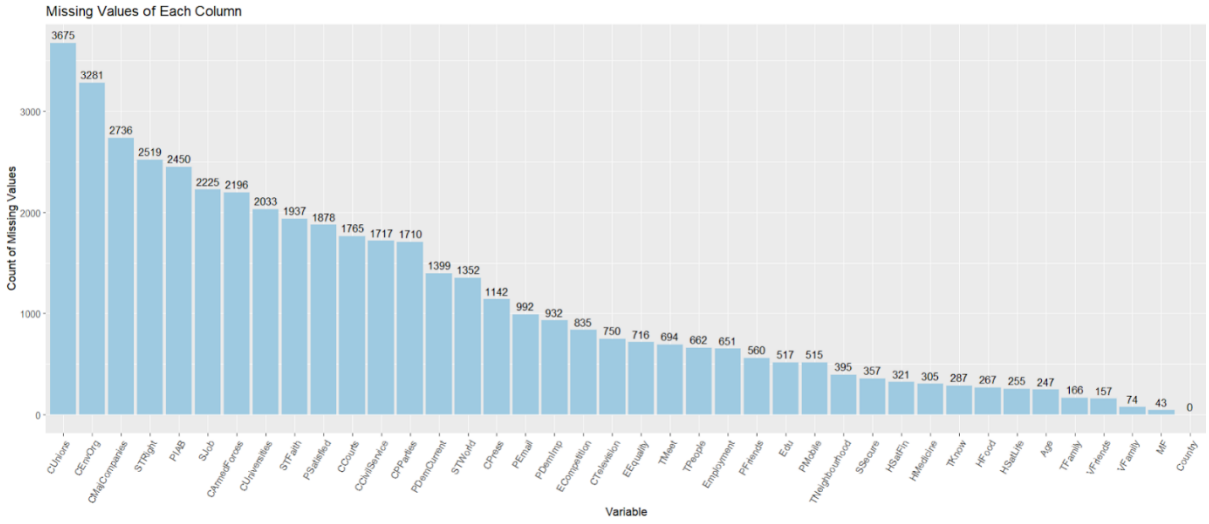
WORLD VALUES SURVEY ANALYSIS

Name: Chai Shou Zheng
Focus Country: JOR (Jordan)

Descriptive analysis

Dimensions:
The "WVSExtract_34035958.csv" dataset comprises **50,000 rows** and **40 columns**.

Missing Value:



In the dataset, missing values are not stored as NA, but are instead coded using specific negative values: -1, -2, -3, -4, and -5, each representing different types of non-responses. The bar chart shows that missing values are common across many variables. Among all the variables, CUnions has the highest number of missing values, while the Country variable has no missing values at all.

Data Pre-Processing and Data Cleaning:

- Replacing negative coded values that indicate missing values with NA

```
Summary of CUnions AFTER cleaning:
> summary(wvs_data_clean$CUnions)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  1.000   2.000   3.000   2.703   3.000   4.000    3675
```

```
Summary of CUnions BEFORE cleaning:
> summary(wvs_data$CUnions)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -5.000   2.000   3.000   2.412   3.000   4.000
```

As shown in the example of the variable CUnions, after data cleaning, the negative values have been removed and replaced with NA. The minimum value after cleaning is now 1, indicating that only valid responses are retained.

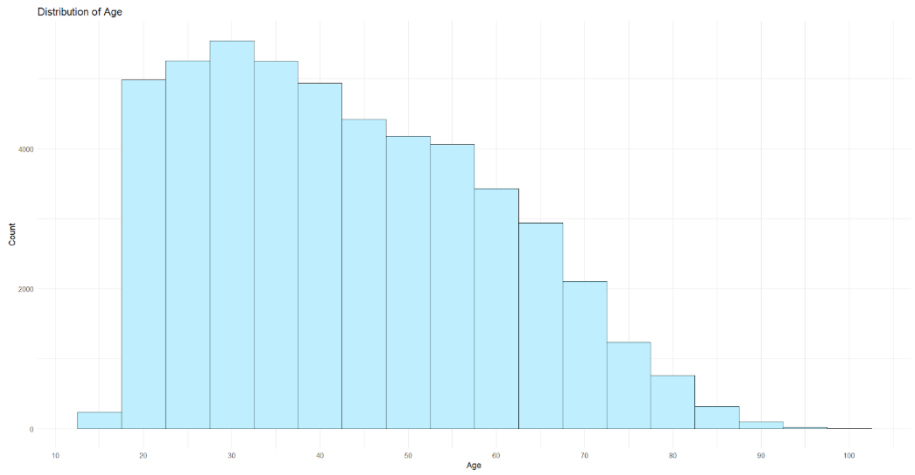
Data Types:

In the dataset, all variables are stored as integers except for the Country, which is stored as a character string. Country is a non-numerical attribute that contains alphabetical country codes and should be treated as a categorical nominal variable.

Many of the categorical variables in the dataset have been numerically encoded. Although they are stored as integers, they represent categorical data, where each numeric value indicates agreement levels. The higher the numeric value generally indicates greater agreement with the statement or frequency of occurrence, making them ordinal categorical variables rather than true numerical measures. Additionally, the dataset includes categorical nominal variables that represent distinct, unordered categories or options.

Categorical Nominal	PIAB, MF, Employment, Country
Categorical Ordinal / Numerical Discrete	TPeople, TFamily, TNeighbourhood, TKnow, TMeet, VFamily, VFriends, HSatLife, HSatFin, HFood, HMedicine, EEquality, ECompetition, SSecure, SJob, STFaith, STRight, STWorld, PMobile, PEmail, PFriends, PDemImp, PDemCurrent, PSatisfied, Edu
Numerical Discrete	Age

Distribution of Numerical Attributes (Age)



