FIT 3152
Assignment 01
33370311

Generative AI was used in this assignment,

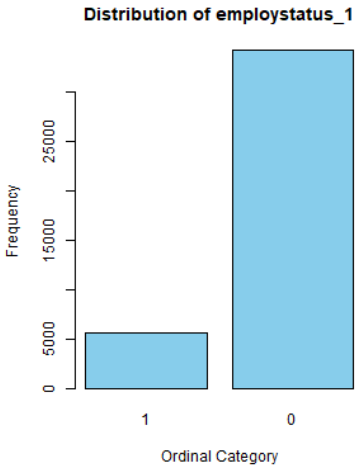
Question 1

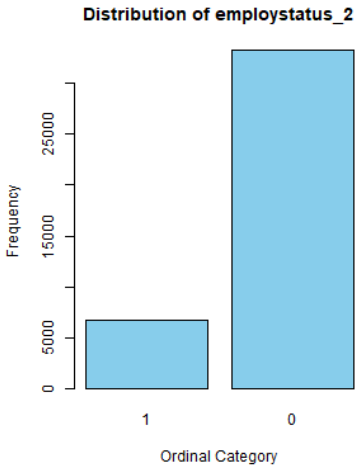
(a) The provided csv for analysis has 40000 rows with 52 columns consisting of data from 110 unique countries. The data has large variety of numerical attributes and few non numerical attributes following is our terminal output

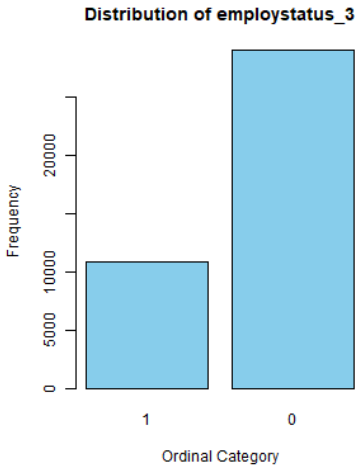
The Numeric Attributes: *employstatus_1 employstatus_2 employstatus_3 employstatus_4 employstatus_5 employstatus_6 employstatus_7 employstatus_8 employstatus_9 employstatus_10 isoFriends_inPerson isoOthPpl_inPerson isoFriends_online isoOthPpl_online lone01 lone02 lone03 happy lifeSat MLQ bor01 bor02 bor03 consp01 consp02 consp03 c19perBeh01 c19perBeh02 c19perBeh03 c19RCA01 c19RCA02 c19RCA03 coronaClose_1 coronaClose_2 coronaClose_3 coronaClose_4 coronaClose_5 coronaClose_6 gender age edu c19ProSo01 c19ProSo02 c19ProSo03 c19ProSo04*

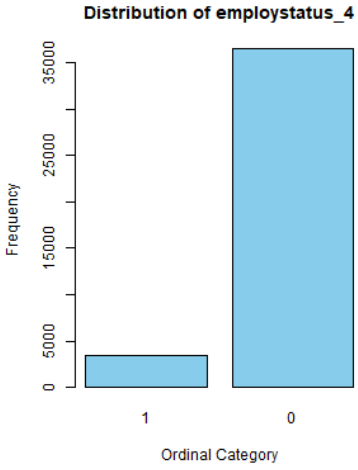
The Non-Numeric Attributes: *rankOrdLife_1 rankOrdLife_2 rankOrdLife_3 rankOrdLife_4 rankOrdLife_5 rankOrdLife_6 coded_country*
The data has a lot of NA values. Notably, attributes related to respondents' perceptions and behaviours during the COVID-19 pandemic, such as coronaClose_1 to coronaClose_6, exhibit high counts of missing values, ranging from 10,805 to 39,436. Demographic attributes like age, gender, and edu also show notable numbers of missing values, with 236, 214, and 280 missing values, respectively. Additionally, attributes related to psychological factors (happy, lifeSat, MLQ) and conspiracy beliefs (bor01 to bor03, consp01 to consp03) demonstrate varying counts of missing values, ranging from 82 to 1,571. Employment status attributes (employstatus_1 to employstatus_10) also exhibit a range of missing values, with the highest count in employstatus_1 (34,300 missing values) and the lowest count in employstatus_3 (29,057 missing values).

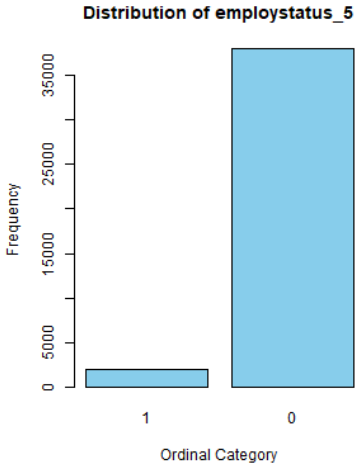
Now we will discuss the distribution of numerical attributes,
If you see the plots below for employment status variables we see a lot of values are zero or NA, now one of the relevant thing I would like to mention is that the NA values simply does not mean they are meaning less here but rather if one belongs to particular status it would be one if not then NA so 0 acts like a no here. The visualisation provides us with an insight that most of the participants work 40 hours or more (employstatus_3) or are students (employstatus_9)

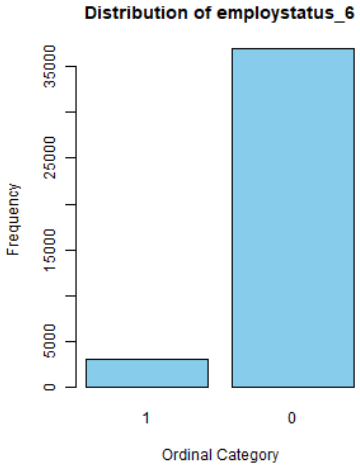

Distribution of employstatus_1
Frequency
0
5000
15000
25000
1
0
Ordinal Category

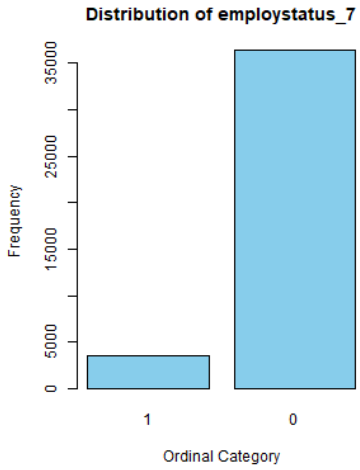

Distribution of employstatus_2
Frequency
0
5000
15000
25000
1
0
Ordinal Category

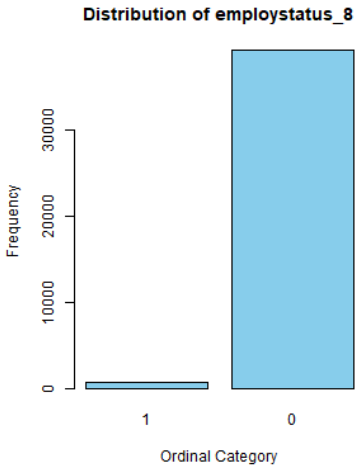

Distribution of employstatus_3
Frequency
0
5000
10000
20000
1
0
Ordinal Category

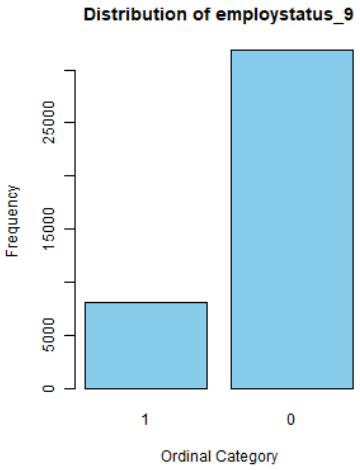

Distribution of employstatus_4
Frequency
0
5000
15000
25000
35000
1
0
Ordinal Category

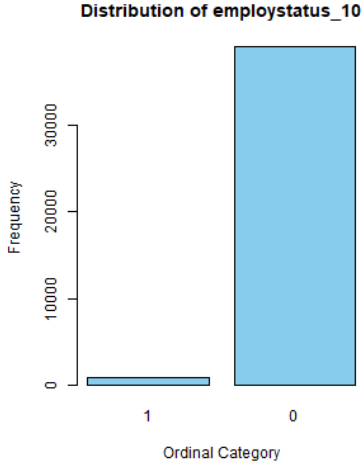

Distribution of employstatus_5
Frequency
0
5000
15000
25000
35000
1
0
Ordinal Category

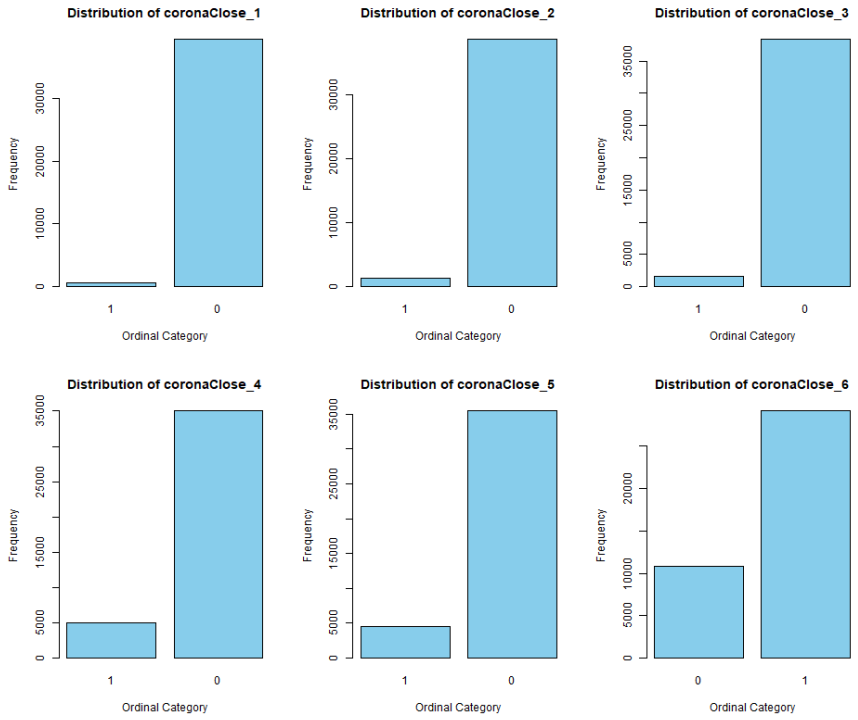

Distribution of employstatus_6
Frequency
0
5000
15000
25000
35000
1
0
Ordinal Category


Distribution of employstatus_7
Frequency
0
5000
15000
25000
35000
1
0
Ordinal Category

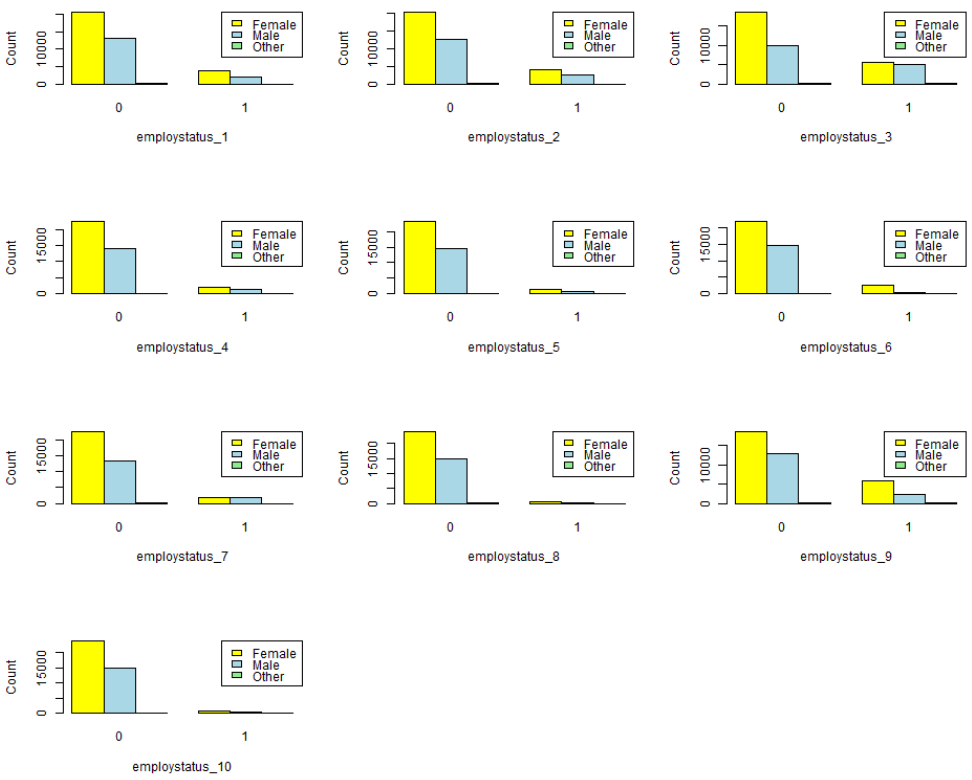

Distribution of employstatus_8
Frequency
0
10000
20000
30000
1
0
Ordinal Category


Distribution of employstatus_9
Frequency
0
5000
15000
25000
1
0
Ordinal Category


Distribution of employstatus_10
Frequency
0
10000
20000
30000
1
0
Ordinal Category

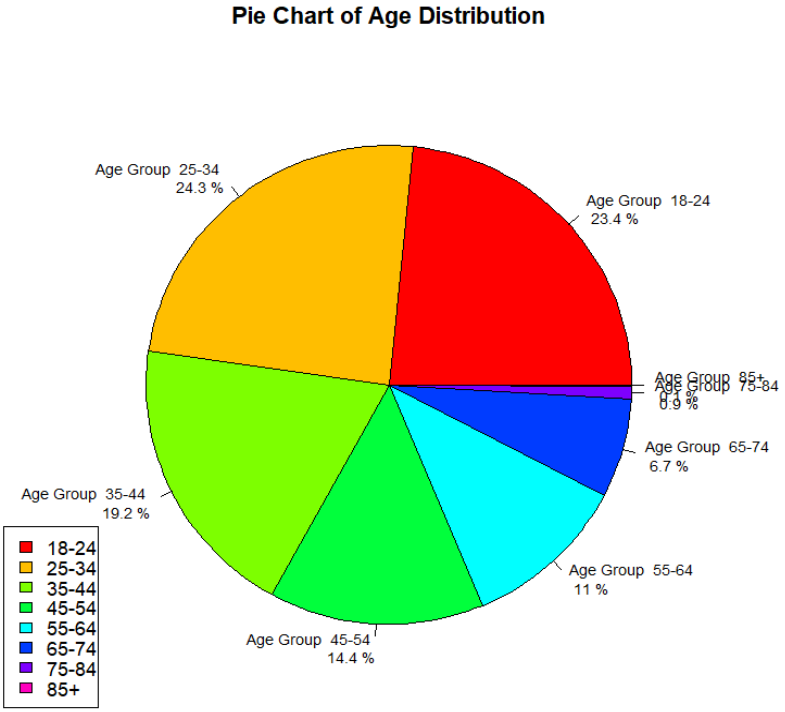


(Figure b)



(Figure c)

