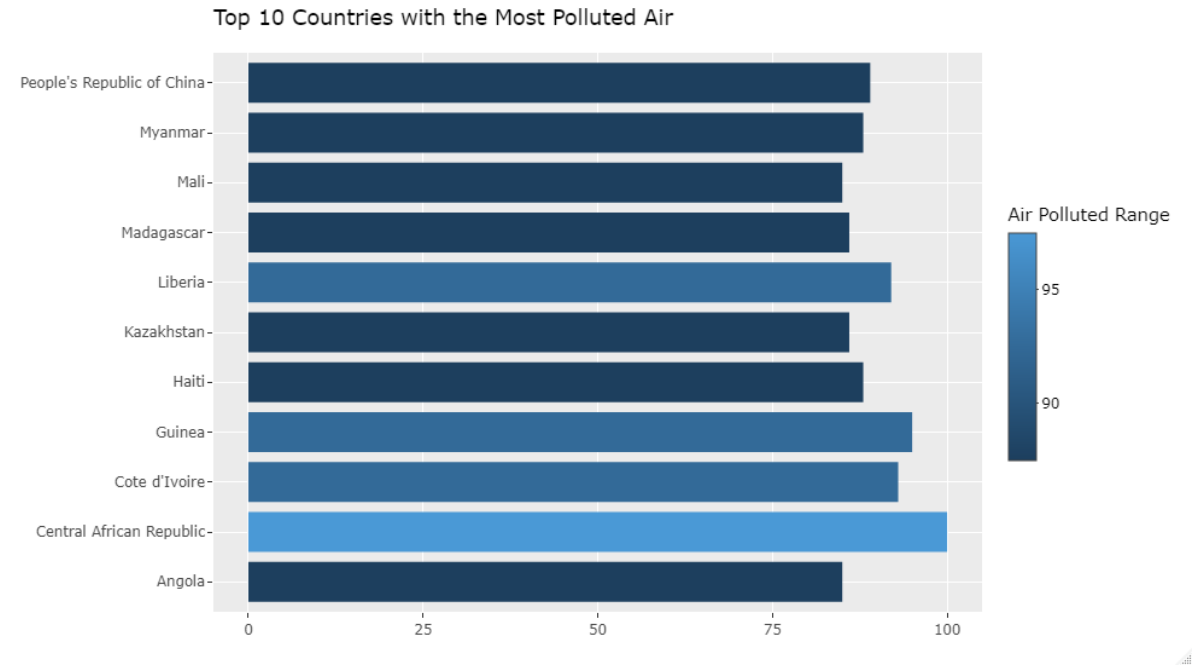

Figure 5.5: Data Visualisation of Top 10 Countries Hold the Most Polluted Air (top10_polluted_air.png)

In Figure 5.5, this visualisation shows top 10 countries with the most polluted air, the range is indicated from dark blue to blue. As the polluted air varies from 0 (good) to 100 (extremely bad) which related to the scales shown. Value will be accurately shown from the dataframe to this visualisation by hovering it. We can know that Central African Republic has the highest air polluted points (extremely bad quality of air).