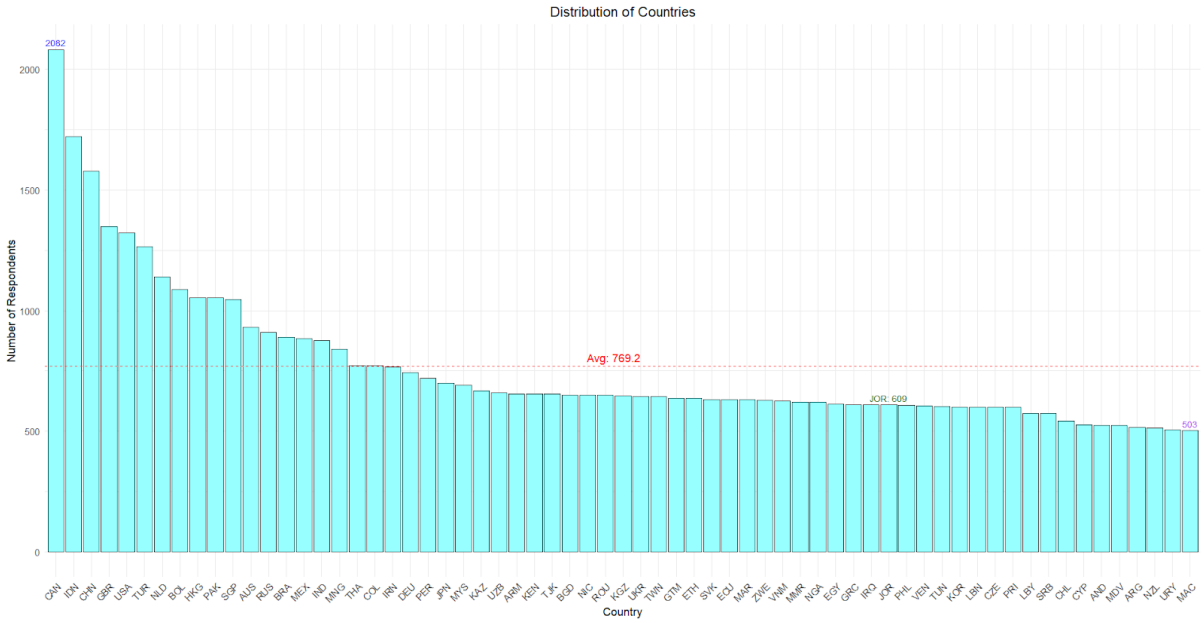

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  16.00   29.00   41.00   43.12   55.00  100.00     247
```

The distribution of age is right-skewed, with a higher concentration of respondents between the ages of 20 and 50. The most common age group appears to be around 30 to 35 years old. Based on the summary statistics, the minimum recorded age is 16, while the maximum is 100. The first quartile is 29, the median is 41, and the third quartile is 55, indicating that 50% of respondents are aged between 29 and 55. The average age is 43.12, slightly higher than the median, which supports the observation of a right-skewed distribution. There are 247 missing age values in the dataset. There is a gradual decline in the number of participants as the age increases beyond 50, with relatively few respondents aged above 80. Additionally, there are a small number of extreme values below 20 and above 90.

Description of Coded_Country

Frequency table for 'Country'

CAN	IDN	CHN	GBR	USA	TUR	NLD	BOL	HKG	PAK	SGP	AUS	RUS	BRA	MEX	IND	MNG	THA	COL	IRN	DEU	PER
2082	1722	1579	1349	1321	1265	1140	1087	1054	1054	1047	930	910	889	885	877	841	773	772	768	744	721
JPN	MYS	KAZ	UZB	ARM	KEN	TJK	BGD	NIC	ROU	KGZ	UKR	TWN	GTM	ETH	SVK	ECU	MAR	ZWE	VNM	MMR	NGA
698	691	667	661	654	654	654	650	649	649	648	645	644	637	636	632	630	630	628	626	621	621
EGY	GRC	IRQ	JOR	PHL	VEN	TUN	KOR	LBN	CZE	PRI	LBY	SRB	CHL	CYP	AND	MDV	ARG	NZL	URY	MAC	
614	611	611	609	607	606	602	601	600	599	599	575	574	543	526	525	525	516	514	505	503	



Observations:

The graph displays the number of survey participants for each country. The X-axis corresponds to a distinct country, and the Y-axis represents the number of participants from that country.

As we can observe from the frequency table and the graph, the highest number of respondents is 2082, observed in CAN, followed by IDN with 1,722 respondents. The lowest number of respondents is 503, observed in MAC. Our assigned research country JOR has 609 respondents, placing it toward the lower end of the distribution compared to many other countries.
On average, there are about 796.2 respondents per country across the dataset.

Analysis of the distribution of countries and discussion of the impact:

This uneven distribution implies that any cross-country comparison needs to be approached with caution. Overrepresented countries may drive overall trends, and analyses involving underrepresented countries like JOR might be less accurate due to lower statistical power.

Countries with a large number of observations (e.g., CAN, IDN) can dominate overall trends and summary statistics. This means that the aggregate patterns we observe may primarily reflect the opinions and behaviors of respondents from these highly represented countries. If the analysis is not adjusted for the sample size, the overrepresented countries may skew the results.

Countries with relatively few observations (e.g., JOR, MAC) have less statistical power. This makes it more challenging to detect significant effects or differences within these subgroups. The smaller sample size increases the risk of random error and may lead to unreliable estimates. Underrepresented countries may yield estimates with wider confidence intervals, making any inferences drawn from them less reliable.

Multivariate Graph of Life Satisfaction by Age and Gender

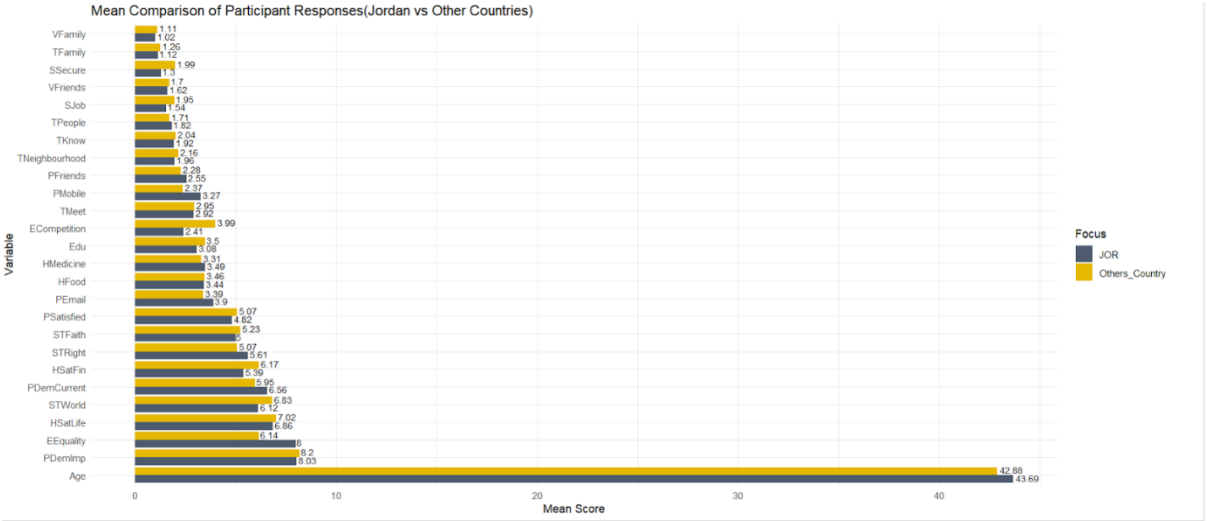
Male (Left)

- Life satisfaction tends to increase from younger ages (20s) and peaks around the 30–40 age range.
- After that, it shows a decline in middle age (40s to 50s), followed by a slight rise and stabilization into older age.
- There's more variability in younger and older ages as shown by the steeper trend lines at both ends of the age spectrum.

Female (Right)

- Satisfaction is fairly stable in early life, showing a gradual increase through middle age.
- Then it seems to peak around age 50–60 and slightly decline into older age.
- The curve is smoother and less sharp than for males, suggesting less variation in trends over age.

2. Focus country vs all other countries as a group.
Mean Comparisons of Participant Responses between JOR and Others_Country

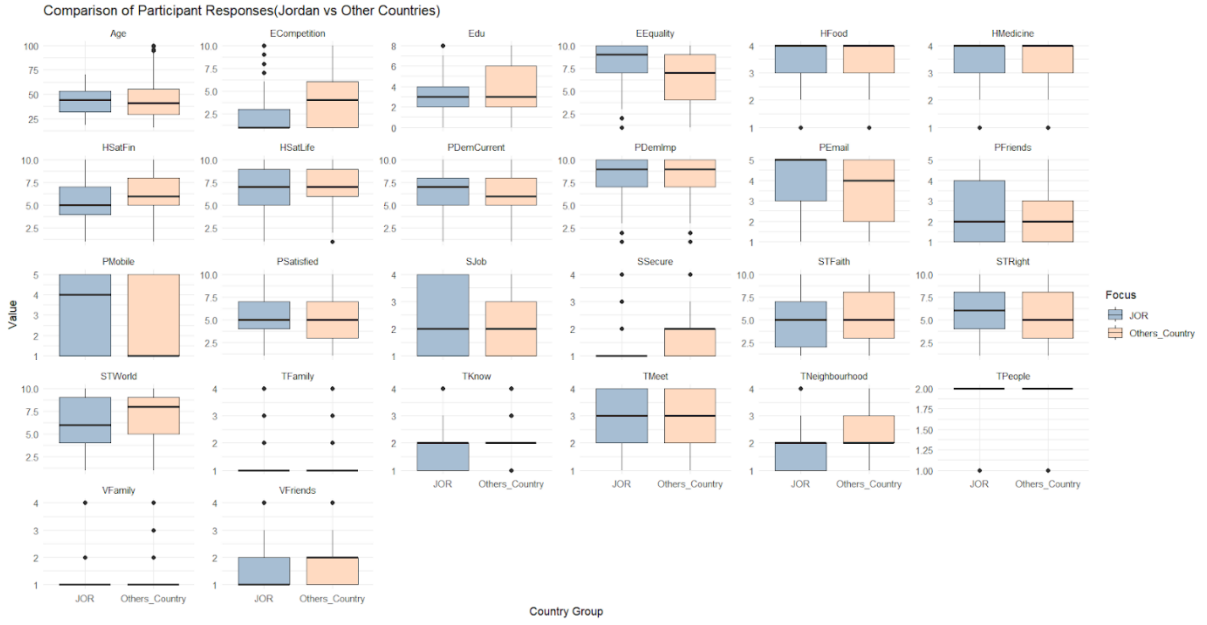


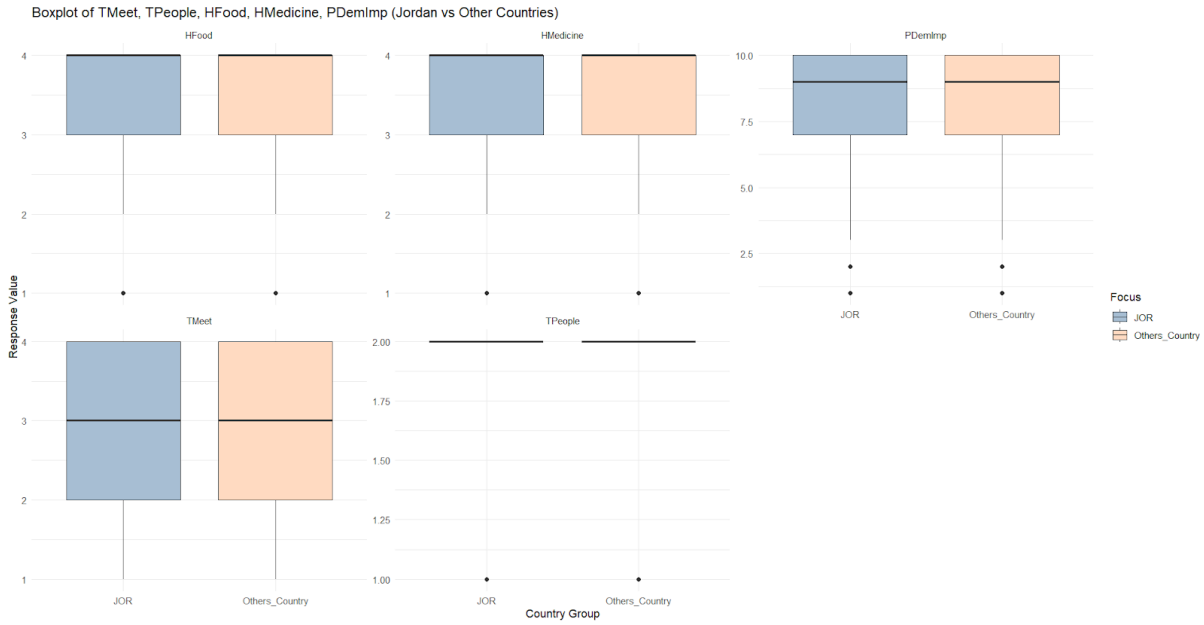
Jordan reports higher means in **TPeople, PF riends, PMobile, PEmail, HMedicine, EEquality, PDemCurrent, STRight,** and **Age**.

Conversely, other countries show higher means in **TFamily, TNeighbourhood, TKnow, TMeet, VFamily, VFriends, HSatLife, HSatFin, HFood, ECompetition, SSecure, SJob, PDemImp, PSatisfied, STFaith, STWorld,** and **Edu**.

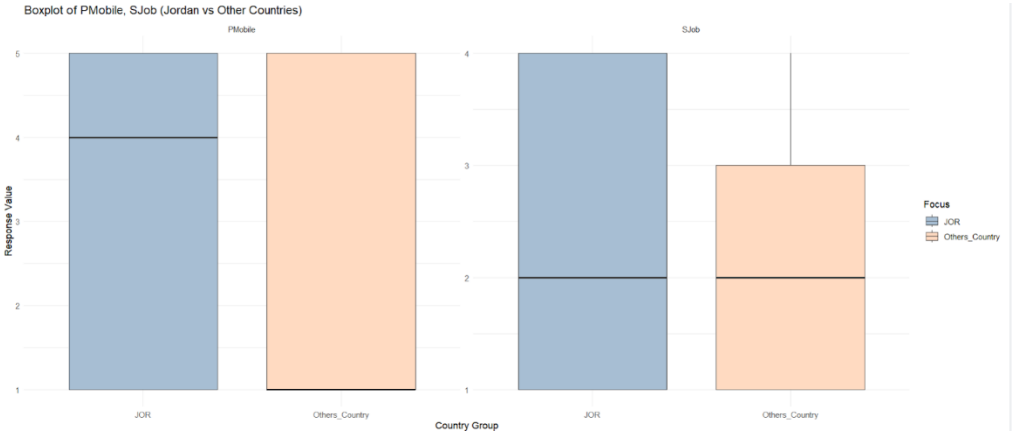
Overall, other countries report a higher number of participant responses with greater mean values compared to Jordan.