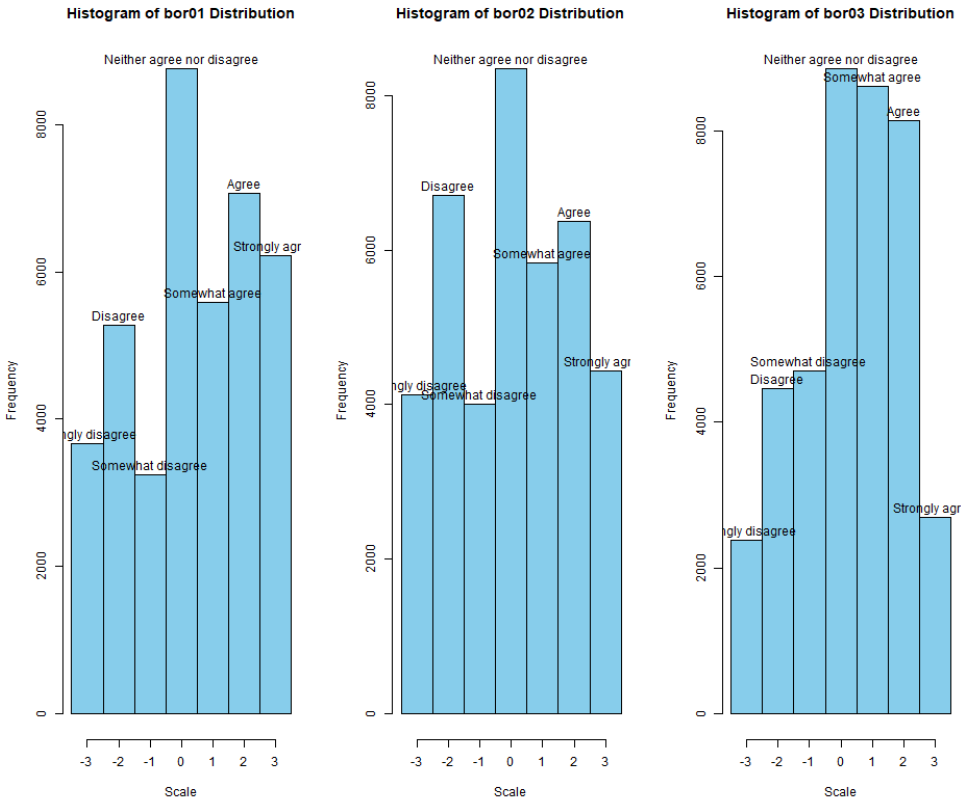
Same as employment status if refer **(Figure b)** we see that in our responses many participants knew no one close who had coronavirus. Moving on to more depth we see a lot **(Figure c)** of female participants in our dataset.



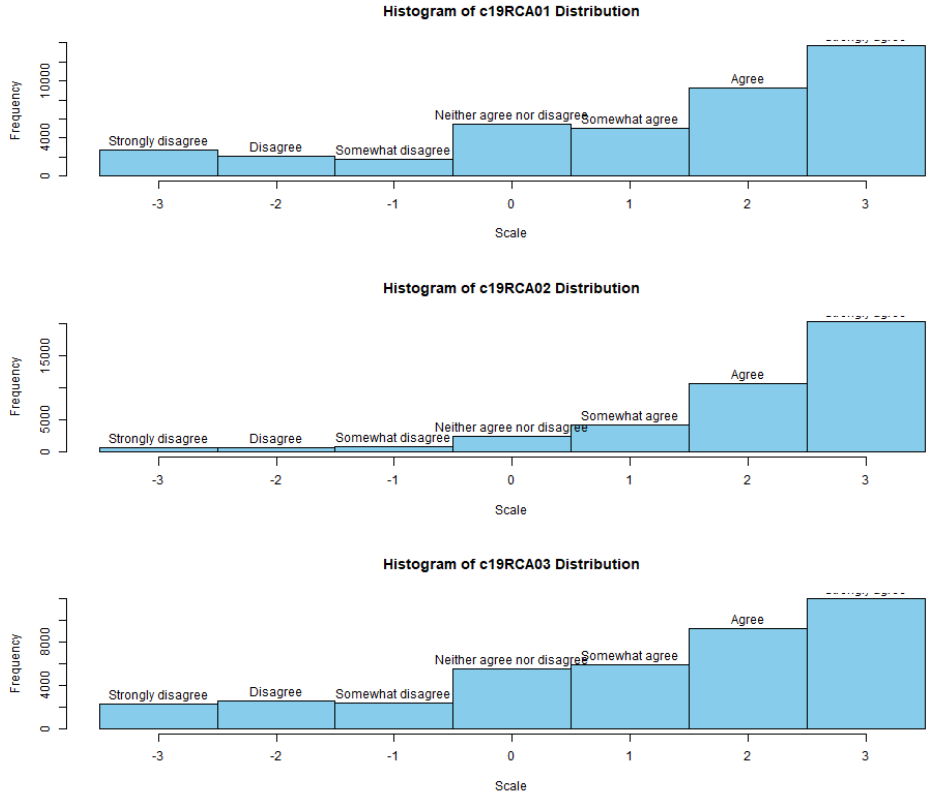
(Figure d)

Through age distribution in **(Figure d)** we can see that our many participants are from the age group of 25 -34 and least are 85+
Below are some self explanatory distributions where I have shown each attribute distribution with proper labelling.

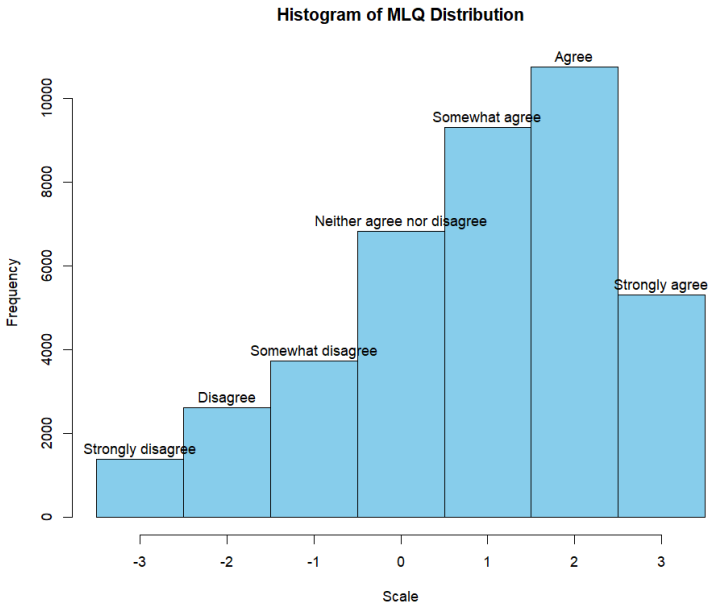
For our numerical attributes for Boredom we get this distribution **(Figure e)**
For our numerical attribute distribution for Corona RadicalAction **(Figure f)**
For our numerical attribute distribution for MLQ **(Figure g)**
For our numerical attribute distribution for Life Satisfaction **(Figure g)**
For our numerical attribute distribution for Loneliness **(Figure h)**
For our numerical attribute distribution for Isolation Online**(Figure i)**

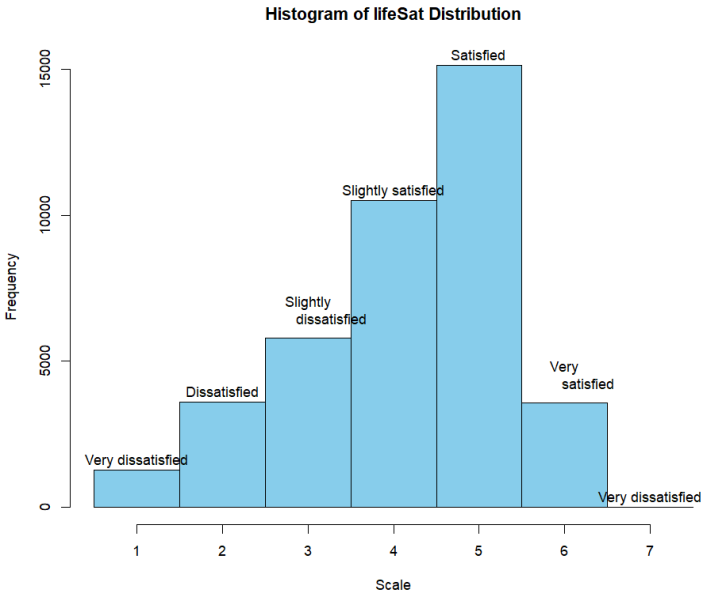


(Figure e)

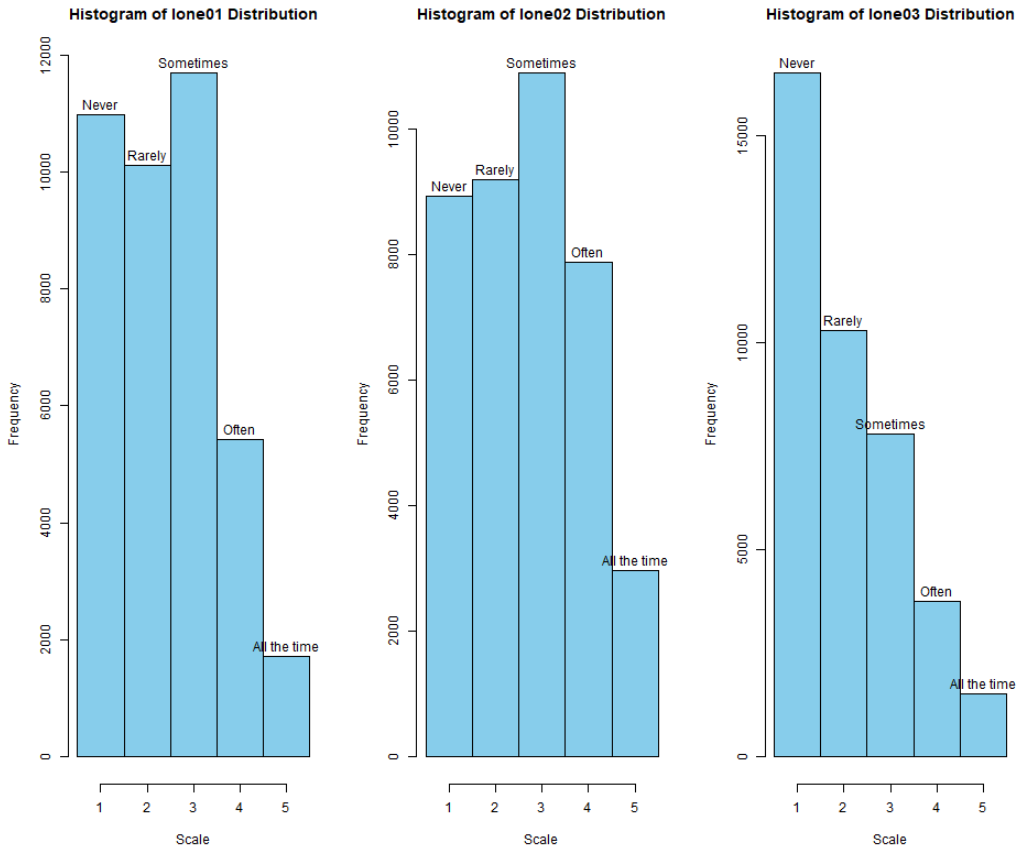


(Figure f)

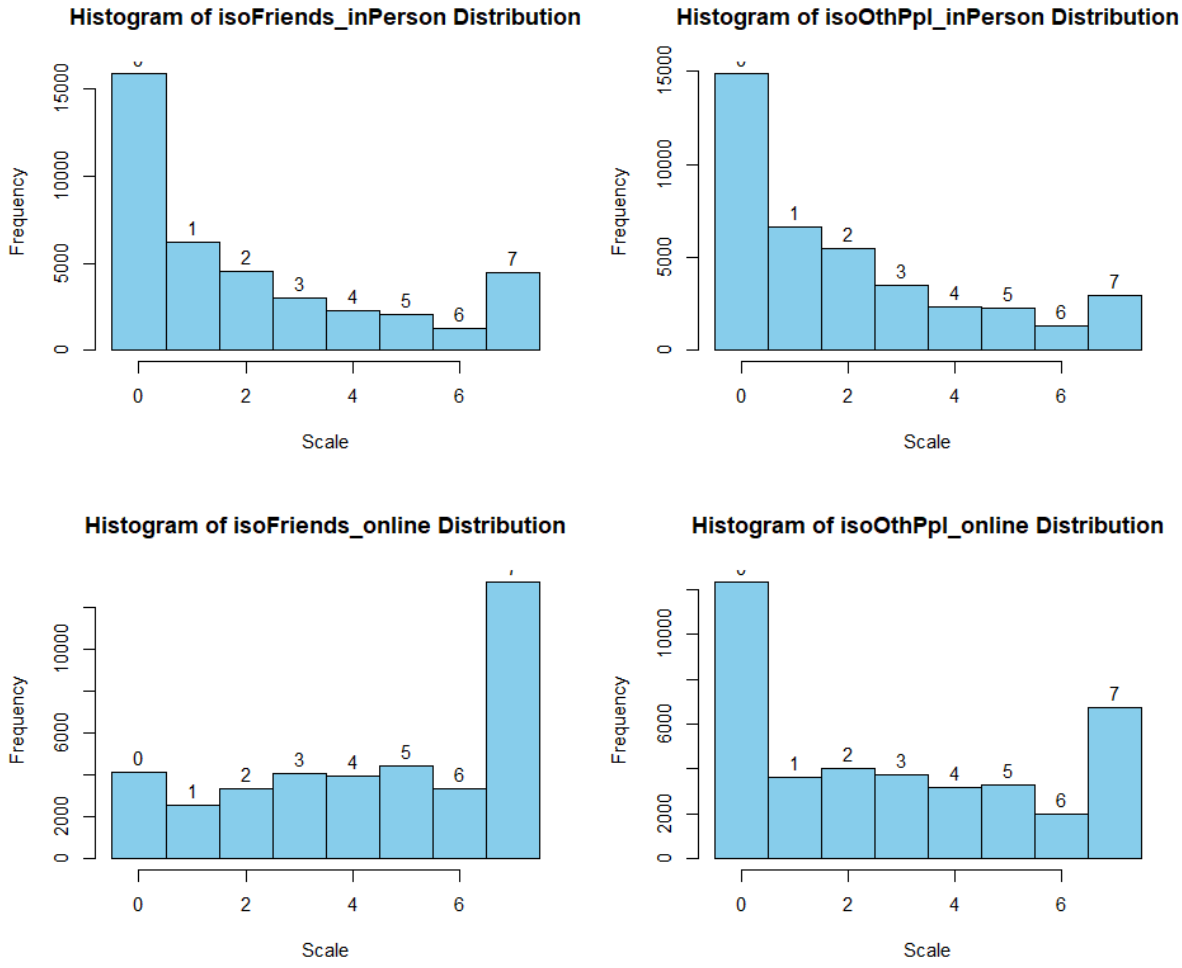




(Figure g)



(Figure h)