5.6. Top 10 Countries Hold the Most Polluted Water

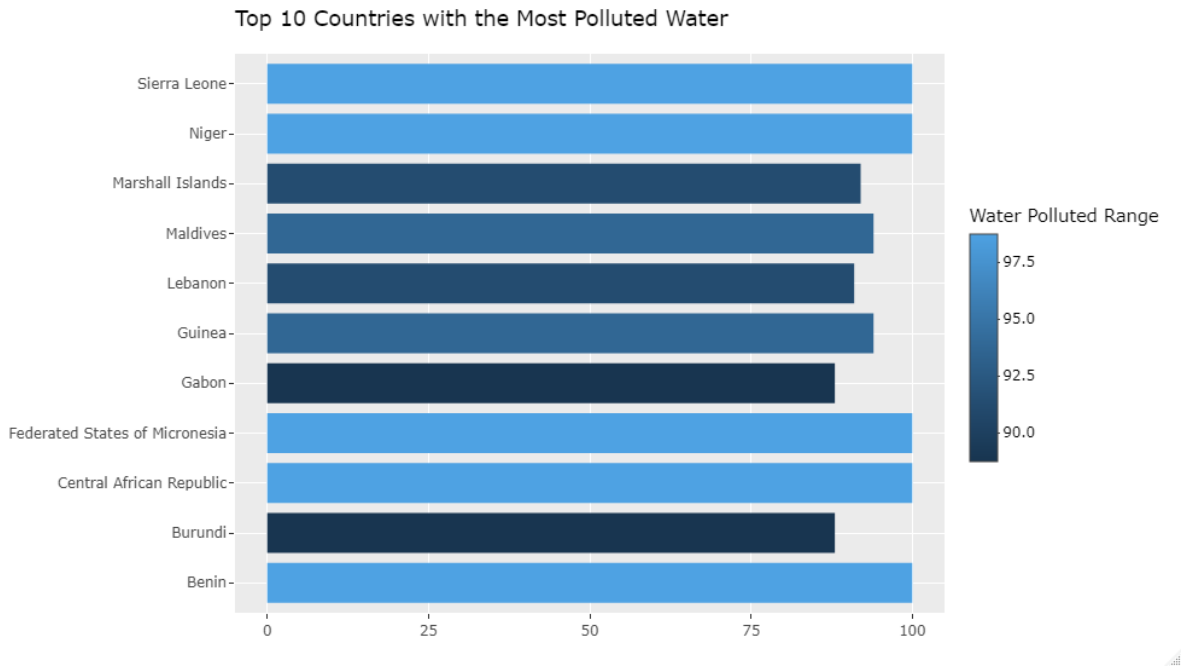

Figure 5.6: Data Visualisation of Top 10 Countries Hold the Most Polluted Water (top10_polluted_water.png)

In Figure 5.6, this visualisation shows top 10 countries with the most polluted water, the range is indicated from dark blue to blue. As the polluted water varies from 0 (good) to 100 (extremely bad) which related to the scales shown. Value will be accurately shown from the dataframe to this visualisation by hovering it. We can know that Sierra Leone, Niger, Federated States of Micronesia, Central African Republic, and Benin have the highest air polluted points (extremely bad quality of air).

6. EVALUATION

This system can be run under around a total of 10 seconds to make all objects ready to show by using view() functions shown in Figure 6.1.

```
> time.taken <- end.time - start.time
> time.taken
Time difference of 10.82073 secs
>
```

Figure 6.1: Running time on R file

This system must run R code manually from the file provided and view the visulisation by using view(*variable*) shown in Figure 6.2

```
view(aqdata)
```

Figure 6.2: Sample of viewing visualisations

7. DISCUSSIONS

In the data visualisation shown in R and presented in map and bar charts. In these maps, the values of water and air pollution are shown clearly in every region by hovering the regions in the maps displayed. The colour can be easily categorized the whole map into several places that will be easier to differentiate. For bar charts, countries with the most values are leading to the top 10 list in these charts. The plots are clearly inside the bar when hovering the value will display out as the water and air pollution points and the top 10 countries will be indicated clearly with different level of colors.

8. CONCLUSIONS AND FUTURE WORK

By using visualisation to present the water and air pollution across the world, these can remind of users to take note and beware of taking less impact to environment[5], and these are map with a several bar charts to show top 10 polluted or cleanest air or water across the world.

Things can be extended and improved more in a several ways. The first thing is to make it more interactive on changing colours bu adjusting scales to a more user friendly and let users to choose their favourite colour. Extended things like input values manually to update the newest value is also encouraged to do so to make things more up to date. The next thing that can be improved is ararnging bar values in charts to a more ordered way. This could make it tidier and easily to indiciate the most to the least. Finally, the code can be improved more to make execution time shorter, and users won't need to wait for a longer time.

9. BIBLIOGRAPHY

[1] Bishoi, Biswanath, Amit Prakash, and V. K. Jain. "A comparative study of air quality index based on factor analysis and US-EPA methods for an urban environment." *Aerosol and Air Quality Research* 9.1 (2009): 1-17.Sam Anzaroot and Andrew McCallum. 2013. UMass Citation Field Extraction Dataset. Retrieved May 27, 2019 from http://www.iesl.cs.umass.edu/data/data-umasscitationfield

[2] Astoreca, Rosa, Veronique Rousseau, and Christiane Lancelot. "Coloured dissolved organic matter (CDOM) in Southern North Sea waters: Optical characterization and possible origin." *Estuarine, Coastal and Shelf Science* 85.4 (2009): 633-640. Jon M. Kleinberg. 1999. Authoritative sources in a hyperlinked environment. J. ACM 46, 5 (September 1999), 604–632. https://doi.org/10.1145/324133.324140

[3] Themis, Kavour. 2021 November. cities_air_quality_water_pollution. Retrieved 19 April 2022 from%20https:/www.kaggle.com/code/themiskavour/eda-air-and-water-quality/notebook

[4] CITY-API-IO. 2021 November. cities_air_quality_water_pollution. Retrieved 19 April 2022 https://www.kaggle.com/datasets/cityapiio/world-cities-air-quality-and-water-polution

Sarkar, Deepayan. *Lattice: multivariate data visualization with R*. Springer Science & Business Media, 2008.

Wickham, Hadley. *ggplot2: elegant graphics for data analysis*. springer, 2016.

Wickham, Hadley. "Data manipulation with dplyr." *R user conference*. Vol. 30. 2014.

A APPENDICES

A.1 General Guidelines

1. Download R and csv file from github repositories (http://https://github.com/WorkingSteve/Information-Visualisation-Project).
2. Open R file in R Studio or any IDE that supports R language.
3. Edit the directory for reading csv file that you downloaded at.
4. Install necessary libraries.
5. Load libraries.
6. Run the rest of the codes.
7. View dataframes using view().
8. Only execute the lines with ggplotly() after everything is executed to show visualisations.