Box Plot Comparisons of Participant Responses between JOR and Others_Country

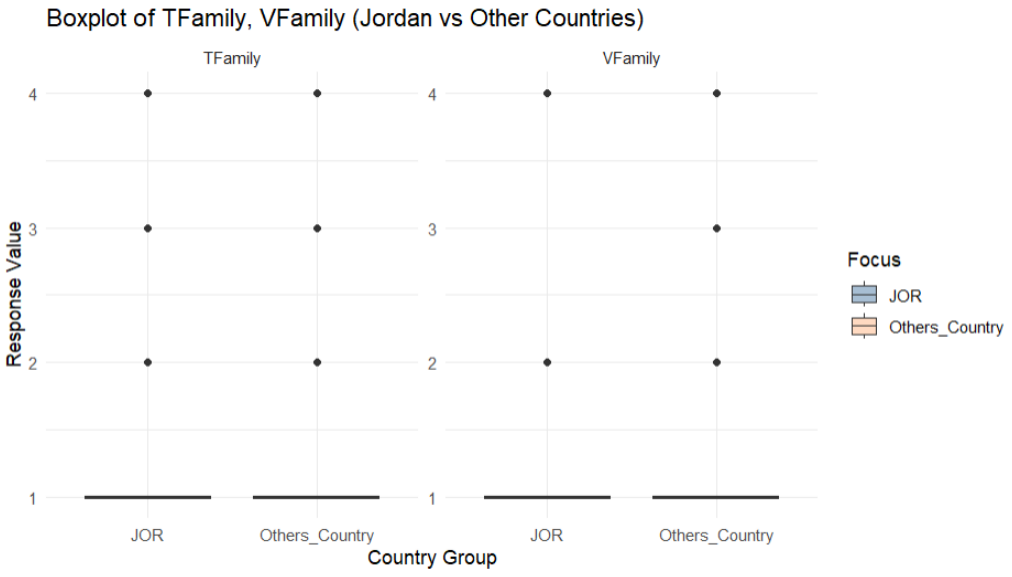




The boxplots for **TMeet, TPeople, HFood, HMedicine, and PDemImp** show that the distributions of participant responses in Jordan and Other Countries are almost the same. These variables have similar medians, interquartile ranges, whisker lengths, and numbers of outliers, indicating that the overall spread and central tendencies of participant responses do not differ much between the two groups.



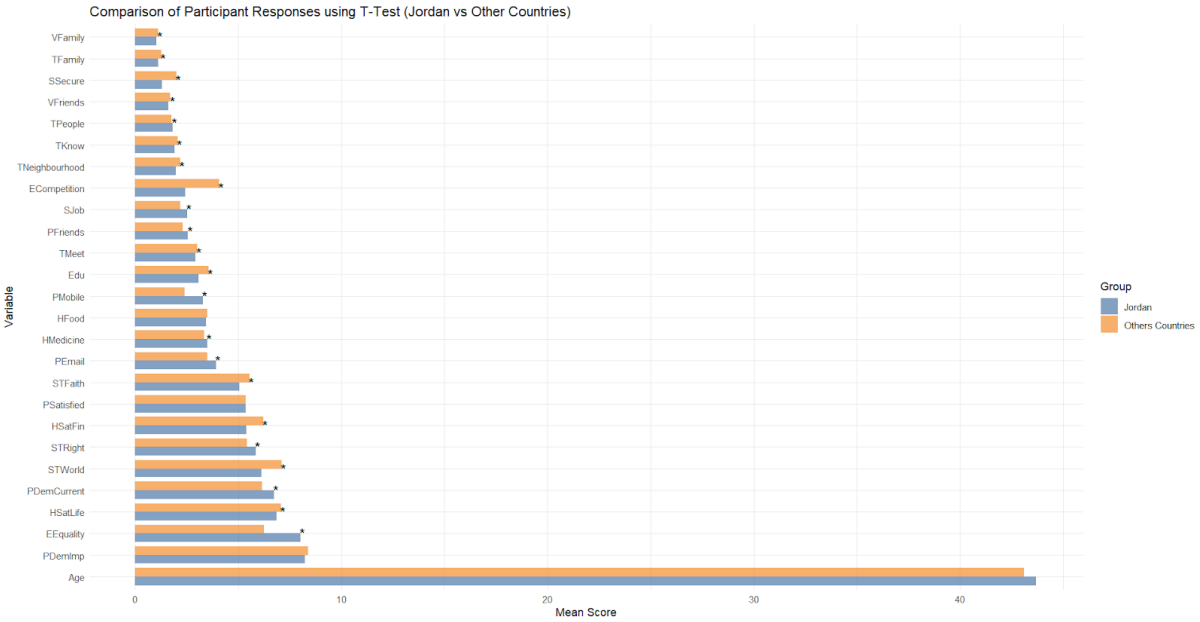
The boxplots for **PMobile and SJob** show a wide range of responses for both Jordan and Other Countries, indicating that participant responses for these two variables are highly varied across both groups.



On the other hand, for **TFamily** and **VFamily,** although the medians are both at 1 for Jordan and other countries, indicating that most participants selected the lowest response, there are many outliers present. This suggests that while the majority responded similarly, a notable number of participants gave higher ratings, pointing to some variation.

Overall, while a few variables such as TPeople, TMeet and PDemImp show participant responses that are almost the same between Jordan and other countries, the majority of the participant responses (attributes) show differences, suggesting a moderate to high level of variability in participant responses across both Jordan and other countries.

Comparison of Participant Responses using T-test



Observation

The graph displays the results of a t-test comparing participant responses between Jordan (in blue) and other countries (in orange). Variables marked with an asterisk (*) indicate a statistically significant difference ($p < 0.05$).

Jordan scores higher in:

TPeople, HMedicine, EEquality, SJob, PDemCurrent, STRight, PMobile, PEmail, PFriends

Other countries score higher in:

TFamily, TNeighbourhood, TKnow, TMeet, VFamily, VFriends, HSatLife, HSatFin, ECompetition, SSecure, STFaith, STWorld, Edu

Variables with no significant difference:

HFood, PDemImp, PSatisdied, Age

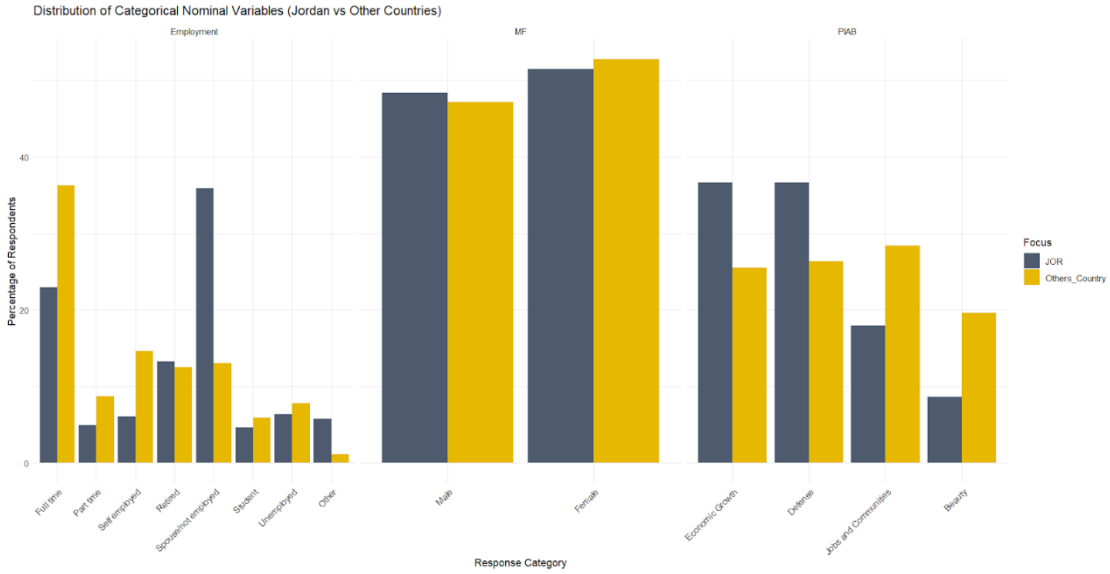
Overall, participant responses for Jordan differ notably from other countries across most attributes, with Jordan scoring higher on some and other countries scoring higher on others.. However, a few variables show no significant difference, indicating some common ground in responses as well.

Categorical Nominal Attribute comparisons between JOR and Others_Country based on Mode

	Focus	PIAB	Employment	MF
	<fct>	<fct>	<fct>	<fct>
1	JOR	Defense	Spouse/not employed	Female
2	Others_Country	Jobs and Communities	Full time	Female

We computed the mode for each nominal variable (PIAB, Employment, and MF) separately for participants from Jordan and other countries to identify the most common responses within each group and examine whether participant responses for our focus country differ from those of other countries. The result shows that most participants from Jordan were female, prioritized Defense as the most important issue for government spending, were not employed (spouse). In contrast, most participants from other countries were also female, prioritized Jobs and Communities, were employed full-time. This suggests that while gender distribution is similar across both groups, while other categorical nominal variables differ between Jordan and other countries.

Distribution of Categorical Nominal Variables (JOR vs Other Countries)



The graph shows clear differences in categorical responses between Jordan and other countries. Jordan shows higher proportions of respondents who are spouse/not employed or retired. In contrast, other countries have more full-time workers. However, for the variable Gender, there is no substantial difference between the two groups. These patterns align with the earlier mode table and chi-square test results.

2(b).

We used linear regression to predict confidence in social organisations in Jordan. The linear regression included all participant response variables, but this increases the risk of overfitting. To reduce this risk, we applied stepwise regression to eliminate less useful predictors that don't significantly improve the model. From the table below, we can see that the resulting Multiple R-squared values ranged from approximately 0.060 to 0.156. This means that the participant response variables explained between 6.0% and 15.6% of the variability in confidence levels. These values indicate relatively low explanatory power, suggesting that the models perform weakly in predicting confidence in social organisations. While the predictors offer some insight, a substantial portion of the variation remains unexplained, implying that other unmeasured

factors may influence confidence levels.