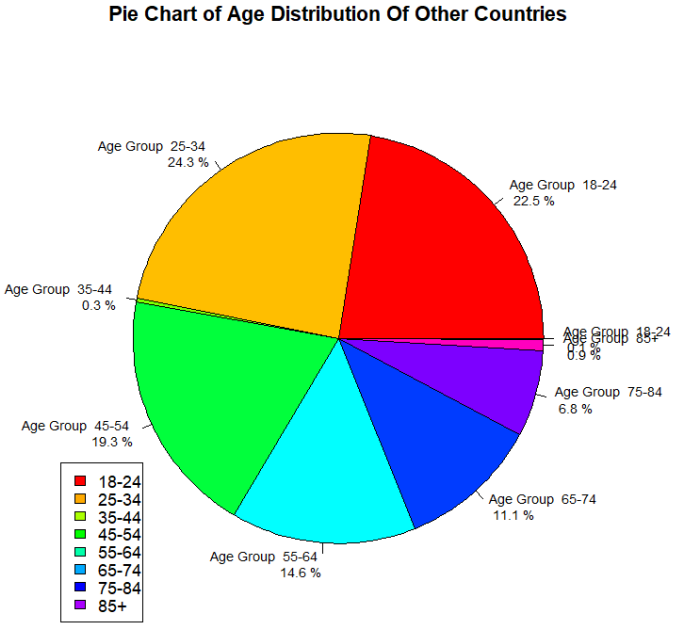
(Figure i)

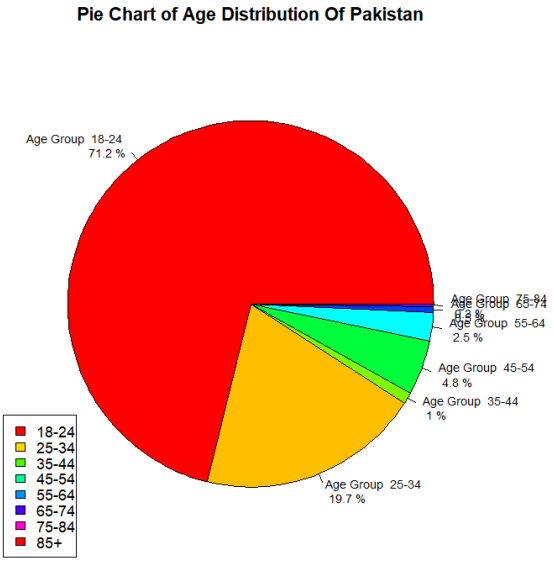
b)

For the analysis provided in part (a), several pre-processing steps and data manipulations were necessary to ensure the accuracy and reliability of the results. Initially, I randomly sampled 40,000 rows from the dataset to facilitate efficient computation while maintaining a representative sample size. Additionally, I replaced any missing values (NA) in the employment status columns and corona proximity columns with zeros, as these variables likely indicate absence or lack of response rather than true missing values. This step ensured consistency in subsequent analyses involving these columns and I would have rather used omit I would lose half of my data. Furthermore, I created customised histograms to visualise the distribution of various attributes, such as employment status, corona-related behaviours, boredom levels, life satisfaction, and loneliness, among others. These histograms were tailored to represent ordinal categorical variables appropriately, utilising defined breaks and labels to accurately depict the underlying data distributions. Overall, these pre-processing and data manipulation steps were essential to ensure the validity and interpretability of the subsequent analyses.

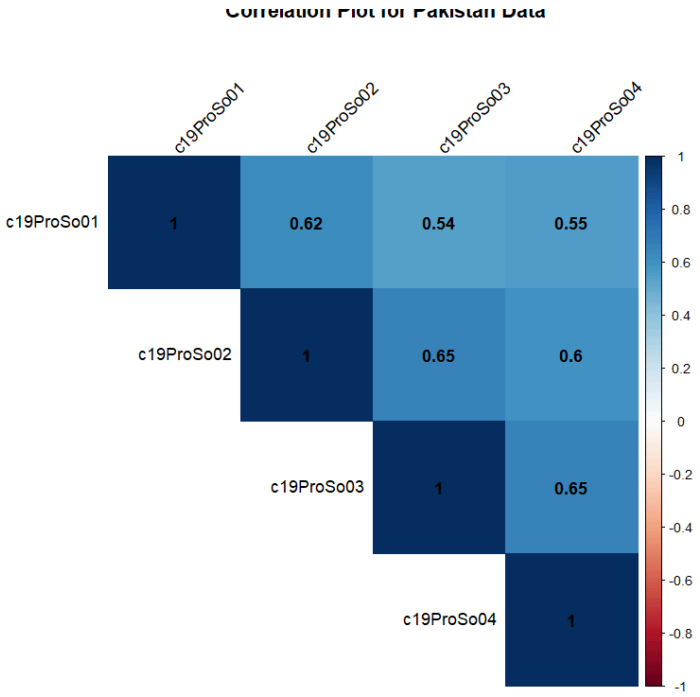
Question 2

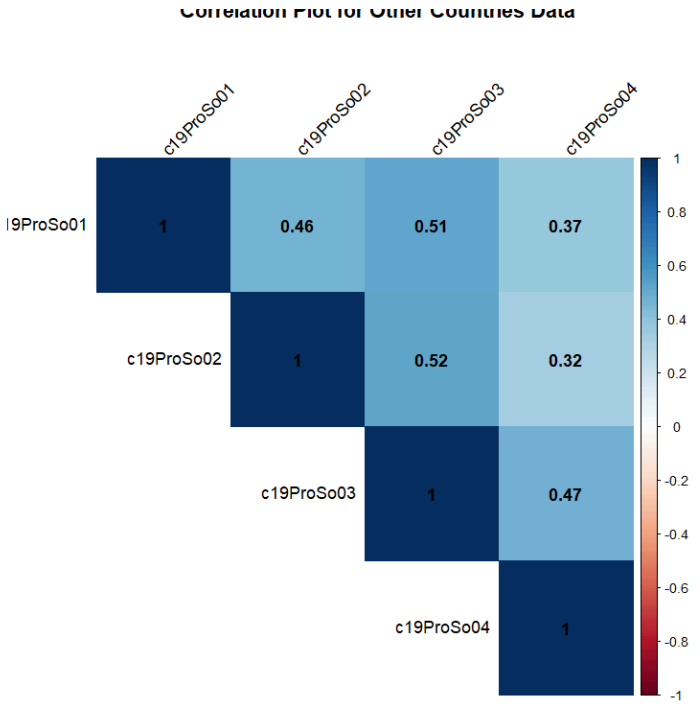
a)
As per getting the summary of each dataset and max occurrence The summary statistics for Pakistan and other countries reveal some interesting differences. In Pakistan, the mean values for employment status indicators such as employment status 1, 2, and 3 are relatively lower compared to other countries. However, Pakistan shows a higher mean for employment status 9, indicating a higher percentage of respondents being unemployed but actively seeking employment. Regarding social behaviour during the COVID-19 pandemic, Pakistan reports slightly higher mean values for in-person interaction with friends and acquaintances compared to other countries. In terms of loneliness indicators (lone01, lone02, lone03), Pakistan tends to report slightly lower mean values compared to other countries, suggesting potentially lower levels of loneliness. Additionally, Pakistan reports higher mean values for life satisfaction as the most occurring value in Pakistan for life satisfaction is 7 very satisfied and happiness compared to other countries which is 5 means slightly satisfied.



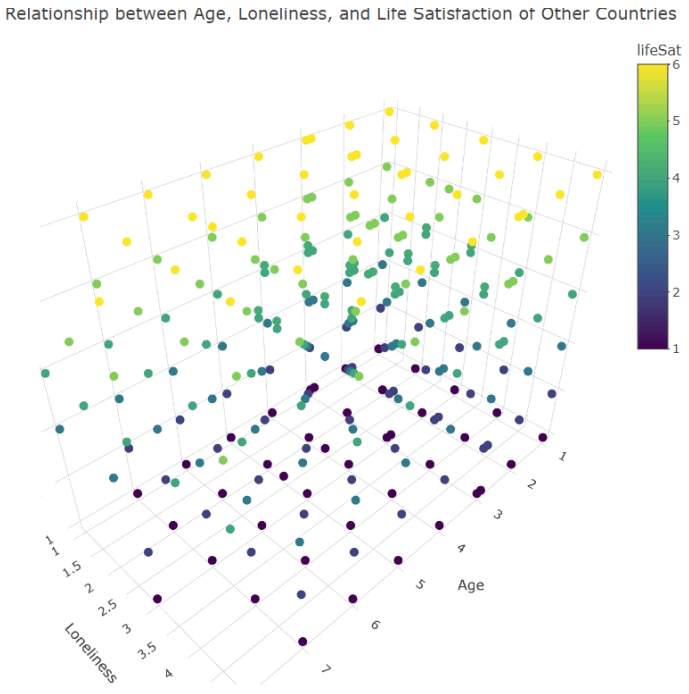


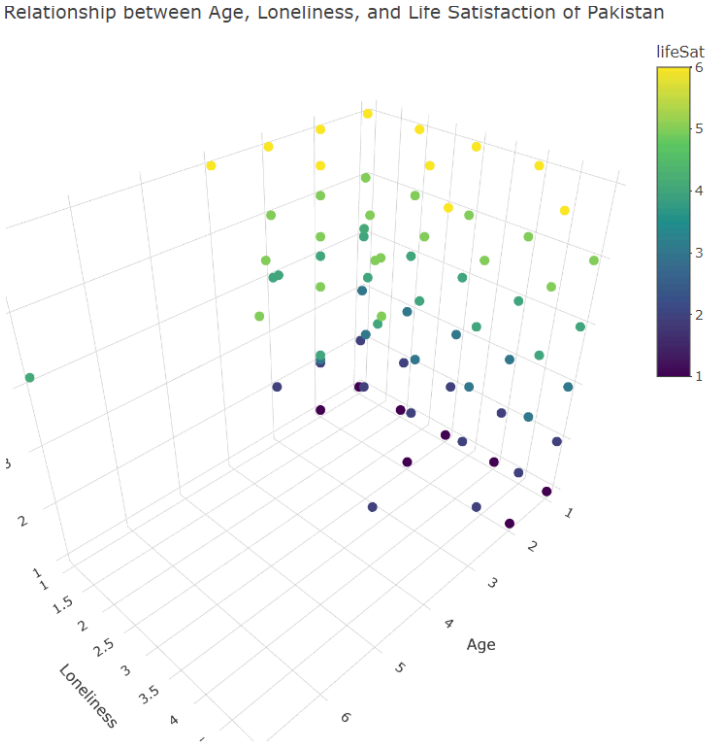
if you notice the pie charts Pakistan"s most of the respondents are from the age group of 18-24
Below you will find the correlation difference of both the datasets between Corona ProSocial Behaviour attributes





even though the Pakistan dataset is less and undoubtedly we have good correlation but this matrix gives us a notable difference which is that both datasets do not follow the same pattern because the least correlated in other countries is c19ProSo01 and c19ProSo04 but in Pakistan least correlated are c19ProSo01 and c19ProSo03.





Since this is just an image so its hard to compare because I have used plotly and it gives you analysis on just the plot itself but both the 3d scatter plots provide us with insights into the loneliness(lone01) and life satisfaction of the respondents based on their age so we see that most of the participants belonging from age group 2 have life satisfaction of 1 in both the plots similarly most of the participants from age group 1 have life satisfaction of 6, also quite interesting that mostly in both the scatterplots we see yellow(life satisfaction of 6) means satisfied with their life have loneliness value of 1 which means they never felt lonely. If we move on to the differences we see that most of the respondents in Pakistan's data are from the age group 2 or 1. We can observe that participants belonging to age group 7 (75-85) are not satisfied with their life by see most purple dots.

b)
The linear regression model summary for predicting c19ProSo01 in Pakistan reveals several important attributes that contribute to explaining variations in pro-social attitudes during the COVID-19 pandemic.

c19ProSo02, c19ProSo03, c19ProSo04: These attributes all exhibit significant positive coefficients, indicating strong predictors of pro-social attitudes (c19ProSo01). Higher scores on c19ProSo02, c19ProSo03, and c19ProSo04 correspond to more positive pro-social attitudes.
employstatus_8: This attribute has a notably negative coefficient, implying that being from employstatus_8 means less positive pro-social attitudes during the pandemic.
lone01: With a positive coefficient, lone01 emerges as a significant predictor, indicating that feelings of loneliness are associated with more positive pro-social attitudes.
c19RCA01: Similar to the previous model, c19RCA01 displays a positive coefficient, suggesting that higher levels of perceived risk are associated with more positive pro-social attitudes.
bor03:bor03 shows a negative coefficient, indicating a decrease in pro-social attitudes with higher participants who have higher control of time.
The linear regression model summary for predicting c19ProSo02 in Pakistan reveals several important attributes that contribute to explaining variations in pro-social attitudes during the COVID-19 pandemic.
c19ProSo01, c19ProSo03, c19ProSo04: These attributes all exhibit significant positive coefficients, indicating strong predictors of pro-social attitudes (c19ProSo02). Higher scores on c19ProSo01, c19ProSo03, and c19ProSo04 correspond to more positive pro-social attitudes.
age: age attribute has also been significant but due to negative coefficient.
The linear regression model summary for predicting c19ProSo03 in Pakistan reveals several important attributes that contribute to explaining variations in pro-social attitudes during the COVID-19 pandemic.
c19ProSo01, c19ProSo02, c19ProSo04: These attributes all exhibit significant positive coefficients, indicating strong predictors of pro-social attitudes (c19ProSo03). Higher scores on c19ProSo01, c19ProSo02, and c19ProSo04 correspond to more positive pro-social attitudes.
employstaus_9: This attribute has a notably negative coefficient, implying that being from employstatus_9 means less positive pro-social attitudes during the pandemic.

The linear regression model summary for predicting c19ProSo04 in Pakistan reveals several important attributes that contribute to explaining variations in pro-social attitudes during the COVID-19 pandemic.
c19ProSo01, c19ProSo03, c19ProSo02: These attributes all exhibit significant positive coefficients, indicating strong predictors of pro-social attitudes (c19ProSo04). Higher scores on c19ProSo01, c19ProSo03, and c19ProSo02 correspond to more positive pro-social attitudes.
employstaus_5: This attribute has a notably negative coefficient, implying that being from employstatus_5 means less positive pro-social attitudes during the pandemic.
For all the models I looked at the coefficients value and p_value to determine their significance.

c)

Model for c19ProSo01:
The regression analysis for c19ProSo01 in other_countries_data highlights several significant predictors. Notably, employment status categories 4, 5, 6, 7, 9, and 10 show significant associations with c19ProSo01, with category 10 having the strongest positive association. Additionally, positive perceptions and behaviours related to COVID-19 (c19perBeh01, c19perBeh02, and c19perBeh03) demonstrate strong positive associations with c19ProSo01 scores. Variables such as MLQ and bor03 also exhibit significant positive associations, while variables like lone01 and lone02 show significant negative associations. Demographic factors such as age and gender are also significant predictors, with gender showing a notable positive association. However, some variables such as rankOrdLife_12 and edu do not appear to be significant predictors. Overall, the model explains a moderate amount of variance in c19ProSo01 (Adjusted R-squared = 0.3491), and the F-statistic indicates the overall significance of the regression model ($p < 0.001$), suggesting that the included predictors collectively contribute to explaining the variability in c19ProSo01 scores among individuals in other countries.

Model for c19ProSo02:
The regression analysis for c19ProSo02 in other_countries_data reveals several significant predictors. Employment status categories 2, 3, 4, 5, 6, and 8 exhibit significant associations with c19ProSo02, with category 4 having the most substantial negative association. In terms of social interactions during the pandemic, in-person interactions with friends and other people show significant associations, while online interactions with friends also have a significant positive association. Psychological factors like loneliness (lone01 and lone02), happiness, and life satisfaction demonstrate notable associations with c19ProSo02, as does the Multidimensional Leadership Questionnaire (MLQ). COVID-19 perception and behaviour variables (c19perBeh01, c19perBeh03) also show significant positive associations, as well as variables related to risk perception (bor01 and bor02) and conspiracy beliefs (consp01 and consp03). Demographic factors such as age and education level are significant predictors, with education level showing a particularly strong positive association. Interestingly, variables related to the perceived closeness of COVID-19 are not consistently significant predictors. Overall, the model explains a moderate amount of variance in

c19ProSo02 (Adjusted R-squared = 0.3835), and the F-statistic indicates the overall significance of the regression model ($p < 0.001$), suggesting that the included predictors collectively contribute to explaining the variability in c19ProSo02 scores among individuals in other countries.

Model for c19ProSo03:

The regression analysis for c19ProSo03 in other_countries_data indicates several significant predictors. Notably, employment status categories 3, 7, and 26 display significant associations with c19ProSo03. In-person interactions with friends and other people also exhibit significant associations, as well as online interactions with other people. Psychological factors like loneliness (lone02 and lone03) and life satisfaction show significant associations with c19ProSo03. Beliefs related to conspiracy theories (consp01, consp02, and consp03) demonstrate significant associations, as well as variables related to risk perception (bor02 and bor03). Similarly, certain ordinal life ranking categories show significant associations with c19ProSo03. COVID-19 perception and behavior variables (c19perBeh01 and c19perBeh03) also show significant associations. Demographic factors such as age and education level are significant predictors, with age displaying a particularly strong negative association. Interestingly, the perception of the closeness of COVID-19 does not consistently emerge as a significant predictor. Overall, the model explains a considerable amount of variance in c19ProSo03 (Adjusted R-squared = 0.442), and the F-statistic indicates the overall significance of the regression model ($p < 0.001$), suggesting that the included predictors collectively contribute to explaining the variability in c19ProSo03 scores among individuals in other countries.

The model for c19ProSo04:

The regression analysis for c19ProSo04 in other_countries_data reveals several significant predictors. Employment status categories 1, 2, 3, 4, 5, 7, 8, and 10 display significant associations with c19ProSo04. In-person interactions with friends and online interactions with friends both show significant associations, while in-person interactions with other people and online interactions with other people only exhibit significant associations in the case of online interactions with other people. Loneliness (lone01, lone02, and lone03) also displays significant associations with c19ProSo04, as does life satisfaction. Some beliefs related to conspiracy theories (consp01, consp02, and consp03) show significant associations, as well as variables related to risk perception (bor01, bor02, and bor03). Certain ordinal life ranking categories also show significant associations with c19ProSo04. COVID-19 perception and behaviour variables (c19perBeh01, c19perBeh02, and c19perBeh03) exhibit significant associations, as well as variables related to COVID-19 risk perception and close contact (c19RCA01, c19RCA02, c19RCA03, and coronaClose_6). Age is a significant predictor, with a positive association with c19ProSo04. Variables related to happiness, gender, and education level do not consistently emerge as significant predictors. Overall, the model explains a significant amount of variance in c19ProSo04 (Adjusted R-squared = 0.3207), and the F-statistic indicates the overall significance of the regression model ($p < 0.001$), suggesting that the included predictors collectively contribute to explaining the variability in c19ProSo04 scores among individuals in other countries.

Question 3

a)

My focus country is Pakistan.

In figuring out which countries are similar to Pakistan, I took a thorough approach. I started by combining data from different sources, making sure everything matched up. This meant dealing with any missing info properly, so our data stayed reliable. Then, I looked at all sorts of indicators, like how corrupt a government is, how well it works, how stable the country is politically, and even stuff like immunisation and disease reporting. My health data was from a different file and the government performance data was from a different file but both had data of 2019 and 2021 so computed average values for each country across the years 2019 and 2021. and then I had to use a merge function to combine both the files after handling missing values and changing column names as per my need.

These are the names of my columns

[1] "Country"
[2] "Control of Corruption Estimate"
[3] "Government Effectiveness Estimate"
[4] "Political Stability and Absence of Terrorism Estimate"
[5] "Immunization"
[6] "Vaccination_rates"
[7] "Overall_Health_Score"
*[8]"EARLY_DETECTION/*REPORTING.FOR.EPIDEMICS.OF.POTENTIAL.INT.L.CONCERN"

Once I had all that data together, I used a method called k-means clustering. It's like sorting countries into groups based on how alike they are when you look at all those different indicators. To decide how many groups to make, I used something called the elbow method. It showed that five groups would work best for what we needed.

After sorting the countries into their groups, I checked which ones were in the same group as Pakistan. These are the countries that have similar situations across all those different indicators. It gives a good picture of where Pakistan stands compared to its peers.

Countries similar to Pakistan include Afghanistan, Algeria, Angola, Benin, Bolivia, Burkina Faso, Burundi, Cambodia, Cameroon, Central African Republic, Chad, Comoros, Djibouti, Dominican Republic, Eritrea, Ethiopia, Guatemala, Guinea, Guinea-Bissau, Haiti, Honduras, Iraq, Lebanon, Lesotho, Libya, Madagascar, Malawi, Mali, Mozambique, Myanmar, Nepal, Niger, Nigeria, Papua New Guinea, Somalia, South Sudan, Sudan, Suriname, Tajikistan, Tanzania, Togo, Uganda, Ukraine, and Zimbabwe.

b)

The linear regression model for c19ProSo01 in similar countries data reveals several significant predictors. Notably, employment status emerges as a significant predictor, with employstatus_10 showing a positive association with c19ProSo01 (Estimate = 0.7927, $p < 0.01$). Additionally, the MLQ variable is significant (Estimate = 0.0956, $p < 0.01$). Among the variables related to COVID-19 behaviours, c19ProSo02 (Estimate = 0.2241, $p < 0.001$), c19ProSo03 (Estimate = 0.1974, $p < 0.001$), and c19ProSo04 (Estimate = 0.1743, $p < 0.001$) all show significant positive associations with c19ProSo01.

The linear regression model for c19ProSo02 in similar countries data reveals several significant predictors. Notably, c19ProSo01 (Estimate = 0.1853, $p < 0.001$), c19ProSo03 (Estimate = 0.4640, $p < 0.001$), and c19ProSo04 (Estimate = 0.1244, $p < 0.001$) all show significant positive associations with c19ProSo02. Among the socio-demographic variables,

age is significant (Estimate = -0.0851, p = 0.015), suggesting that younger individuals may perceive lower levels of social support during the pandemic compared to older individuals. Additionally, employstatus_5 (Estimate = -0.3908, p = 0.0348) shows a significant negative association with c19ProSo02.

The regression analysis for c19ProSo03 in the dataset of similar countries yields some significant predictors. Notably, c19ProSo01 (Estimate = 0.1541, p < 0.001), c19ProSo02 (Estimate = 0.4382, p < 0.001), and c19ProSo04 (Estimate = 0.2477, p < 0.001) all demonstrate significant positive associations with c19ProSo03. Among the socio-demographic variables, education (Estimate = 0.0626, p = 0.023) emerges as significant, suggesting that higher levels of education are associated with higher perceptions of social support during the pandemic. Other variables such as age, employstatus, and gender do not appear to have significant associations with c19ProSo03 in this analysis. Overall, the model explains a moderate amount of variance in c19ProSo03 (Adjusted R-squared = 0.5506) and the F-statistic indicates that the overall regression model is significant (p < 0.001).

The regression analysis for c19ProSo04 reveals several significant predictors. Notably, positive perceptions and behaviours related to COVID-19, represented by c19ProSo01, c19ProSo02, and c19ProSo03, demonstrate strong positive associations with c19ProSo04. Education also shows a weak negative association, while certain categories of rankOrdLife_15, rankOrdLife_54, and c19RCA02 display significant associations with c19ProSo04. However, socio-demographic variables such as age, employment status, and gender do not appear to be significant predictors. Overall, the model explains a moderate amount of variance in c19ProSo04 (Adjusted R-squared = 0.4029), and the F-statistic indicates the overall significance of the regression model (p < 0.001).

The regression analysis for perceived social support (c19ProSo04) in Pakistan and similar countries reveals notable differences and similarities. In Pakistan, employment status (employstatus_5), age, and COVID-19 related factors (c19ProSo01, c19ProSo02, c19ProSo03) emerge as significant predictors. However, in similar countries, employment status (employstatus_5) is not significant, while variables like happiness (happy), perceived life satisfaction (lifeSat), and certain COVID-19 related factors (c19ProSo01, c19ProSo02, c19ProSo03) play significant roles. Additionally, gender and education are significant predictors in similar countries but not in Pakistan.

The regression analysis reveals differences in predictors of perceived social support (c19ProSo03) between similar countries and Pakistan. In similar countries, variables such as employment status (employstatus_8), happiness (happy), and education (edu) play significant or marginally significant roles alongside COVID-19 related factors (c19ProSo01, c19ProSo02, c19ProSo04). However, in Pakistan, employment status (employstatus_9) emerges as a significant predictor alongside COVID-19 related factors, while other predictors like age and education are not significant.

The comparison between similar countries and Pakistan also reveals differences in the attributes that play a significant role in predicting perceived social support (c19ProSo02). In the model for similar countries, variables such as age and lifeSat are significant predictors of perceived social support, with age showing a negative coefficient and lifeSat showing a positive coefficient. This suggests that older individuals in similar countries perceive less social support, while those with higher life satisfaction perceive more. However, in the Pakistan model, age emerges as a significant predictor with a negative coefficient, indicating that older individuals in Pakistan also perceive less social support. However, variables such as lifeSat and gender, which are significant in the similar countries model, are not significant

predictors in the Pakistan model. This suggests that factors influencing perceived social support may differ between similar countries and Pakistan, possibly due to cultural, social, or economic differences.

In comparing the significant predictors between the models for similar countries to Pakistan and Pakistan itself, several factors stand out. First, variables related to social behaviour during the pandemic, such as c19ProSo02, c19ProSo03, and c19ProSo04, are consistently significant in both models. These variables likely capture important aspects of individuals' responses to the pandemic, such as adherence to protective behaviours, which are crucial for understanding variations in perceived social support. Additionally, employment status (employstatus_10 in the similar countries model and lone01 in Pakistan's model) emerges as significant in both cases, underscoring the role of economic factors in shaping individuals' perceptions of social support during crises. Moreover, while life satisfaction (lifeSat) appears significant in the similar countries model, it loses significance in Pakistan's model, suggesting potential cultural or contextual differences in the importance of this factor. These comparisons highlight the robustness of certain predictors across contexts, such as social behaviours and employment status, while also indicating nuanced differences that may reflect unique cultural or socio-economic factors within Pakistan.

We have gone through all the details of the significant predictors of both similar countries and other countries and as per my analyses we see a lot of significant predictors in other countries model for Corona pro social behaviours rather than similar countries and Pakistan corona pro social behaviour model.

The differences observed between "other countries" and Pakistan, as well as the similarities between Pakistan and the identified "similar countries," can be attributed to a range of socio-economic, cultural, and contextual factors that influence the predictors of perceived social support during the COVID-19 pandemic.

In the analysis of "other countries," specific employment status categories (e.g., employstatus_10) emerge as significant predictors of perceived social support. This suggests that economic factors related to employment play a crucial role in shaping social support perceptions in these countries.

Conversely, in Pakistan and similar countries, different employment status categories (e.g., employstatus_5 in Pakistan) are significant predictors. This highlights potential differences in the employment landscape and its impact on social support perceptions within these specific contexts.

Variables like happiness and education exhibit significant associations with perceived social support in the regression models for similar countries but do not emerge as significant predictors in Pakistan.Age consistently shows a negative association with perceived social support in both "other countries" and Pakistan:

Variables related to COVID-19 perceptions and behaviours (c19ProSo01, c19ProSo02, c19ProSo03, c19ProSo04) consistently demonstrate significant associations with perceived social support across all analysed contexts. This underscores the overarching influence of pandemic-related experiences and responses on individuals' social support perceptions worldwide. The differences in significant predictors between "other countries" and Pakistan highlight unique contextual nuances. While employment status remains influential across both contexts, other predictors like happiness and education vary in their significance, reflecting distinct socio-cultural landscapes.

APPENDIX

Generative AI use 1

Technology used was Chatgpt3.5
Question: best method to check the accuracy of my clustering be very precise
Answer: The elbow method is a technique used to determine the optimal number of clusters in a dataset for K-means clustering. It involves plotting the variance explained as a function of the number of clusters, and identifying the "elbow" point where the rate of decrease in variance slows down significantly.