```
----- Linear Regression R-squared Values (JOR) -----
> print(r_squared_df)
       CVariable  R_squared
1  CUniversities 0.08704317
2   CArmedForces 0.11530239
3         CPress 0.11361204
4    CTelevision 0.12284027
5        CUnions 0.15583310
6        CCourts 0.11970611
7      CPParties 0.14754774
8  CCivilService 0.10686206
9  CMajCompanies 0.13326881
10       CEnvOrg 0.11369691
```

```
----- Stepwise Regression R-squared Values (JOR) -----
> print(stepwise_r_squared_df)
       CVariable  R_squared
1  CUniversities 0.06072277
2   CArmedForces 0.09178255
3         CPress 0.07932621
4    CTelevision 0.10013156
5        CUnions 0.12564694
6        CCourts 0.10120117
7      CPParties 0.11623645
8  CCivilService 0.08187342
9  CMajCompanies 0.10788924
10       CEnvOrg 0.09041802
```

Then, we extracted the best predictors for each social organisation which have p value <0 .05 and counted how frequently each predictor appeared across models to identify the most consistently important predictors.

Best predictors from Linear Regression (JOR)

```
       CVariable                                                                  Best_Predictors
1  CUniversities                                                          HSatFin, STRight, Edu
2   CArmedForces                                        SSecure, STWorld, PDemCurrent, MFFemale
3         CPress                                               TKnow, STRight, EmploymentOther
4    CTelevision                                                    TKnow, MFFemale, Edu
5        CUnions                                PIABDefense, STRight, MFFemale, Edu, EmploymentOther
6        CCourts                                     TNeighbourhood, HSatLife, HFood, SSecure
7      CPParties TKnow, PIABJobs and Communities, PEmail, MFFemale, EmploymentSpouse/not employed
8  CCivilService                                  ECompetition, PIABJobs and Communities, MFFemale
9  CMajCompanies                                        HSatFin, SSecure, PSatisfied, Age, Edu
10       CEnvOrg            TPeople, HSatFin, SSecure, SJob, PMobile, Age, Edu, EmploymentStudent
```

```
----- Top 5 Best Predictors from Linear Regression (JOR) -----
> print(top_5_lm_predictors)
all_lm_significant_predictors
     Edu MFFemale  SSecure  HSatFin  STRight
       5        5        4        3        3
```

Best predictors from Stepwise Regression (JOR)