Here's how to use the elbow method:

Perform K-means clustering on the dataset for a range of cluster numbers (typically from 1 to some maximum value).
For each clustering result, calculate the total within-cluster sum of squares (WCSS), which is a measure of the variability within each cluster.
Plot the number of clusters against the corresponding WCSS values.
Identify the point on the plot where the rate of decrease in WCSS slows down abruptly. This point is often referred to as the "elbow."
The number of clusters at the elbow point is considered to be the optimal number of clusters for the dataset.
In essence, the elbow method helps to find the point where increasing the number of clusters does not significantly improve the clustering performance, as adding more clusters does not substantially reduce the within-cluster variation. This optimal number of clusters strikes a balance between capturing meaningful patterns in the data while avoiding overfitting.
Use: I used the idea in 3a to determine the right k value by using elbow method

Generative AI use 2

The technology used was ChatGPT3.5
Question: in my data set, I have 1,2,3 .. to represent the age groups I wanna make a visualisation but first, I need to count the ages and assign them to their respective class. What is the best way?

```
# Define age group boundaries
age_group_boundaries <- c(17, 24, 34, 44, 54, 64, 74, 84, Inf)

# Assign age group based on age
dataset$age_group <- cut(dataset$age, breaks = age_group_boundaries, labels = FALSE)

# Count the number of individuals in each age group
age_counts <- table(dataset$age_group)
print(age_counts)
```

Used to create pie chart for age attribute distribution
Generative AI use 3

The technology used was ChatGPT3.5
Question:I cant include a file in my dataset i would rather have to have the whole table in my report
Answer: Tabulate the Data: Use R's kable function from the knitr package to create a nicely formatted table directly in your report. This function allows you to generate tables in various formats, including Markdown and HTML. Here's an example:
 # Install and load the knitr package if you haven't already
install.packages("knitr")
library(knitr)
kable(merged_data)
Used it to print the table and attach it as an appendix in my report for question 3a

Table used for 3a

	Country	Year	Control of Corruption Estimate	Government Effectiveness Estimate	Political Stability and Absence of Terrorism Estimate	Immunization	Vaccination_rates	Overall_Health_Score	EARLY_DETECTION/REPORTING.FOR.EPIDEMICS.OF.POTENTIAL.INT.L.CONCERN
1	Afghanistan	2019	-1.4194992	-1.5186142	-2.6524069	50	50	27.9	20.0
2	Afghanistan	2021	-1.1523274	-1.6695621	-2.5185304	50	50	28.8	20.6
3	Albania	2019	-0.5638996	-0.0621441	0.1100497	100	100	46.2	45.4
4	Albania	2021	-0.5765355	-0.0353601	0.1962940	100	100	45.0	40.0
5	Algeria	2019	-0.6558951	-0.5677304	-1.0558299	75	75	26.8	8.5
6	Algeria	2021	-0.6356576	-0.6532953	-0.9924372	50	50	26.2	12.6
9	Andorra	2019	1.1765922	1.8520651	1.5772115	75	75	30.5	2.2
10	Andorra	2021	1.2798718	1.7492533	1.5810674	100	100	34.7	2.2
11	Angola	2019	-1.0584179	-1.1285623	-0.3700947	50	50	25.2	13.3

12	Angola	2021	-0.6516102	-1.1280763	-0.7093695	50	50	29.1	13.3
19	Argentina	2019	-0.1069402	-0.1206776	-0.0978643	75	75	56.1	54.6
20	Argentina	2021	-0.4222729	-0.3896721	0.0004683	75	75	54.4	56.7
21	Armenia	2019	-0.2269752	-0.2259586	-0.4200442	100	100	63.2	67.9
22	Armenia	2021	0.0489783	-0.2819272	-0.8040366	100	100	61.8	69.6
25	Australia	2019	1.7881731	1.5387604	0.9173131	75	75	73.2	79.6
26	Australia	2021	1.7076461	1.4739963	0.8353159	75	75	71.1	82.2
27	Austria	2019	1.5214090	1.4922441	0.8920804	75	75	57.4	38.8
28	Austria	2021	1.2429926	1.5300202	0.8994187	75	75	56.9	41.4
29	Azerbaijan	2019	-0.8494244	-0.1281072	-0.6934121	100	100	34.2	21.7
30	Azerbaijan	2021	-0.8473721	0.2134166	-0.8378164	100	100	34.7	21.7
35	Bahrain	2019	-0.0380973	0.2452668	-0.6268352	100	100	38.9	33.5
36	Bahrain	2021	0.1445442	0.6833251	-0.5247797	100	100	36.3	37.2
37	Bangladesh	2019	-1.0169935	-0.7537483	-0.9286134	75	75	35.4	39.6
38	Bangladesh	2021	-0.9858609	-0.6556278	-1.0366328	100	100	35.5	43.8
39	Barbados	2019	1.1594402	0.5921367	1.0328128	50	50	32.2	7.9

40	Barbados	2021	1.2134006	0.4981979	1.1184167	50	50	34.9	13.8
41	Belarus	2019	0.0072857	-0.1743137	0.3449067	50	50	41.8	26.1
42	Belarus	2021	-0.2581499	-0.8045502	-0.7542452	50	50	43.9	34.4
43	Belgium	2019	1.4489959	1.1134673	0.4583162	75	75	61.9	52.9
44	Belgium	2021	1.4555250	1.0870334	0.6625663	75	75	59.3	52.9
45	Belize	2019	-0.1910709	-0.6341446	0.0619344	75	75	30.2	20.4
46	Belize	2021	-0.3300229	-0.5262241	0.2723819	100	100	29.7	20.4
47	Benin	2019	-0.3301082	-0.4905158	-0.4202595	50	50	27.0	18.3
48	Benin	2021	-0.1709435	-0.2376709	-0.3963478	50	50	25.4	14.2
51	Bhutan	2019	1.5722663	0.2516028	1.0653510	100	100	40.8	27.1
52	Bhutan	2021	1.5071211	0.7017297	0.7971277	75	75	39.8	33.3
53	Bolivia	2019	-0.7809188	-0.7904927	-0.7283386	50	50	32.0	21.3
54	Bolivia	2021	-0.8831868	-0.7623890	-0.2788130	50	50	29.9	21.3
59	Botswana	2019	0.6862887	0.3756971	1.0952075	50	50	31.1	18.9
60	Botswana	2021	0.6658261	0.3238047	1.0412596	50	50	33.6	29.3
61	Brazil	2019	-0.4076793	-0.2339713	-0.7143936	50	50	51.0	51.5

62	Brazil	2021	-0.4987977	-0.4922956	-0.4196965	50	50	51.2	53.6
67	Bulgaria	2019	-0.1807891	0.1738259	0.5611983	75	75	61.4	61.7
68	Bulgaria	2021	-0.2576372	-0.1729552	0.3646476	75	75	59.9	61.7
69	Burkina Faso	2019	-0.2201210	-0.7783582	-1.3021532	50	50	34.4	37.6
70	Burkina Faso	2021	-0.0830340	-0.7639615	-1.6428697	50	50	29.8	33.9
71	Burundi	2019	-1.4968066	-1.3747797	-1.6249934	50	50	22.7	14.2
72	Burundi	2021	-1.5993913	-1.3574017	-1.2583575	75	75	22.1	14.2
73	Cabo Verde	2019	0.8510368	0.2787158	0.8595210	75	75	32.6	10.6
74	Cabo Verde	2021	1.0169486	-0.0013689	0.8942331	75	75	34.1	14.7
75	Cambodia	2019	-1.3231657	-0.6236061	-0.0755314	50	50	31.0	32.9
76	Cambodia	2021	-1.1974894	-0.4781099	-0.1315732	75	75	31.1	37.1
77	Cameroon	2019	-1.2187133	-0.8661700	-1.5617648	50	50	32.2	31.4
78	Cameroon	2021	-1.1171824	-0.9043247	-1.3940357	0	0	28.6	30.8
79	Canada	2019	1.7299765	1.6973951	0.9948179	75	75	67.6	64.6
80	Canada	2021	1.6165298	1.5635246	0.9621960	75	75	69.8	70.8
83	Central African Republic	2019	-1.2339615	-1.7801981	-2.1357780	50	50	20.7	12.5

84	Central African Republic	2021	-1.2462837	-1.6720594	-2.1267397	50	50	18.6	12.5
85	Chad	2019	-1.4170589	-1.6070483	-1.3504070	50	50	24.5	18.3
86	Chad	2021	-1.4957544	-1.4607559	-1.3898128	50	50	23.9	18.3
87	Chile	2019	0.9791573	0.8635918	-0.0161684	75	75	53.0	43.5
88	Chile	2021	0.9567333	0.5922043	0.1613522	75	75	56.2	58.1
89	China	2019	-0.3124051	0.5449034	-0.2598109	50	50	49.0	48.5
90	China	2021	0.0303935	0.8093318	-0.5167289	50	50	47.5	48.5
91	Colombia	2019	-0.2831572	0.0327499	-0.9794673	75	75	50.0	43.3
92	Colombia	2021	-0.3654351	-0.0466582	-0.9545575	75	75	53.2	57.9
93	Comoros	2019	-1.0578653	-1.7532390	-0.1768335	50	50	25.2	15.8
94	Comoros	2021	-1.3096988	-1.8138077	-0.2280336	50	50	24.9	17.9
105	Costa Rica	2019	0.6272395	0.3477333	0.4302428	75	75	40.5	33.1
106	Costa Rica	2021	0.4704830	0.2241175	0.8602900	75	75	40.8	33.1
111	Croatia	2019	0.0527427	0.4569915	0.6854161	100	100	49.8	37.8
112	Croatia	2021	0.0379616	0.5530890	0.6214828	100	100	48.8	37.8
113	Cuba	2019	0.0146971	-0.1777930	0.6104111	100	100	32.3	6.8

114	Cuba	2021	-0.0332559	-0.2487406	0.3927019	100	100	30.5	13.1
115	Cyprus	2019	0.5992194	0.9559514	0.5445133	75	75	42.3	21.4
116	Cyprus	2021	0.3696348	0.6993143	0.4220542	75	75	41.9	25.0
121	Denmark	2019	2.1216173	1.8733810	0.9674850	75	75	67.3	60.4
122	Denmark	2021	2.3337526	1.9617968	0.9285245	75	75	64.4	64.6
123	Djibouti	2019	-0.8802125	-0.8179903	-0.3434125	75	75	23.9	10.0
124	Djibouti	2021	-0.8047198	-0.8317110	-0.5663399	75	75	25.2	14.2
125	Dominica	2019	0.5216411	-0.2531514	1.0298235	25	25	27.1	10.0
126	Dominica	2021	0.5474969	-0.1208519	1.3339591	25	25	26.4	14.2
127	Dominican Republic	2019	-0.8387812	-0.3512725	-0.0031599	50	50	35.8	27.9
128	Dominican Republic	2021	-0.5900345	-0.0019536	0.2873006	50	50	34.5	30.0
129	Ecuador	2019	-0.5090747	-0.3676400	-0.2382494	50	50	48.2	45.3
130	Ecuador	2021	-0.5940891	-0.2407347	-0.2574279	50	50	50.8	51.5
135	El Salvador	2019	-0.5376289	-0.5310541	-0.1202680	75	75	42.9	50.4
136	El Salvador	2021	-0.5530355	-0.3409142	-0.0915463	25	25	40.8	52.5
137	Equatorial Guinea	2019	-1.5633754	-1.0412003	-0.1479361	0	0	18.0	0.0

138	Equatorial Guinea	2021	-1.5868884	-1.1847091	-0.2027806	0	0	17.4	0.0
139	Eritrea	2019	-1.4070611	-1.8025423	-0.7059686	75	75	22.5	10.4
140	Eritrea	2021	-1.2571419	-1.6955115	-1.0556570	75	75	21.4	10.4
141	Estonia	2019	1.5254382	1.1397940	0.6301070	75	75	55.6	41.3
142	Estonia	2021	1.5068430	1.3450180	0.7484661	75	75	55.5	41.3
147	Ethiopia	2019	-0.4354776	-0.6615252	-1.3037219	50	50	37.4	23.5
148	Ethiopia	2021	-0.4229709	-0.6487360	-2.1837132	50	50	37.8	29.7
149	Fiji	2019	0.6880963	0.7624882	0.7797738	75	75	25.4	6.3
150	Fiji	2021	0.4432103	0.6543344	0.7241088	75	75	25.8	6.3
151	Finland	2019	2.1140051	1.9728713	0.8351541	75	75	72.0	65.4
152	Finland	2021	2.2382171	1.9206853	0.9631349	75	75	70.9	67.5
153	France	2019	1.2483281	1.3382878	0.2712749	75	75	62.6	45.1
154	France	2021	1.2822850	1.2306139	0.3255238	75	75	61.9	45.7
157	Gabon	2019	-0.8881114	-0.9575180	-0.0751414	0	0	19.9	3.3
158	Gabon	2021	-0.8695272	-0.8195783	-0.0757728	0	0	21.8	7.5
163	Georgia	2019	0.7163765	0.7662318	-0.4981859	75	75	48.2	51.5

164	Georgia	2021	0.6611901	0.6181614	-0.4296466	100	100	52.6	65.1
165	Germany	2019	1.8653662	1.4954659	0.5484546	75	75	65.7	70.3
166	Germany	2021	1.7831718	1.2908278	0.7256667	75	75	65.5	72.4
167	Ghana	2019	-0.1088939	-0.2900706	0.1184709	75	75	31.6	22.6
168	Ghana	2021	-0.1285271	-0.1763141	0.0657853	75	75	34.3	33.1
169	Greece	2019	0.0140720	0.3144917	0.1623694	75	75	50.6	48.9
170	Greece	2021	0.1834526	0.4067216	0.1013132	25	25	51.5	48.9
173	Grenada	2019	0.3379768	-0.1086683	0.9379679	0	0	25.6	5.8
174	Grenada	2021	0.4941749	0.0180175	1.0401636	25	25	26.7	10.0
177	Guatemala	2019	-0.9153916	-0.7264336	-0.5773494	75	75	31.0	30.8
178	Guatemala	2021	-1.1921371	-0.7823205	-0.3955471	50	50	29.1	30.8
179	Guinea	2019	-0.9159014	-0.8557539	-0.8425967	50	50	28.5	28.3
180	Guinea	2021	-1.0223854	-0.9539956	-0.9398943	50	50	26.8	28.3
181	Guinea-Bissau	2019	-1.4979489	-1.5601293	-0.5615771	50	50	19.3	12.5
182	Guinea-Bissau	2021	-1.3199301	-1.4479542	-0.2800246	50	50	21.4	16.7
183	Guyana	2019	-0.1428223	-0.3955183	-0.2541044	75	75	30.0	11.0

184	Guyana	2021	-0.1874441	-0.2613762	-0.1371637	75	75	30.8	11.0
185	Haiti	2019	-1.3468052	-2.0670664	-0.8876247	50	50	30.1	38.3
186	Haiti	2021	-1.4413589	-2.2187500	-1.1280994	50	50	30.4	38.3
187	Honduras	2019	-0.8756863	-0.6190965	-0.5690662	75	75	26.3	12.5
188	Honduras	2021	-1.0910184	-0.8181291	-0.6278241	75	75	26.2	12.5
191	Hungary	2019	0.0311543	0.4505960	0.7618216	100	100	55.0	38.1
192	Hungary	2021	0.0125177	0.5976323	0.7984638	100	100	54.4	38.1
193	Iceland	2019	1.6708744	1.4821239	1.6196480	100	100	47.6	32.2
194	Iceland	2021	1.7606179	1.5952553	1.3721787	100	100	48.5	36.4
195	India	2019	-0.3022053	0.1308447	-0.7968406	75	75	43.6	37.2
196	India	2021	-0.3164935	0.2508563	-0.6919979	75	75	42.8	43.5
197	Indonesia	2019	-0.4731803	0.1387610	-0.5021567	50	50	49.2	45.4
198	Indonesia	2021	-0.4490747	0.3473631	-0.5314672	50	50	50.4	55.4
203	Iraq	2019	-1.3898027	-1.3106563	-2.6091480	75	75	23.3	15.8
204	Iraq	2021	-1.2687253	-1.3196481	-2.3845594	75	75	24.0	24.2
205	Ireland	2019	1.4588335	1.2576550	0.9593033	50	50	55.1	49.9

206	Ireland	2021	1.6196781	1.4654449	0.8435314	50	50	55.3	50.4
207	Israel	2019	0.7741138	1.2887442	-0.8155604	100	100	50.7	43.3
208	Israel	2021	0.8296165	1.2526623	-1.1222825	100	100	47.2	46.7
209	Italy	2019	0.2342612	0.4510613	0.3810405	75	75	51.9	49.7
210	Italy	2021	0.5174145	0.3260358	0.5504139	75	75	51.9	49.7
211	Jamaica	2019	-0.1138340	0.5890101	0.3882355	100	100	30.9	18.8
212	Jamaica	2021	-0.0529802	0.3783157	0.2179147	75	75	31.8	19.3
213	Japan	2019	1.4311672	1.5501909	1.0196950	100	100	58.8	56.1
214	Japan	2021	1.5360502	1.3627553	1.0153564	75	75	60.5	71.1
217	Jordan	2019	0.0804249	0.0748955	-0.2744396	100	100	41.2	27.2
218	Jordan	2021	0.0268350	0.1955090	-0.3157934	100	100	42.8	32.5
219	Kazakhstan	2019	-0.2856953	0.0689593	-0.1679799	100	100	44.7	22.4
220	Kazakhstan	2021	-0.2591599	0.0303978	-0.2313275	100	100	46.1	29.2
221	Kenya	2019	-0.8074189	-0.4418037	-1.1057360	50	50	43.1	51.5
222	Kenya	2021	-0.7358834	-0.3247459	-1.0324142	50	50	38.8	55.7
223	Kiribati	2019	0.3671322	-0.0515527	1.1242330	0	0	21.6	0.6

224	Kiribati	2021	0.2817656	0.0944074	1.1389109	25	25	26.2	4.7
231	Kuwait	2019	-0.1594345	-0.0172721	0.1751136	100	100	40.1	17.9
232	Kuwait	2021	-0.0562744	-0.0731540	0.2629825	75	75	36.8	17.9
233	Kyrgyz Republic	2019	-0.9598920	-0.7316424	-0.2661922	100	100	43.0	26.7
234	Kyrgyz Republic	2021	-1.1497189	-0.7765979	-0.4671561	100	100	42.4	26.7
239	Latvia	2019	0.4808072	1.0658338	0.4267935	75	75	59.8	72.9
240	Latvia	2021	0.7208624	0.8341649	0.6733230	100	100	61.9	77.1
241	Lebanon	2019	-1.1687958	-0.8090068	-1.6722910	50	50	36.8	41.0
242	Lebanon	2021	-1.2480382	-1.3116974	-1.4819446	0	0	33.4	38.9
243	Lesotho	2019	-0.1236888	-0.9026215	-0.4279990	75	75	32.6	8.5
244	Lesotho	2021	-0.3466125	-0.9483575	-0.2132691	25	25	30.9	8.5
245	Liberia	2019	-0.9190943	-1.3950703	-0.3391354	0	0	34.5	23.8
246	Liberia	2021	-0.9392408	-1.4103801	-0.2863615	0	0	35.7	24.6
247	Libya	2019	-1.5750152	-1.7741348	-2.5657713	75	75	23.3	22.1
248	Libya	2021	-1.5844948	-1.7474649	-2.3083458	50	50	25.3	28.3
249	Liechtenstein	2019	1.9210759	1.6505092	1.5974932	50	50	45.0	17.1

250	Liechtenstein	2021	1.6653777	1.4651920	1.5954915	50	50	46.4	17.1
251	Lithuania	2019	0.6666125	1.0077667	0.7705232	75	75	54.9	62.2
252	Lithuania	2021	0.8248296	1.0187372	0.8097184	75	75	59.5	64.3
253	Luxembourg	2019	2.0725117	1.6956311	1.3331749	25	25	48.6	33.3
254	Luxembourg	2021	1.8411816	1.6790369	1.1939195	25	25	48.4	33.3
257	Madagascar	2019	-1.0522057	-1.1932924	-0.3139838	50	50	30.9	27.5
258	Madagascar	2021	-0.9524589	-1.0277181	-0.5422798	50	50	30.4	31.7
259	Malawi	2019	-0.7854043	-0.7987269	-0.2868479	50	50	27.8	14.7
260	Malawi	2021	-0.3231938	-0.7928368	-0.1093158	50	50	28.5	10.6
261	Malaysia	2019	0.2333076	0.9678217	0.1460664	100	100	55.1	57.5
262	Malaysia	2021	0.1467863	0.9535441	0.0587540	75	75	56.4	72.5
263	Maldives	2019	-0.2632436	-0.1957321	0.0276085	50	50	30.8	16.7
264	Maldives	2021	-0.3799698	0.3241976	0.5347784	50	50	32.0	20.8
265	Mali	2019	-0.7052555	-1.0992519	-2.2142057	50	50	30.6	24.6
266	Mali	2021	-0.8880186	-1.2519928	-2.3328054	50	50	29.0	25.1
267	Malta	2019	0.2100382	0.8243102	1.0135281	75	75	39.3	19.7

268	Malta	2021	0.2930495	0.8533724	0.8873797	100	100	40.2	21.8
269	Marshall Islands	2019	0.3671322	-0.1663450	1.1751025	0	0	18.8	1.7
270	Marshall Islands	2021	0.4167231	0.0241439	0.9339380	0	0	24.6	20.4
273	Mauritania	2019	-0.8657974	-0.6093966	-0.5629307	0	0	25.4	24.6
274	Mauritania	2021	-0.8418127	-0.7751758	-0.5555299	0	0	26.2	26.7
275	Mauritius	2019	0.2098069	0.8623255	0.8019058	100	100	38.3	35.8
276	Mauritius	2021	0.4424288	0.8109112	0.8324654	100	100	39.7	32.2
277	Mexico	2019	-0.9171847	-0.3066530	-0.8436288	50	50	55.1	50.1
278	Mexico	2021	-1.0203100	-0.3422988	-0.6883609	50	50	57.0	54.3
283	Moldova	2019	-0.6444892	-0.4406213	-0.3873099	75	75	40.8	34.2
284	Moldova	2021	-0.4689192	-0.4378827	-0.2053421	100	100	41.0	34.2
285	Monaco	2019	1.7807099	1.8520651	1.5772115	0	0	33.8	20.6
286	Monaco	2021	1.2798718	2.0267587	1.1724969	0	0	33.3	20.6
287	Mongolia	2019	-0.4553134	-0.2345575	0.6366597	100	100	40.9	37.9
288	Mongolia	2021	-0.5529425	-0.5164195	0.7057989	100	100	41.0	37.9
289	Montenegro	2019	-0.0298483	0.1066157	0.0580168	75	75	40.8	17.5

290	Montenegro	2021	-0.0434206	-0.0299340	-0.0525021	75	75	44.1	32.1
291	Morocco	2019	-0.3356444	-0.2515638	-0.3458297	100	100	35.6	27.9
292	Morocco	2021	-0.4209625	-0.1734744	-0.4025913	100	100	33.6	27.9
293	Mozambique	2019	-0.8589432	-0.8883838	-0.7667162	50	50	29.6	24.2
294	Mozambique	2021	-0.8189727	-0.7543381	-1.2593542	75	75	30.4	28.3
295	Myanmar	2019	-0.6452308	-1.1853216	-1.3304590	75	75	37.8	38.5
296	Myanmar	2021	-1.0480769	-1.4123977	-2.0839694	75	75	38.3	46.8
297	Namibia	2019	0.3275282	0.0947278	0.5288066	50	50	30.9	35.4
298	Namibia	2021	0.2360043	0.0270601	0.5186849	50	50	30.3	31.8
299	Nauru	2019	0.5295177	-0.0037915	0.7927320	25	25	19.5	0.0
300	Nauru	2021	0.6041547	0.0982897	0.7871121	50	50	18.0	0.0
301	Nepal	2019	-0.6910474	-1.0925521	-0.4529429	50	50	35.6	23.9
302	Nepal	2021	-0.5561397	-0.9312484	-0.1888078	50	50	34.0	28.1
303	Netherlands	2019	1.8678021	1.7672653	0.8212212	75	75	67.7	61.3
304	Netherlands	2021	2.0046263	1.7270592	0.8929240	75	75	64.7	57.1
307	New Zealand	2019	2.1310298	1.6326025	1.4157494	75	75	55.8	47.6

308	New Zealand	2021	2.1701555	1.3068949	1.3952363	75	75	62.5	75.3
309	Nicaragua	2019	-1.0320594	-0.7773451	-0.9924811	75	75	40.0	30.8
310	Nicaragua	2021	-1.2547038	-0.8843739	-0.4719982	100	100	36.3	23.3
311	Niger	2019	-0.5701931	-0.8404199	-1.4049586	50	50	29.7	28.3
312	Niger	2021	-0.5826251	-0.6438968	-1.5384851	50	50	28.7	24.2
313	Nigeria	2019	-1.1151304	-1.2133290	-1.9330711	50	50	37.0	35.8
314	Nigeria	2021	-1.1032567	-1.0278199	-1.7872993	50	50	38.0	37.9
319	North Macedonia	2019	-0.4566225	-0.1238972	0.0080933	100	100	40.1	30.8
320	North Macedonia	2021	-0.3746143	-0.1211693	0.0624172	75	75	42.2	37.1
321	Norway	2019	2.0338359	1.8277836	1.1419865	75	75	61.4	52.5
322	Norway	2021	2.1087081	1.7979517	1.0884751	100	100	60.2	46.3
323	Oman	2019	0.4215426	0.2064655	0.5891687	100	100	40.9	33.5
324	Oman	2021	0.0621689	-0.1551568	0.4318880	100	100	39.1	33.5
325	Pakistan	2019	-0.8793301	-0.7181478	-2.2516375	50	50	31.3	25.0
326	Pakistan	2021	-0.8097374	-0.4383641	-1.7136562	50	50	30.4	29.2
327	Palau	2019	0.5295177	0.2279621	1.1716704	0	0	19.9	1.7

328	Palau	2021	0.6041547	0.3708985	1.1722332	25	25	25.5	17.5
329	Panama	2019	-0.6435570	0.0999357	0.2908781	75	75	50.4	51.7
330	Panama	2021	-0.5912358	0.1242338	0.3071201	100	100	53.5	50.4
331	Papua New Guinea	2019	-1.0023326	-0.7924939	-0.7050464	50	50	26.3	18.8
332	Papua New Guinea	2021	-0.7702186	-0.8861411	-0.5767089	50	50	25.0	14.6
333	Paraguay	2019	-0.8934145	-0.5767058	-0.0180594	75	75	39.8	30.4
334	Paraguay	2021	-1.0259774	-0.6554369	-0.0009996	75	75	40.3	28.3
335	Peru	2019	-0.5092068	-0.1257993	-0.1636593	50	50	53.8	48.9
336	Peru	2021	-0.6532260	-0.2944655	-0.3552855	50	50	54.9	57.8
337	Philippines	2019	-0.6059991	0.0599960	-0.9280454	50	50	43.5	34.3
338	Philippines	2021	-0.5295781	0.0356625	-0.9840031	50	50	45.7	52.6
339	Poland	2019	0.6109892	0.5129446	0.5506312	75	75	54.3	31.0
340	Poland	2021	0.5464258	0.2548574	0.4918013	75	75	55.7	42.5
341	Portugal	2019	0.7457133	1.1342911	1.0516324	100	100	58.7	44.7
342	Portugal	2021	0.7423466	0.9536869	0.9372513	100	100	54.7	42.6
345	Qatar	2019	0.8172509	0.7074524	0.6842756	75	75	45.1	33.5

346	Qatar	2021	0.7799670	1.0764580	0.8332614	100	100	48.7	39.7
349	Romania	2019	-0.2365268	-0.2150140	0.5421712	50	50	45.5	33.6
350	Romania	2021	-0.0609266	-0.1632058	0.5816803	50	50	45.7	44.0
355	Rwanda	2019	0.5210270	0.0929243	0.0569405	100	100	31.1	24.6
356	Rwanda	2021	0.5732059	0.2279074	0.0719058	75	75	33.1	34.6
357	Samoa	2019	0.6480354	0.4362015	1.1637257	50	50	29.7	0.0
358	Samoa	2021	0.5925877	0.3771847	1.1062322	50	50	28.8	4.2
359	San Marino	2019	1.1765922	1.5796709	1.2258400	50	50	32.2	17.2
360	San Marino	2021	1.2798718	1.7492533	1.1724969	50	50	32.9	21.4
365	Saudi Arabia	2019	0.2464770	0.2613156	-0.6234558	100	100	45.0	50.0
366	Saudi Arabia	2021	0.2827371	0.4658907	-0.5911313	100	100	44.9	52.1
367	Senegal	2019	-0.0038142	-0.1343478	0.0375260	50	50	35.9	28.3
368	Senegal	2021	0.0421113	0.0218494	-0.1798180	50	50	32.8	28.3
369	Serbia	2019	-0.4499893	-0.0166273	-0.0780941	75	75	45.0	28.6
370	Serbia	2021	-0.4585689	0.0126692	-0.0852662	75	75	45.0	28.6
371	Seychelles	2019	1.1759818	0.6225972	0.6586941	50	50	33.2	22.9

372	Seychelles	2021	1.6004549	0.8995878	0.7446139	50	50	31.8	18.8
373	Sierra Leone	2019	-0.4716616	-1.1745967	-0.0569588	50	50	34.1	31.4
374	Sierra Leone	2021	-0.4563256	-1.1506950	-0.0970259	0	0	32.7	31.4
375	Singapore	2019	2.1201060	2.2317193	1.4803056	75	75	55.8	49.0
376	Singapore	2021	2.1398079	2.2501128	1.4423250	75	75	57.4	61.1
381	Slovenia	2019	0.8889422	1.0419170	0.8014835	75	75	68.6	66.7
382	Slovenia	2021	0.6934186	1.1386214	0.7554614	75	75	67.8	70.8
383	Solomon Islands	2019	-0.0503519	-0.8811658	0.4956767	0	0	21.8	4.2
384	Solomon Islands	2021	-0.1635806	-0.8337227	0.5416700	0	0	23.3	4.2
385	Somalia	2019	-1.7237843	-2.2808883	-2.3862884	50	50	17.9	15.8
386	Somalia	2021	-1.7951125	-2.0838933	-2.7271757	50	50	16.0	11.7
387	South Africa	2019	-0.0310517	0.1278810	-0.2832497	50	50	47.5	52.1
388	South Africa	2021	-0.0341142	-0.0710127	-0.7509814	50	50	45.8	50.0
391	South Sudan	2019	-1.7998860	-2.3836708	-2.5183439	50	50	21.6	16.7
392	South Sudan	2021	-1.8368162	-2.3965635	-2.2872014	50	50	21.3	14.6
393	Spain	2019	0.6665267	0.9688812	0.2927009	75	75	60.4	64.6

394	Spain	2021	0.7155032	0.9092734	0.5091319	75	75	60.9	70.8
395	Sri Lanka	2019	-0.3038257	-0.1248893	-0.2155698	100	100	33.1	32.9
396	Sri Lanka	2021	-0.3563530	-0.1144987	-0.3768038	100	100	34.1	35.6
409	Sudan	2019	-1.4078447	-1.6586438	-1.6961982	50	50	30.0	15.8
410	Sudan	2021	-1.2792203	-1.6632018	-1.9875722	50	50	28.3	15.8
411	Suriname	2019	-0.4198544	-0.6268236	0.0876267	50	50	33.2	20.0
412	Suriname	2021	-0.4186403	-0.6806551	0.3483017	50	50	35.0	24.2
413	Sweden	2019	2.0920341	1.6740988	1.0117697	100	100	66.4	64.6
414	Sweden	2021	2.0988910	1.6124706	1.0143628	100	100	64.9	62.5
415	Switzerland	2019	1.9450564	1.9163042	1.3108382	75	75	60.4	38.3
416	Switzerland	2021	1.9590089	1.9930376	1.1152221	75	75	58.8	42.5
423	Tajikistan	2019	-1.3455535	-1.1167146	-0.5120729	100	100	29.8	5.8
424	Tajikistan	2021	-1.3540801	-0.6469495	-0.6768180	50	50	29.3	10.6
425	Tanzania	2019	-0.4230077	-0.8546779	-0.3884414	50	50	32.2	29.7
426	Tanzania	2021	-0.3969004	-0.6463974	-0.3355928	50	50	31.3	25.6
427	Thailand	2019	-0.4717595	0.2606455	-0.4914205	100	100	68.9	83.2