```
       CVariable                                                          Best_Predictors
1  CUniversities                                                  HSatFin, STRight, Edu
2   CArmedForces                                SSecure, STFaith, STWorld, PDemCurrent
3         CPress                          TKnow, ECompetition, STRight, PDemCurrent, Edu
4    CTelevision                                    TKnow, PDemCurrent, MFFemale, Edu
5        CUnions TNeighbourhood, SJob, PIABDefense, STRight, PEmail, MFFemale, Edu
6        CCourts                          TNeighbourhood, HSatLife, SSecure, PDemCurrent
7      CPParties                 TKnow, PIABJobs and Communities, PEmail, MFFemale, Edu
8  CCivilService                  ECompetition, PIABJobs and Communities, MFFemale, Edu
9  CMajCompanies                        HSatFin, SSecure, PSatisfied, MFFemale, Age, Edu
10       CEnvOrg          TPeople, HSatFin, EEquality, SSecure, PMobile, MFFemale, Edu
```

```
----- Top 5 Best Predictors from Stepwise Regression (JOR) -----
> print(top_5_predictors)
all_significant_predictors
       Edu    MFFemale PDemCurrent     SSecure     HSatFin
         8           6           4           4           3
```

The best predictors for JOR are **Edu, MFFemale, SSecure, HSatFin, STRight,** and **PDemCurrent**

We also extracted the least frequent predictors to find the worst predictors

```
----- Bottom 3 Least Frequent Predictors from Linear Regression (JOR) -----
> print(bottom_3_lm_predictors)
all_lm_significant_predictors
       STWorld TNeighbourhood        TPeople
             1             1              1
----- Bottom 3 Least Frequent Predictors from Stepwise Regression (JOR) -----
> print(bottom_3__predictors)
all_significant_predictors
STFaith STWorld TPeople
      1       1       1
```

The worst predictors for JOR are **STWorld, TNeighbourhood, TPeople,** and **STFaith.**

Confidence in social organisations that can be more reliably predicted:

- **CUnions**
- **CPParties**

In the linear regression models for Jordan, confidence in CUnions (R^2 = 0.156) and CPParties (R^2 = 0.147) had the highest R^2 values, indicating that these are the most reliably predicted compared to others. The stepwise regression further found that CUnions (R^2 = 0.126) and CPParties (R^2 = 0.116) had the highest R^2 values. In contrast, organisations like CUniversities had lowest R^2 values (R^2 = 0.087) in linear regression and (R^2 = 0.061) in stepwise, suggesting weakest predictive performance.

2(c).

In the group of other countries, the resulting Multiple R-squared values ranged from approximately 0.047 to 0.124. This means that the participant response variables explained between 4.7% and 12.4% of the variability in confidence levels. These values indicate relatively low explanatory power, suggesting that the models perform weakly in predicting confidence in social organisations. While the predictors offer some insight, a substantial portion of the variation remains unexplained, implying that other unmeasured factors may be influencing confidence levels.

```
----- Linear Regression R-squared Values (Others Country) -----
> print(r_squared_df)
       CVariable  R_squared
1  CUniversities 0.06176678
2   CArmedForces 0.10938001
3         CPress 0.08739966
4    CTelevision 0.10455930
5         CUnions 0.06515060
6         CCourts 0.11644389
7      CPParties 0.12437800
8  CCivilService 0.12145105
9  CMajCompanies 0.05629152
10        CEnvOrg 0.04715182

----- Stepwise Regression R-squared Values (Others Country) -----
> print(stepwise_r_squared_df)
       CVariable  R_squared
1  CUniversities 0.06173730
2   CArmedForces 0.10933633
3         CPress 0.08727751
4    CTelevision 0.10454106
5         CUnions 0.06500043
6         CCourts 0.11636645
7      CPParties 0.12430470
8  CCivilService 0.12126508
9  CMajCompanies 0.05620774
10        CEnvOrg 0.04703380
```

Confidence in social organisations that can be more reliably predicted:

- **CPParties**
- **CCivilService**

In both the linear regression and stepwise models for other countries, confidence in CPParties (R^2 = 0.124) and CCivilService (R^2 = 0.121) had the highest R^2 values, indicating that these are the most reliably predicted compared to others. In contrast, organisations like CEnvOrg had lowest R^2 values (R^2 = 0.047) in linear regression and stepwise, suggesting weakest predictive performance.

Then, we extracted the best predictors for each social organisation and counted how frequently each predictor appeared across models to identify the most consistently important predictors.

```
----- Top 5 Best Predictors from Linear Regression (Other Country) -----
> print(top_5_lm_predictors)
all_lm_significant_predictors
       Edu PSatisfied    SSecure    STWorld    TFamily
        10         10         10         10         10
----- Top 5 Best Predictors from Stepwise Regression (Other Country) -----
> print(top_5_predictors)
all_significant_predictors
       Edu PSatisfied    SSecure    STWorld    TFamily
        10         10         10         10         10
```

The best predictors for Other Countries are **Edu, PSatisfied, SSecure, STWorld, TFamily**.

We also extracted the least frequent predictors to find the worst predictors