428	Thailand	2021	-0.4783483	0.2211909	-0.5687262	75	75	68.2	91.5
429	Timor-Leste	2019	-0.4005702	-0.8581024	0.2430168	0	0	24.2	18.3
430	Timor-Leste	2021	-0.0724385	-0.7549486	0.1923154	25	25	27.8	24.6
431	Togo	2019	-0.7472892	-0.9727898	-0.9082437	50	50	26.1	27.1
432	Togo	2021	-0.6916795	-0.6865243	-0.6957114	50	50	27.8	34.6
433	Tonga	2019	-0.3298207	0.2056895	1.0377861	100	100	24.5	4.2
434	Tonga	2021	-0.4286540	0.3034193	1.0789658	100	100	26.4	8.3
435	Trinidad and Tobago	2019	-0.2567028	0.0881585	0.0815371	75	75	37.7	12.1
436	Trinidad and Tobago	2021	-0.2999710	0.1549869	0.2390293	75	75	36.8	12.6
437	Tunisia	2019	-0.1408772	-0.0353329	-0.8770729	75	75	32.1	20.4
438	Tunisia	2021	-0.2636990	-0.2100148	-0.7565514	75	75	31.5	20.4
443	Turkmenistan	2019	-1.3935708	-1.0176277	-0.1781499	100	100	33.3	27.1
444	Turkmenistan	2021	-1.4390671	-0.9620621	-0.3320110	100	100	31.9	27.6
445	Tuvalu	2019	0.3868456	-0.5368156	1.1716704	25	25	20.2	0.0
446	Tuvalu	2021	0.6609164	-0.4026066	1.2676833	25	25	20.0	0.0
447	Uganda	2019	-1.1765792	-0.6487716	-0.6942645	50	50	39.0	35.0

448	Uganda	2021	-1.0305375	-0.6001834	-0.9377432	50	50	36.5	35.6
449	Ukraine	2019	-0.8002426	-0.3341684	-1.4283267	75	75	36.9	23.3
450	Ukraine	2021	-0.7864681	-0.4388590	-1.1269530	75	75	38.9	32.8
451	United Arab Emirates	2019	1.0721111	1.3803160	0.6674451	100	100	40.1	25.1
452	United Arab Emirates	2021	1.1501433	1.3659306	0.5974612	100	100	39.6	22.6
453	United Kingdom	2019	1.7487724	1.4499079	0.5274704	75	75	68.3	62.5
454	United Kingdom	2021	1.6409292	1.2417737	0.4898921	75	75	67.2	70.8
459	Uruguay	2019	1.1954031	0.6431634	1.0281792	75	75	39.1	15.0
460	Uruguay	2021	1.5862535	0.7999353	1.0458239	100	100	40.3	15.0
461	Uzbekistan	2019	-1.0468915	-0.6000623	-0.2966655	100	100	37.7	20.0
462	Uzbekistan	2021	-0.8293310	-0.2661737	-0.2529742	100	100	39.0	18.5
463	Vanuatu	2019	-0.2841584	-0.5175012	1.0021076	50	50	27.0	4.2
464	Vanuatu	2021	-0.0398655	-0.5504693	0.9003791	0	0	25.9	6.8
469	Vietnam	2019	-0.5458180	0.0279832	0.0389102	75	75	42.2	42.1
470	Vietnam	2021	-0.3078603	0.2433947	-0.1181897	75	75	42.9	55.1
479	Zambia	2019	-0.6739175	-0.7419474	-0.1170296	50	50	28.0	18.8

480	Zambia	2021	-0.7597278	-0.8411018	0.0523477	0	0	26.5	19.3
481	Zimbabwe	2019	-1.2711903	-1.3197736	-0.9432861	50	50	33.4	40.4
482	Zimbabwe	2021	-1.2535501	-1.3048168	-0.9544259	50	50	32.4	40.4

R Code

```
rm(list = ls())
set.seed(33370311) # XXXXXXXX = your student ID
cvbase =
read.csv("C:/Users/Home/OneDrive/Desktop/3152/PsyCoronaBaselineExtract.csv")
cvbase <- cvbase[sample(nrow(cvbase), 40000), ] # 40000 rows
# Question 1
# 1(a)
# to check if the csv is loaded correctly
head(cvbase)
# dimensions of the data
dim(cvbase)
# names of all the columns in my data
names(cvbase)
#Examine the structure of the data
str(cvbase)
#using the summary function to check the summary of csv
summary(cvbase)

#checking how many countries exist in the csv
unique_countries <- unique(cvbase$coded_country)
print(length(unique_countries))

#checking all the numeric and non numeric attributes in our data
numerical_attributes <- sapply(cvbase, is.numeric)
numerical_attributes_names <- names(numerical_attributes[numerical_attributes == TRUE])
non_numerical_names <- names(numerical_attributes[numerical_attributes == FALSE])
cat("The Numeric Attributes:", numerical_attributes_names, "\n")
cat("The Non-Numeric Attributes:", non_numerical_names, "\n")
#check for all NA or missing values
na_count <- sapply(cvbase, function(x) sum(is.na(x)))
na_count[na_count > 0]

#the function which replace NA with zero
```

```
replace_na_with_zero <- function(data, columns) {
  for (col in columns) {
    data[[col]] <- ifelse(is.na(data[[col]]), 0, data[[col]])
  }
  return(data)
}

# Columns to process
emp_status_columns <- c("employstatus_1", "employstatus_2", "employstatus_3",
"employstatus_4", "employstatus_5", "employstatus_6", "employstatus_7",
"employstatus_8", "employstatus_9", "employstatus_10")
corona_close_columns <-
c("coronaClose_1","coronaClose_2","coronaClose_3","coronaClose_4","coronaClose_5","co
ronaClose_6")
# Apply the function to replace NA values with 0 for emp_status_columns
cvbase <- replace_na_with_zero(cvbase, emp_status_columns)

cvbase <- replace_na_with_zero(cvbase, corona_close_columns)

# Below function is used to plot and visualize the distribution of all the Employ Status
columns
# Set up the layout for the plots
par(mfrow=c(2, 3)) # 2 rows, 3 columns


# Function to replace NA values with 0 and generate bar plot for each employ status column
replace_na_and_plot <- function(column_name) {
  cvbase[[column_name]] <- ifelse(is.na(cvbase[[column_name]]), 0,
cvbase[[column_name]])
  # Update the column in the original dataframe
  cvbase[[column_name]] <- cvbase[[column_name]]

  counts <- table(cvbase[[column_name]])
  ordered_levels <- names(sort(counts))

  barplot(counts[ordered_levels], main = paste("Distribution of", column_name),
          xlab = "Ordinal Category", ylab = "Frequency", col = "skyblue")
}

# Apply the function to each column
for (col in emp_status_columns) {
  replace_na_and_plot(col)
}
par(mfrow=c(2, 3)) # 2 rows, 3 columns


# Apply the function to each column
for (col in corona_close_columns ) {
```

```
  replace_na_and_plot(col)
}

#The list of plots created by the below function is a more gender based distribution of
Employ Status'

# Set up the layout for the plots
par(mfrow=c(4, 3)) # 2 rows, 3 columns

# Function to generate stacked bar plot for each employ status column
generate_stacked_bar_plot <- function(column_name) {
  # Filter out NA values
  cvbase[[column_name]] <- ifelse(is.na(cvbase[[column_name]]), 0,
cvbase[[column_name]])

  # Create a table to count gender occurrences within each employ status
  counts <- table(cvbase$gender, cvbase[[column_name]])

  # Create stacked bar plot
  barplot(counts, beside = TRUE,
          legend.text = FALSE, col = c("yellow", "lightblue", "lightgreen"),
          xlab = column_name, ylab = "Count")

  # Add legend for gender only
  legend("topright", legend = c("Female", "Male", "Other"), fill = c("yellow", "lightblue",
"lightgreen"))
}

# Apply the function to each employ status column
for (column in emp_status_columns) {
  generate_stacked_bar_plot(column)
}
# Age Distribution
age_counts <- table(cvbase$age)

# Define the age groups
age_groups <- c("18-24", "25-34", "35-44", "45-54", "55-64", "65-74", "75-84", "85+")

# Calculate percentages
age_percentages <- round(prop.table(age_counts) * 100, 1)

# Create a pie chart
pie(age_counts,
    main = "Pie Chart of Age Distribution",
    col = rainbow(length(age_counts)),
    labels = paste("Age Group ", age_groups, "\n", age_percentages, "%"),
    cex = 0.8)
```

```
# Legend indicating the age groups
legend("bottomleft", legend = age_groups, fill = rainbow(length(age_counts)))


par(mfrow=c(1, 3))

create_custom_histogram <- function(data, column, breaks, labels) {
  # Remove NA values from the specified column
  cleaned_data <- na.omit(data[[column]])

  hist(cleaned_data,
       breaks = breaks,
       main = paste("Histogram of", column, "Distribution"),
       xlab = "Scale",
       ylab = "Frequency",
       col = "skyblue",
       labels = labels,
       cex.lab = 1)


}

# Define breaks and labels for the histogram bins
breaks <- seq(-3.5, 3.5, by = 1)
labels <- c("Strongly disagree", "Disagree", "Somewhat disagree", "Neither agree nor
disagree", "Somewhat agree", "Agree", "Strongly agree")

# Specify the columns for which you want to create histograms
columns <- c("bor01", "bor02", "bor03")
# Loop through each column and create a customized histogram
for (col in columns) {
  create_custom_histogram(cvbase, col, breaks, labels)
}


par(mfrow=c(3, 1))

columns <- c("c19RCA01", "c19RCA02", "c19RCA03")
# Loop through each column and create a customized histogram
for (col in columns) {
  create_custom_histogram(cvbase, col, breaks, labels)
}



columns <- c( "c19perBeh01", "c19perBeh02", "c19perBeh03")
# Loop through each column and create a customized histogram
for (col in columns) {
```

```
  create_custom_histogram(cvbase, col, breaks, labels)
}



par(mfrow=c(1, 1))

columns <- c( "MLQ")
# Loop through each column and create a customized histogram
for (col in columns) {
  create_custom_histogram(cvbase, col, breaks, labels)
}


par(mfrow=c(2, 2))
columns <- c( "c19ProSo01", "c19ProSo02", "c19ProSo03", "c19ProSo04")
# Loop through each column and create a customized histogram
for (col in columns) {
  create_custom_histogram(cvbase, col, breaks, labels)
}


#Life Satisfaction Distribution
par(mfrow=c(1, 1))
# Now we just change bin breaks in our histograms
breaks <- seq(0.5, 7.5, by = 1)
# Now we also change labels
labels <- c("Very dissatisfied", "Dissatisfied", "Slightly
          dissatisfied","Slightly satisfied","Satisfied", "Very
          satisfied")

columns <-c("lifeSat")
# Loop through each column and create a customized histogram
for (col in columns) {
  create_custom_histogram(cvbase, col, breaks, labels)
}


# Loneliness Distribution
par(mfrow=c(1, 3))
# Now we just change bin breaks in our histograms
breaks <- seq(0.5, 5.5, by = 1)
# Now we also change labels
labels <- c("Never", "Rarely", "Sometimes", "Often" , "All the time")

columns <-c("lone01","lone02","lone03")
# Loop through each column and create a customized histogram
for (col in columns) {
```

```
  create_custom_histogram(cvbase, col, breaks, labels)
}


# Loneliness Distribution
par(mfrow=c(2, 2))
# Now we just change bin breaks in our histograms
breaks <- seq(-0.5, 7.5, by = 1)
# Now we also change labels
labels <- c("0","1","2","3","4","5","6","7")

columns
<-c("isoFriends_inPerson","isoOthPpl_inPerson","isoFriends_online","isoOthPpl_online")
# Loop through each column and create a customized histogram
for (col in columns) {
  create_custom_histogram(cvbase, col, breaks, labels)
}

#-----------------------------------------------------------------------------------------------------------------------
-------------------------------------------------
#Question 2

# 2(a)
# Load necessary libraries
library(dplyr)
# Impute missing values with mean for numeric columns
imputed_cvbase <- cvbase %>%
  mutate_all(~ifelse(is.na(.), mean(., na.rm = TRUE), .))
# then for non numeric variables we remove na
cleaned_cvbase <- imputed_cvbase %>%
  na.omit()
#below we are just using contrast on rankOrderLife variables
cleaned_cvbase$rankOrdLife_1 <- factor(cleaned_cvbase$rankOrdLife_1)
cleaned_cvbase$rankOrdLife_2 <- factor(cleaned_cvbase$rankOrdLife_2)
cleaned_cvbase$rankOrdLife_3 <- factor(cleaned_cvbase$rankOrdLife_3)
cleaned_cvbase$rankOrdLife_4 <- factor(cleaned_cvbase$rankOrdLife_4)
cleaned_cvbase$rankOrdLife_5 <- factor(cleaned_cvbase$rankOrdLife_5)
cleaned_cvbase$rankOrdLife_6 <- factor(cleaned_cvbase$rankOrdLife_6)
contrasts(cleaned_cvbase$rankOrdLife_1) = contr.treatment(6)
contrasts(cleaned_cvbase$rankOrdLife_2) = contr.treatment(6)
contrasts(cleaned_cvbase$rankOrdLife_3) = contr.treatment(6)
contrasts(cleaned_cvbase$rankOrdLife_4) = contr.treatment(6)
contrasts(cleaned_cvbase$rankOrdLife_5) = contr.treatment(6)
contrasts(cleaned_cvbase$rankOrdLife_6) = contr.treatment(6)

# Subset Data for Pakistan
pakistan_data <- cleaned_cvbase[cleaned_cvbase$coded_country == "Pakistan",]
# Subset Data for Other Countries and Remove NA values
```

```
other_countries_data <- cleaned_cvbase[cleaned_cvbase$coded_country != "Pakistan", ]
#just to know how much rows i still have left
dim(pakistan_data)
# Print unique country codes (just for verification)
print(pakistan_data$coded_country)

# Filter numeric columns
numeric_attributes_pakistan<- pakistan_data[sapply(pakistan_data, is.numeric)]
numeric_attributes_other<- other_countries_data[sapply(other_countries_data, is.numeric)]

# Summarize numeric attributes
summary_stats_pakistan <- summarise_all(numeric_attributes_pakistan, list(mean = mean,
median = median, sd = sd))
#  Aggregate Data for Other Countries
summary_stats_other <- summarise_all(numeric_attributes_other, list(mean = mean, median
= median, sd=sd))
print(summary_stats_pakistan)
print(summary_stats_other)
text_columns <- names(cvbase)

find_max <- function(column) {
  # Calculate frequency of each unique value in the column for both Pakistan and other
countries
  freq_pakistan <- table(pakistan_data[[column]])
  freq_other <- table(other_countries_data[[column]])

  # Find the maximum frequency for Pakistan and other countries
  max_freq_pakistan <- max(freq_pakistan)
  max_freq_other <- max(freq_other)

  # Identify the element(s) with maximum frequency
  max_elem_pakistan <- names(freq_pakistan)[freq_pakistan == max_freq_pakistan]
  max_elem_other <- names(freq_other)[freq_other == max_freq_other]

  # Print the results
  cat("Max occurrence for other countries in", column, ":", max_elem_other, "with frequency",
max_freq_other, "\n")
  cat("Max occurrence for Pakistan in", column, ":", max_elem_pakistan, "with frequency",
max_freq_pakistan, "\n")
}
for (col in text_columns){
  find_max(col)
}


# Subset Data for Pakistan
pakistan_data <- cleaned_cvbase[cleaned_cvbase$coded_country == "Pakistan", ]
```

```
# Subset Data for Other Countries
other_countries_data <- cleaned_cvbase[cleaned_cvbase$coded_country != "Pakistan", ]

# Select relevant columns
columns_of_interest <- c("c19ProSo01", "c19ProSo02", "c19ProSo03", "c19ProSo04")

# Combine Pakistan and Other Countries data
combined_data <- rbind(pakistan_data, other_countries_data)

# Add a column indicating the country
combined_data$country <- factor(ifelse(combined_data$coded_country == "Pakistan",
"Pakistan", "Other Countries"))

install.packages("corrplot")

# Load the corrplot package
library(corrplot)
# Now lets check the correlation matrix for each pro social behaviours for both the datasets

# Select relevant columns
columns_of_interest <- c("c19ProSo01", "c19ProSo02", "c19ProSo03", "c19ProSo04")

# Calculate the correlation matrix
correlation_matrix_pakistan <- cor(pakistan_data[columns_of_interest])
correlation_matrix_other <- cor(other_countries_data[columns_of_interest])

# Create the correlation plot with title
corrplot(correlation_matrix_pakistan, method = "color", type = "upper",
        addCoef.col = "black", tl.col = "black", tl.srt = 45,
        main = "Correlation Plot for Pakistan Data")

# Create the correlation plot
corrplot(correlation_matrix_other, method = "color", type = "upper",
        addCoef.col = "black", tl.col = "black", tl.srt = 45,main = "Correlation Plot for Other
Countries Data")


# Age Distribution
age_counts_pakistan <- table(pakistan_data$age)
# Age Distribution
age_counts_other <- table(other_countries_data$age)

# Define the age groups
age_groups <- c("18-24", "25-34", "35-44", "45-54", "55-64", "65-74", "75-84", "85+")

# Calculate percentages
age_percentages_other <- round(prop.table(age_counts_other) * 100, 1)
age_percentages_pakistan <- round(prop.table(age_counts_pakistan) * 100, 1)
```

```
# Create a pie chart
pie(age_counts_pakistan,
    main = "Pie Chart of Age Distribution Of Pakistan",
    col = rainbow(length(age_counts)),
    labels = paste("Age Group ", age_groups, "\n", age_percentages_pakistan, "%"),
    cex = 0.8)

# Legend indicating the age groups
legend("bottomleft", legend = age_groups, fill = rainbow(length(age_counts_pakistan)))


# Create a pie chart
pie(age_counts_other,
    main = "Pie Chart of Age Distribution Of Other Countries",
    col = rainbow(length(age_counts)),
    labels = paste("Age Group ", age_groups, "\n", age_percentages_other, "%"),
    cex = 0.8)

# Legend indicating the age groups
legend("bottomleft", legend = age_groups, fill = rainbow(length(age_counts_other)))

# Load necessary libraries
library(ggplot2)
library(dplyr)
install.packages("tidyr")

# Load the tidyr package
library(tidyr)

# Assuming you have two datasets: pakistan_data and other_countries_data

# Combine both datasets
combined_data <- bind_rows(
  mutate(pakistan_data, Country = "Pakistan"),
  mutate(other_countries_data, Country = "Other Countries")
)

# Reshape the data for plotting
combined_data_long <- combined_data %>%
  select(Country, starts_with("employstatus_")) %>%
  pivot_longer(cols = starts_with("employstatus_"), names_to = "Employment_Status",
values_to = "Count")

# Create a multivariate bar plot
ggplot(combined_data_long, aes(x = Employment_Status, y = Count, fill = Country)) +
  geom_bar(stat = "identity", position = "stack") +
```

```
  labs(title = "Employment Status Comparison between Pakistan and Other Countries", x =
"Employment Status", y = "Count") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotate x-axis labels if needed


# Load necessary library
library(plotly)

# Create 3D scatter plot
plot_ly(pakistan_data, x = ~age, y = ~lone01, z = ~`lifeSat`,
        color = ~`lifeSat`,
        colors = c("#440154", "#3B528B", "#21918C", "#5DC863", "#FDE725"),
        marker = list(size = 5),
        type = "scatter3d", mode = "markers",
        text = paste("Age: ", pakistan_data$age, "<br>",
                     "Loneliness: ", pakistan_data$lone01, "<br>",
                     "Life Satisfaction: ", pakistan_data$`lifeSat`),
        hoverinfo = "text") %>%
  layout(title = "Relationship between Age, Loneliness, and Life Satisfaction of Pakistan",
         scene = list(xaxis = list(title = "Age"),
                      yaxis = list(title = "Loneliness"),
                      zaxis = list(title = "Life Satisfaction")))


# Load necessary library
library(plotly)

# Create 3D scatter plot
plot_ly(other_countries_data, x = ~age, y = ~lone01, z = ~`lifeSat`,
        color = ~`lifeSat`,
        colors = c("#440154", "#3B528B", "#21918C", "#5DC863", "#FDE725"),
        marker = list(size = 5),
        type = "scatter3d", mode = "markers",
        text = paste("Age: ", other_countries_data$age, "<br>",
                     "Loneliness: ", other_countries_data$lone01, "<br>",
                     "Life Satisfaction: ", other_countries_data$`lifeSat`),
        hoverinfo = "text") %>%
  layout(title = "Relationship between Age, Loneliness, and Life Satisfaction of Other
Countries",
         scene = list(xaxis = list(title = "Age"),
                      yaxis = list(title = "Loneliness"),
                      zaxis = list(title = "Life Satisfaction")))

#2(b)
# Since only one country no need for this column
pakistan_data <- pakistan_data[, !(names(pakistan_data) %in% c("coded_country"))]
```

```
# Fit linear regression model for c19ProSo01, c19ProSo02, c19ProSo03, c19ProSo04
fit_c19ProSo01 <- lm(c19ProSo01 ~ ., data = pakistan_data)
fit_c19ProSo02 <- lm(c19ProSo02 ~ ., data = pakistan_data)
fit_c19ProSo03 <- lm(c19ProSo03 ~ ., data = pakistan_data)
fit_c19ProSo04 <- lm(c19ProSo04 ~ ., data = pakistan_data)

# Check all the models summary
summary(fit_c19ProSo01)
summary(fit_c19ProSo02)
summary(fit_c19ProSo03)
summary(fit_c19ProSo04)

#2 c
# Since only one country no need for this column
other_countries_data <- other_countries_data[, !(names(other_countries_data) %in%
c("coded_country"))]

# Fit linear regression model for c19ProSo01, c19ProSo02, c19ProSo03, c19ProSo04
fit_c19ProSo01 <- lm(c19ProSo01 ~ ., data = other_countries_data)
fit_c19ProSo02 <- lm(c19ProSo02 ~ ., data = other_countries_data)
fit_c19ProSo03 <- lm(c19ProSo03 ~ ., data = other_countries_data)
fit_c19ProSo04 <- lm(c19ProSo04 ~ ., data = other_countries_data)


# Check all the models summary
summary(fit_c19ProSo01)
summary(fit_c19ProSo02)
summary(fit_c19ProSo03)
summary(fit_c19ProSo04)

#--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
# Question 3

# 3a

# Preparing the data since we got it from two different sources
#since we want to read xlsx we need this package
install.packages("readxl")
library(readxl)
library(dplyr)
# reading the first file
file1 <-read_excel("C:/Users/Home/OneDrive/Desktop/3152/governancedata.xlsx")
#reading the second file
file2 <- read.csv("C:/Users/Home/OneDrive/Desktop/3152/Healthdata.csv")
# checking for na
na_count <- sapply(file1, function(x) sum(is.na(x)))
```

```
na_count[na_count > 0]
#getting rid of all the na values
file1 <- file1 %>%
  na.omit()
# in file 2 we had a lot of complex predictors hence I picked the ones needed
file2 <- file2[, c("X1.6..Immunization", "X1.6.1..Vaccination.rates", "OVERALL.SCORE",
"Year", "Country",
"X2..EARLY.DETECTION...REPORTING.FOR.EPIDEMICS.OF.POTENTIAL.INT.L.CONCER
N")]
#change column names as per my need
colnames(file1)[which(colnames(file1) == "Time")] <- "Year"
colnames(file1)[which(colnames(file1) == "Country Name")] <- "Country"
colnames(file1)[which(colnames(file1) == "Control of Corruption: Estimate [CC.EST]")] <-
"Control of Corruption Estimate"
colnames(file1)[which(colnames(file1) == "Government Effectiveness: Estimate [GE.EST]")]
<- "Government Effectiveness Estimate"
colnames(file1)[which(colnames(file1) == "Political Stability and Absence of
Violence/Terrorism: Estimate [PV.EST]")] <- "Political Stability and Absence of Terrorism
Estimate"
colnames(file1)[which(colnames(file1) == "Political Stability and Absence of
Violence/Terrorism: Estimate [PV.EST]")] <- "Political Stability and Absence of Terrorism
Estimate"

colnames(file2)[which(colnames(file2) == "X1.6.1..Vaccination.rates")] <- "Vaccination_rates"
colnames(file2)[which(colnames(file2) == "X1.6..Immunization")] <- "Immunization"

colnames(file2)[which(colnames(file2) == "OVERALL.SCORE")] <- "Overall_Health_Score"

colnames(file2)[which(colnames(file2) ==
"X2..EARLY.DETECTION...REPORTING.FOR.EPIDEMICS.OF.POTENTIAL.INT.L.CONCER
N")] <-
"EARLY_DETECTION/REPORTING.FOR.EPIDEMICS.OF.POTENTIAL.INT.L.CONCERN"

file1 <- subset(file1, select = -c(`Country Code`,`Time Code`))
print(colnames(file1))



# Merge file1 and file2 based on "Country" and "Year" columns
merged_data <- merge(file1, file2, by = c("Country", "Year"), all = TRUE)

# since my data is from 2019 and 2021 I wanna get an avg hence changing column names
to numeric from character
merged_data <- merged_data %>%
  mutate(`Control of Corruption Estimate` = as.numeric(`Control of Corruption Estimate`),
         `Government Effectiveness Estimate` = as.numeric(`Government Effectiveness
Estimate`),
```

```
`Political Stability and Absence of Terrorism Estimate` = as.numeric(`Political Stability and Absence of Terrorism Estimate`))

# getting rid of na values
merged_data <- na.omit(merged_data)
#install.packages("knitr")
#library(knitr)

# Print the table using kable
#kable(merged_data)

#we have our final data which has all the avg
countries_data <- merged_data %>%
  group_by(Country) %>%
  summarise(across(everything(), mean, na.rm = TRUE))%>%
  select(-Year)
# checking for na
na_count <- sapply(countries_data, function(x) sum(is.na(x)))
na_count[na_count > 0]

# Next step is to perform clustering and hence we first perform scaling
library(dplyr)
install.packages("tidyr")
library(tidyr)

# Select numeric columns for scaling
numeric_cols <- select(countries_data, -Country)

# Scale the numeric columns
scaled_data <- scale(numeric_cols)

# Combine scaled numeric columns with non-numeric columns
scaled_avg_data <- bind_cols(select(countries_data, Country), as.data.frame(scaled_data))


# Print the first few rows of scaled data
print(colnames(scaled_avg_data))

library(ggplot2)

# Calculate the within-cluster sum of squares (WCSS) for different values of k
wcss <- numeric(length = 10)
for (i in 1:10) {
  countryclust <- kmeans(scaled_avg_data[, -1], centers = i)
  wcss[i] <- countryclust$tot.withinss
}

# Plot the elbow method
```

```
plot(1:10, wcss, type = "b", pch = 19, frame = FALSE, xlab = "Number of Clusters (k)", ylab = "Within-Cluster Sum of Squares (WCSS)",
     main = "Elbow Method for Optimal Number of Clusters")
# as per the plot we see elbow between 4 and 6 hence k = 5
# Set seed for reproducibility
set.seed(123)
# Perform k-means clustering
k <- 5  # You can adjust the number of clusters as needed
countryclust <- kmeans(scaled_avg_data[, -1], centers = k)  # Excluding 'Country' for clustering
print(countryclust)

# Extract cluster assignments from clustering results
cluster_assignments <- countryclust$cluster

# Find the index of the focus country (e.g., Pakistan) in the dataset
focus_country_index <- which(scaled_avg_data$Country == "Pakistan")

# Find the cluster assignment of the focus country
focus_country_cluster <- cluster_assignments[focus_country_index]

# Find countries in the same cluster as the focus country
similar_countries <- scaled_avg_data$Country[cluster_assignments == focus_country_cluster]

# Display the similar countries
print(similar_countries)


# 3b

# List of countries similar to Pakistan's cluster
countries_to_check <- c("Afghanistan", "Algeria", "Angola", "Benin", "Bolivia", "Burkina Faso", "Burundi", "Cambodia", "Cameroon", "Central African Republic", "Chad", "Comoros", "Djibouti", "Dominican Republic", "Eritrea", "Ethiopia", "Guatemala", "Guinea", "Guinea-Bissau", "Haiti", "Honduras", "Iraq", "Lebanon", "Lesotho", "Libya", "Madagascar", "Malawi", "Mali", "Mozambique", "Myanmar", "Nepal", "Niger", "Nigeria", "Pakistan", "Papua New Guinea", "Somalia", "South Sudan", "Sudan", "Suriname", "Tajikistan", "Tanzania", "Togo", "Uganda", "Ukraine", "Zimbabwe")

# Check if countries exist in cleaned_cvbase
existing_countries <- countries_to_check[countries_to_check %in% cleaned_cvbase$coded_country]

# Print the existing countries
print(existing_countries)
# Filter the cleaned_cvbase dataset for the existing countries
```

```
similar_countries_data <- cleaned_cvbase[cleaned_cvbase$coded_country %in%
existing_countries & cleaned_cvbase$coded_country != "Pakistan", ]

# Check the dimensions of the filtered dataset
dim(similar_countries_data)
print
# check all exist or not
print(unique(similar_countries_data$coded_country))
# Since only one country no need for this column
similar_countries_data <- similar_countries_data[, !(names(similar_countries_data) %in%
c("coded_country"))]
# Fit linear regression model for c19ProSo01, c19ProSo02, c19ProSo03, c19ProSo04
fit_proso_01 <- lm(c19ProSo01 ~ ., data = similar_countries_data)
fit_proso_02 <- lm(c19ProSo02 ~ ., data = similar_countries_data)
fit_proso_03 <- lm(c19ProSo03 ~ ., data = similar_countries_data)
fit_proso_04 <- lm(c19ProSo04 ~ ., data = similar_countries_data)
# Check all the models summary
summary(fit_proso_01)
summary(fit_proso_02)
summary(fit_proso_03)
summary(fit_proso_04)
```

Reference for the data in 3a

Global Health Security Index: Reports and Data
https://www.ghsindex.org/report-model/

World Health Organization
https://www.who.int/