```
----- Bottom 3 Least Frequent Predictors from Linear Regression (Other Country) -----
> print(bottom_3_lm_predictors)
all_lm_significant_predictors
         PFriends EmploymentRetired   EmploymentOther
                3                 2                 1
----- Bottom 3 Least Frequent Predictors from Stepwise Regression (Other Country) -----
> print(bottom_3__predictors)
all_significant_predictors
         PFriends EmploymentRetired   EmploymentOther
                3                 2                 1
```

The worst predictors are **PFriends, EmploymentRetired, EmploymentOther.**

Best Predictors (Most Frequent Predictors) for JOR and Other Countries

Best Predictors for JOR	Edu, MFFemale, PDemCurrent, STRight, SSecure, HSatFin
Best Predictors for Other Countries	Edu, PSatisfied, SSecure, STWorld, TFamily
Common Best Predictors (Most frequent Predictors) between JOR and Other Countries	Edu, SSecure

Both Jordan and the group of other countries share **Edu** and **SSecure** as common strong predictors, highlighting their consistent importance in shaping public confidence in social organisations across regions. However, differences in the remaining top predictors suggest that

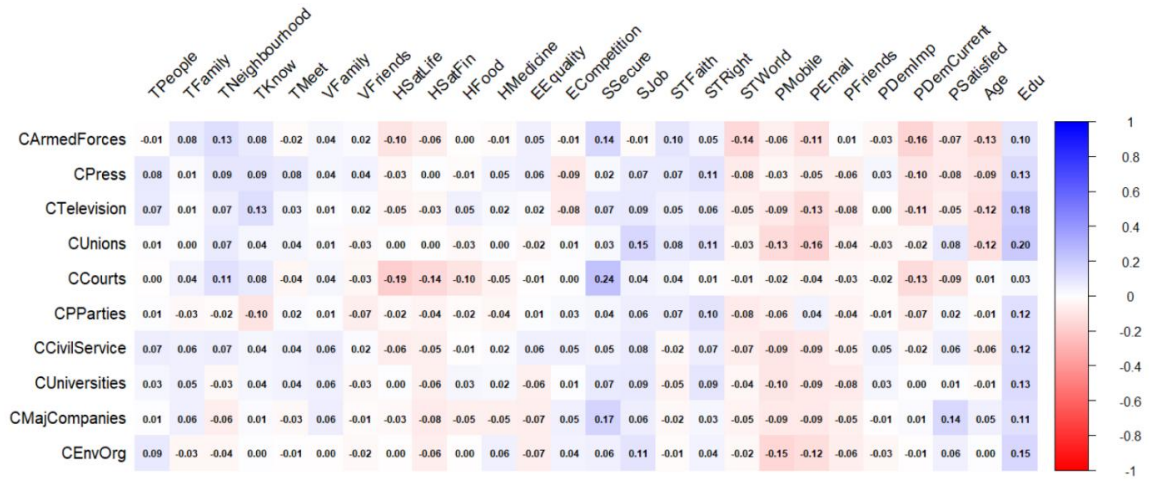
the factors influencing confidence vary by region, reflecting the unique social, political, and cultural contexts of each country.

Worst Predictors (Least Frequent Predictors) for JOR and Other Countries

Worst Predictors for JOR	STWorld, TNeighbourhood, TPeople and STFaith
Worst Predictors for Other Countries	PFriends, EmploymentRetired, EmploymentOther

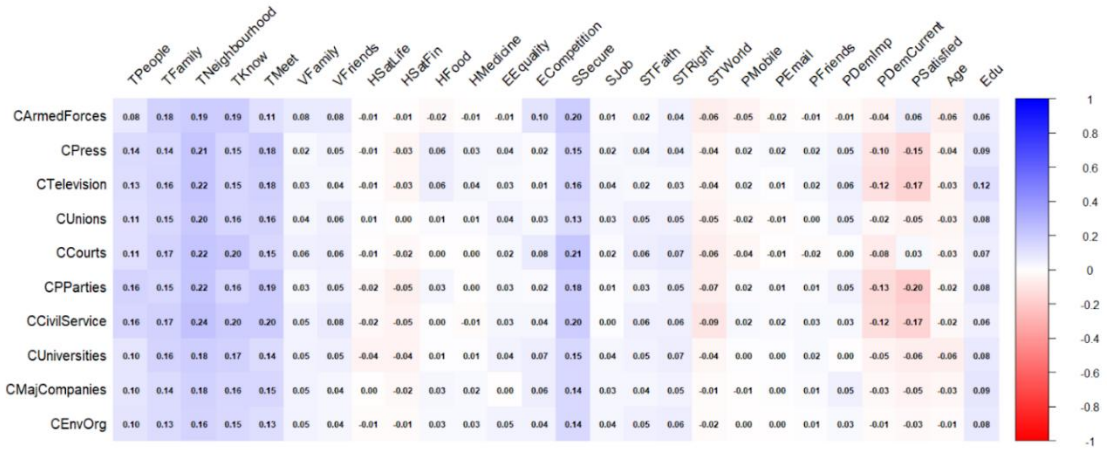
Similarly, weak predictors also vary, further indicating that different attributes influence confidence depending on the region.

Correlation Between Predictor Variables and Confidence in Social Organisations for JOR



We used correlation to examine the relationship between the non-categorical (ordinal/numeric) variables and the target variable for Jordan. The results further support that SSecure and Edu are important predictors, as they exhibit relatively high correlation values with confidence in social organisations.

Correlation Between Predictor Variables and Confidence in Social Organisations for Other Countries



The results further support that SSecure, TFamily, Edu are important predictors for other countries as the relatively high correlation values with confidence in social organisations.

3. Focus country vs cluster of similar countries.

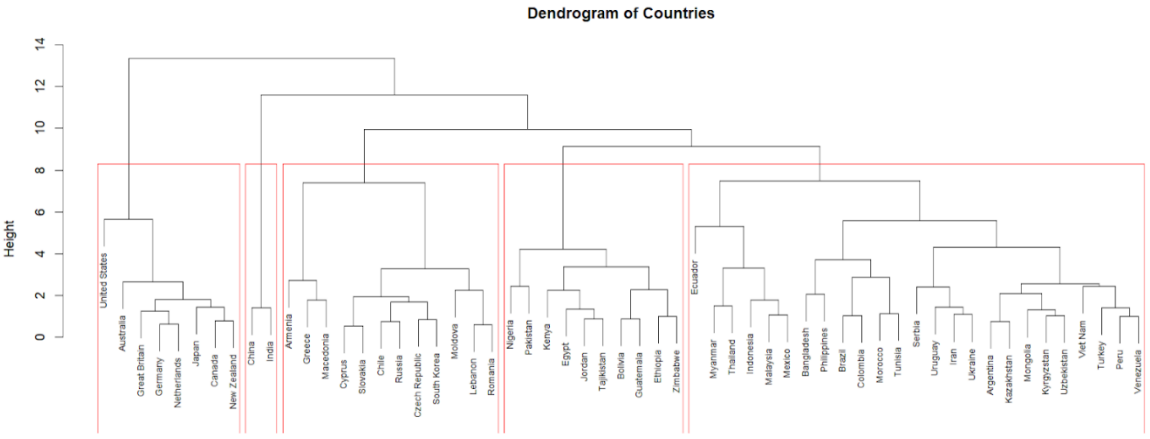
To identify countries that are similar to Jordan, we used hierarchical clustering based on several socio-economic and technological indicators obtained from a dataset country_profile_variables.csv from Kaggle.

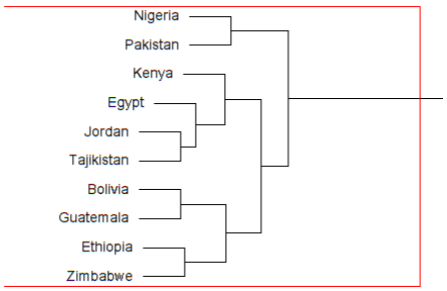
The indicators included

1. Population in 2017 (in thousands)
2. GDP per capita (current US$)
3. Fertility rate (live births per woman)
4. Health expenditure (% of GDP)
5. Unemployment rate (% of labour force)
6. Internet users per 100 inhabitants
7. Primary education gross enrolment ratio (average of male and female).

Since the external dataset did not use abbreviated country codes, we matched the full country names from the World Values Survey (WVS) list with those in the external dataset and filtered out any countries do not present in the WVS. This was done because our analysis was limited to countries included in the WVS, in alignment with the baseline data used in our study, and also to reduce visual clutter in the dendrogram.

Before clustering, we cleaned the data by replacing invalid values such as -99 and ... with NA and then removed any rows with missing values to ensure complete data for clustering. Next, all numeric variables were normalized using z-score scaling so that each feature contributed equally to the clustering process. We applied hierarchical clustering using Ward's method with Euclidean distance and visualized the resulting dendrogram. The dendrogram was then cut into 5 clusters, and countries grouped in the same cluster as Jordan were identified as the most similar based on the selected indicators.





6	Bolivia	4
16	Egypt	4
17	Ethiopia	4
20	Guatemala	4
26	Jordan	4
28	Kenya	4
40	Nigeria	4
41	Pakistan	4
52	Tajikistan	4
64	Zimbabwe	4

Countries that are in the same cluster 4 as Jordan (JOR) include **Egypt (EGY), Bolivia (BOL), Kenya (KEN), Nigeria (NGA), Pakistan (PAK), Ethiopia (ETH) and Zimbabwe (ZWE),** among others.

3(b).

For the cluster of countries most similar to Jordan (Cluster 4), the table below shows multiple R-squared values from the full linear regression models ranged from approximately 0.07 to 0.56. This means that participant response variables explained between 7% and 56% of the variation in confidence levels toward different social organisations. Most of the models had relatively low explanatory power, with R-squared values below 0.10, indicating that only a small portion of the variability in confidence could be attributed to the selected predictors.

```
----- Linear Regression R-squared Values (Cluster 4)
> print(r_squared_df_lm)
       CVariable  R_squared
1  CUniversities 0.07772687
2   CArmedForces 0.56015931
3         CPress 0.09183988
4    CTelevision 0.10078147
5        CUnions 0.12284872
6        CCourts 0.54617540
7      CPParties 0.11912888
8  CCivilService 0.12660820
9  CMajCompanies 0.09406451
10       CEnvOrg 0.07975109
```

```
----- Stepwise Regression R-squared Values (Cluster 4) -----
> print(r_squared_df_step)
       CVariable  R_squared
1  CUniversities 0.07580618
2   CArmedForces 0.55993862
3         CPress 0.08827401
4    CTelevision 0.09949289
5        CUnions 0.12212315
6        CCourts 0.54524574
7      CPParties 0.11757252
8  CCivilService 0.12331635
9  CMajCompanies 0.09278435
10       CEnvOrg 0.07787172
```

However, there were two notable exceptions. Confidence in social organisations can be more reliably predicted are CArmedForces (R^2 = 0.56) and CCourts (R^2 = 0.54). These models demonstrated moderate to strong predictive strength, suggesting that for these particular organisations, the participant response variables were relatively effective at explaining differences in confidence levels.

This indicates that while the overall models perform weakly across most outcomes, there is evidence that specific social organisations can still be reliably predicted using the chosen set of attributes. Therefore, although the models may not capture the full complexity of institutional confidence across the board, they still provide meaningful predictive insights for particular domains such as CArmedForces and CCourts.

When comparing the overall model performance between the cluster of countries similar to Jordan (Cluster 4) and the group of other countries in Question 2(c), the Multiple R-squared values from the full linear regression models were generally higher in Cluster 4. Specifically, R-squared values in Cluster 4 ranged from approximately 0.07 to 0.56 and the R-squared values for the other countries group ranged from around 0.047 to 0.124. This suggests that models based on the cluster of similar countries provided better explanatory power.

Then, we extracted the best predictors for each social organisation and counted how frequently each predictor appeared across models to identify the most consistently important predictors.

```
----- Top 5 Best Predictors from Linear Regression (Cluster 4) -----
> print(top_5_lm)
all_lm_predictors_cluster4
          Edu       SSecure         TKnow TNeighbourhood       STRight
           10            10            10            10             9
```

```
----- Top 5 Best Predictors from Stepwise Regression (Cluster 4) -----
> print(top_5_step)
all_stepwise_predictors_cluster4
           Edu        SSecure          TKnow TNeighbourhood        STRight
            10             10             10             10              9
```

The best 5 predictors for Cluster 4 are **Edu, SSecure, Tknow, Neighbourhood,** and **STRight**

We also extracted the least frequent predictors to find the worst predictors

```
----- Bottom 3 Least Frequent Predictors from Linear Regression (Cluster 4) -----
> print(bottom_3_lm)
all_lm_predictors_cluster4
   HSatLife PIABDefense     TPeople
          1           1           1

----- Bottom 3 Least Frequent Predictors from Stepwise Regression (Cluster 4) -----
> print(bottom_3_step)
all_stepwise_predictors_cluster4
  EEquality     HSatFin PIABDefense
          1           1           1
```

The worst predictors for Cluster 4 are **HSatLife, PIABDefense, TPeople, EEquality, HSatFin**

Best Predictors (Most Frequent Predictors) for Cluster 4 and Other Countries

Best Predictors for Cluster 4	Edu, SSecure, Tknow, Neighbourhood, STRight
Best Predictors for Other Countries	Edu, PSatisfied, SSecure, STWorld, TFamily
Common Best Predictors (Most frequent Predictors) between JOR and Other Countries	Edu, SSecure

When comparing the results from the cluster of countries similar to Jordan (Cluster 4) with those from the group of all other countries, we observe both common and differences in the important predictors of confidence in social organisations. The most frequent predictors in both groups, Edu and SSecure, consistently emerged as significant, suggesting these variables are universally important. However, the clustering identified TKnow, TNeighbourhood, and STRight as additional important predictors, while the other countries group highlighted PSatisfied, STWorld, and TFamily as important predictors.

Worst Predictors (Least Frequent Predictors) for Cluster 4 and Other Countries

Worst Predictors for JOR	HSatLife, PIABDefense, TPeople, EEquality, HSatFin
Worst Predictors for Other Countries	PFriends, EmploymentRetired, EmploymentOther

The least frequent predictors differ substantially. In cluster 4, variables such as HSatLife, PIABDefense, and TPeople had minimal predictive value, whereas in other countries, PFriends and employment statuses like Retired and Other were among the least impactful.

Comparison between the best predictor from group of other countries in 2(c) and the Cluster 4 group of similar countries in 3(b) with best predictors of JOR

Best Predictors for Cluster 4	Edu, SSecure, Tknow, Neighbourhood, STRight
Best Predictors for Other Countries	Edu, PSatisfied, SSecure, STWorld, TFamily
Best Predictors for JOR	Edu, MFFemale, PDemCurrent, STRight, SSecure, HSatFin

When comparing the important predictors of confidence in social organisations, the cluster of similar countries (Question 3b) aligns more closely with our focus country Jordan than the group of other countries (Question 2c), as shown in the table above. Several of Jordan's top predictors Edu, STRight and SSecure also appear in the cluster group, indicating a better match for identifying influential attributes in the focus country JOR. In contrast, only Edu and SSecure overlap between Jordan and the group of other countries. Therefore, the cluster-based grouping in Question 3(b) appears to give a more meaningful and relevant understanding of the factors influencing institutional confidence in Jordan than the broader comparison with all other countries in Question 2(c).

Appendix

Q1(a):

```
# ------------------------------------------------
# R script: FIT3152_Ass1_Q1.R
# Project: FIT3152 Assignment1
#
# Date: 16/4/2025
# Time: 7:20pm
# Author: CHAI SHOU ZHENG
# Assigned Country: JOR (Jordan)
# ------------------------------------------------

# Create individual data set
rm(list = ls())
set.seed(34035958)
VCData = read.csv("WVSExtract.csv")
VC = VCData[sample(1:nrow(VCData),50000, replace=FALSE),]
VC = VC[,c(1:6, sort(sample(7:46,17, replace = FALSE)), 47:53,
        sort(sample(54:69,10, replace = FALSE)))]

# Save the extracted data set as a new CSV file
write.csv(VC, "WVSExtract_34035958.csv", row.names = FALSE)

# Load libraries
library(dplyr)
library(ggplot2)
library(gridExtra)
library(knitr)
library(tidyr)
library(broom)
library(janitor)
library(corrplot)
options(contrasts = c("contr.treatment", "contr.poly"))
setwd("FIT3152/Week1/Assignment1")
getwd()

# Read the extracted data set
wvs_data <- read.csv("WVSExtract_34035958.csv", stringsAsFactors = FALSE)

# ------------------------------------------------
# Q1.Descriptive analysis
# ------------------------------------------------
# 1(a).

# Dimensions
dim(wvs_data)

# Data Types
str(wvs_data)
```

```
# Missing Values
sum(is.na(wvs_data))

# Count of negative survey responses (missing value) per column
coded_missing <- function(x) {
  sum(x %in% c(-1, -2, -3, -4, -5))
}
missing_counts <- sapply(wvs_data, coded_missing)
missing_counts

# Convert to data frame for plotting
missing_df <- data.frame(
  Variable = names(missing_counts),
  Missing = as.numeric(missing_counts)
)

# ------------------------------------------------
# Plot the missing values graph
# ------------------------------------------------
ggplot(missing_df, aes(x = reorder(Variable, -Missing), y = Missing)) +
  geom_bar(stat = "identity", fill = "#9ecae1") +
  geom_text(aes(label = Missing), vjust = -0.5, size = 3.5) +  # Add numbers above bars
  labs(title = "Missing Values of Each Column",
       x = "Variable",
       y = "Count of Missing Values") +
  theme(axis.text.x = element_text(angle = 60, hjust = 1))

# ------------------------------------------------
# Data Pre-Processing and Data Cleaning
# ------------------------------------------------
# Replace coded missing values in all numeric columns only
wvs_data_clean <- wvs_data
# Loop through columns and replace coded values with NA
wvs_data_clean[] <- lapply(wvs_data_clean, function(col) {
  if (is.numeric(col)) {
    col[col %in% c(-1, -2, -3, -4, -5)] <- NA
  }
  return(col)
})

# Summary before cleaning
cat("Summary of CUnions BEFORE cleaning:\n")
summary(wvs_data$CUnions)

# Summary after cleaning
cat("\nSummary of CUnions AFTER cleaning:\n")
summary(wvs_data_clean$CUnions)

# ------------------------------------------------
# Distribution of Numerical Attributes (Age)
# ------------------------------------------------
```

```
# Plot a graph of Distribution of Numerical Attributes (Age)
ggplot(wvs_data_clean, aes(x = Age)) +
  geom_histogram(binwidth = 5, fill = "lightblue1", color = "black") +
  scale_x_continuous(breaks = seq(0, 100, by = 10)) +  # custom x-axis ticks
  labs(title = "Distribution of Age",
       x = "Age",
       y = "Count") +
  theme_minimal()

# ------------------------------------------------
# Description of 'coded_country'
# ------------------------------------------------

# Identify rows where 'Country' is NA or an empty string
missing_country <- wvs_data[is.na(wvs_data$Country) | wvs_data$Country == "", ]
nrow(missing_country)

# Convert Country column to a factor
wvs_data$Country <- as.factor(wvs_data$Country)

# Compute the frequency table for coded_country in descending order
country_counts <- sort(table(wvs_data$Country), decreasing = TRUE)
country_counts

# ------------------------------------------------
# Draw the plot of Distribution of Age
# ------------------------------------------------
# Calculate country counts
country_counts <- wvs_data %>%
  group_by(Country) %>%
  summarise(count = n())

# Determine min, max, average, and the count for JOR
max_count <- max(country_counts$count)

min_count <- min(country_counts$count)
avg_count <- mean(country_counts$count)
jor_count <- country_counts$count[country_counts$Country == "JOR"]

# Create the bar chart
p_country <- ggplot(country_counts, aes(x = reorder(Country, -count), y = count)) +
  geom_col(fill = "darkslategray1", color = "black") +

  # Add a horizontal dashed line for the average count
  geom_hline(yintercept = avg_count, linetype = "dashed", color = "red") +

  # Annotate the average count on the plot
  annotate("text",
           x = nrow(country_counts) / 2,  # roughly center the text horizontally
           y = avg_count,
           label = paste0("Avg: ", round(avg_count, 1)),
```

```
        vjust = -0.5,
        color = "red") +

  # Label the bar with the maximum count
  geom_text(
    data = filter(country_counts, count == max_count),
    aes(label = paste0(count)),
    vjust = -0.5,
    color = "blue",
    size = 3
  ) +

  # Label the bar with the minimum count
  geom_text(
    data = filter(country_counts, count == min_count),
    aes(label = paste0(count)),
    vjust = -0.5,
    color = "purple",
    size = 3
  ) +

  # Annotate the count for the focus country JOR
  geom_text(
    data = filter(country_counts, Country == "JOR"),
    aes(label = paste0("JOR: ", count)),
    vjust = -0.5,
    color = "darkgreen",
    size = 3
  ) +

  ggtitle("Distribution of Countries") +
  xlab("Country") +
  ylab("Number of Respondents") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        plot.title = element_text(hjust = 0.5))  # Center the title

# Print the plot
print(p_country)

# ------------------------------------------------
# multivariate scatter plot (interaction between Age,Edu,HSatLife and Gender (MF))
# ------------------------------------------------

wvs_data$MF <- factor(wvs_data$MF,
                      levels = c(1, 2),
                      labels = c("Male", "Female"))

wvs_sample$Edu <- factor(wvs_sample$Edu)
```

```
wvs_data$HSatLife <- ifelse(wvs_data$HSatLife %in% c(-1, -2, -3, -4, -5), NA,
wvs_data$HSatLife)

# Filter to remove NA values
wvs_clean <- wvs_data %>%
  filter(!is.na(MF), !is.na(Age), !is.na(HSatLife))

# Sample 1000 rows
set.seed(123)
wvs_sample <- wvs_clean[sample(nrow(wvs_clean), 1000), ]

ggplot(wvs_sample, aes(x = Age, y = HSatLife, color = MF)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "loess", se = FALSE, size = 1.2) +
  labs(title = "Life Satisfaction by Age and Gender")


ggplot(wvs_sample, aes(x = Age, y = HSatLife)) +
  geom_point(alpha = 0.5, size = 2, color = "steelblue") +
  geom_smooth(method = "loess", se = TRUE, color = "darkblue", size = 1) +
  facet_wrap(~ MF) +
  labs(
    title = "Life Satisfaction by Age (Faceted by Gender)",
    x = "Age",
    y = "Life Satisfaction (1–10)"
  ) +
  theme_minimal(base_size = 13) +
  theme(
    plot.title = element_text(face = "bold"),
    plot.subtitle = element_text(color = "gray30"),
    strip.text = element_text(face = "bold"),
    legend.position = "none"
  )
```

Q2(a).

```
# ------------------------------------------------
# Q2.Focus country vs all other countries as a group.
# ------------------------------------------------
# 2(a).

# Read the extracted data set
wvs_data <- read.csv("WVSExtract_34035958.csv", stringsAsFactors = FALSE)
# Create a new categorical column 'Focus' to distinguish the focus country (JOR)
# Assign "JOR" to rows where the country is Jordan, and "Others_Country" otherwise
wvs_data <- wvs_data %>%
  mutate(Focus = recode(Country,
                        "JOR" = "JOR",
                        .default = "Others_Country") %>%
```

```
        factor(levels = c("JOR", "Others_Country")))

# ------------------------------------------------
# Means Comparison of Participant Responses using JOR and Others_Country using graph
# ------------------------------------------------
predictor_vars <- c(
  "TPeople", "TFamily", "TNeighbourhood", "TKnow", "TMeet",
  "VFamily", "VFriends",
  "HSatLife", "HSatFin", "HFood", "HMedicine",
  "EEquality", "ECompetition", "SSecure", "SJob",
  "PDemImp", "PDemCurrent", "PSatisfied",
  "STFaith", "STRight", "STWorld","PMobile",
  "PEmail", "PFriends","Age", "Edu"
)

# Summarize means by group
basic_summary <- wvs_data %>%
  select(Focus, all_of(predictor_vars)) %>%
  group_by(Focus) %>%
  summarise(across(everything(), ~ mean(.x, na.rm = TRUE)))

summary_long <- basic_summary %>%
  pivot_longer(-Focus, names_to = "Variable", values_to = "Mean")

ggplot(summary_long, aes(x = reorder(Variable, -Mean), y = Mean, fill = Focus)) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.9)) +
  geom_text(aes(label = round(Mean, 2)),
            position = position_dodge(width = 0.9),
            vjust = 0.3, hjust = -0.1, size = 3) +
  labs(
    title = "Mean Comparison of Participant Responses(Jordan vs Other Countries)",
    x = "Variable",
    y = "Mean Score"
  ) +
  scale_fill_manual(values = c("JOR" = "#4E5B6E", "Others_Country" = "#E6B800")) +
  coord_flip(clip = "off") +
  theme_minimal() +
  theme(plot.margin = margin(5.5, 30, 5.5, 5.5))


# ------------------------------------------------
# Means Comparison of Participant Responses using JOR and Others_Country using boxplot
# ------------------------------------------------
# Clean negative values in predictors
wvs_data[predictor_vars] <- lapply(wvs_data[predictor_vars], function(x) {
  x[x < 0] <- NA
  return(x)
})

wvs_long <- wvs_data %>%
  select(Focus, all_of(predictor_vars)) %>%
```

```
  pivot_longer(cols = -Focus, names_to = "Attribute", values_to = "Value")

ggplot(wvs_long, aes(x = Focus, y = Value, fill = Focus)) +
  geom_boxplot() +
  facet_wrap(~Attribute, scales = "free_y") +
  scale_fill_manual(values = c("JOR" = "#A7BED3", "Others_Country" = "#FFDAC1")) +
  theme_minimal() +
  labs(
    title = "Comparison of Participant Responses(Jordan vs Other Countries)",
    x = "Country Group",
    y = "Value"
  )

# ------------------------------------------------
# Boxplot for TMeet, TPeople, HFood, HMedicine, and PDemImp
# ------------------------------------------------

# Define the selected variables for plotting
selected_vars <- c("TMeet", "TPeople", "HFood", "HMedicine", "PDemImp")

# Clean negative values for selected variables
wvs_data[selected_vars] <- lapply(wvs_data[selected_vars], function(x) {
  x[x < 0] <- NA
  return(x)
})

# Convert to long format for ggplot
wvs_long_selected <- wvs_data %>%
  select(Focus, all_of(selected_vars)) %>%
  pivot_longer(cols = -Focus, names_to = "Attribute", values_to = "Value")

# Create boxplot
ggplot(wvs_long_selected, aes(x = Focus, y = Value, fill = Focus)) +
  geom_boxplot() +
  facet_wrap(~Attribute, scales = "free_y") +
  scale_fill_manual(values = c("JOR" = "#A7BED3", "Others_Country" = "#FFDAC1")) +
  theme_minimal() +
  labs(
    title = "Boxplot of TMeet, TPeople, HFood, HMedicine, PDemImp (Jordan vs Other Countries)",
    x = "Country Group",
    y = "Response Value"
  )

# ------------------------------------------------
# Boxplot for PMobile and SJob
# ------------------------------------------------

# Define the selected variables for plotting
selected_vars <- c("PMobile","SJob")
```

```
# Clean negative values for selected variables
wvs_data[selected_vars] <- lapply(wvs_data[selected_vars], function(x) {
 x[x < 0] <- NA
 return(x)
})

# Convert to long format for ggplot
wvs_long_selected <- wvs_data %>%
 select(Focus, all_of(selected_vars)) %>%
 pivot_longer(cols = -Focus, names_to = "Attribute", values_to = "Value")

# Create boxplot
ggplot(wvs_long_selected, aes(x = Focus, y = Value, fill = Focus)) +
 geom_boxplot() +
 facet_wrap(~Attribute, scales = "free_y") +
 scale_fill_manual(values = c("JOR" = "#A7BED3", "Others_Country" = "#FFDAC1")) +
 theme_minimal() +
 labs(
  title = "Boxplot of PMobile, SJob (Jordan vs Other Countries)",
  x = "Country Group",
  y = "Response Value"
 )

# ------------------------------------------------
# Boxplot for TFamily and VFamily
# ------------------------------------------------

# Define the selected variables for plotting
selected_vars <- c("TFamily","VFamily")

# Clean negative values for selected variables
wvs_data[selected_vars] <- lapply(wvs_data[selected_vars], function(x) {
 x[x < 0] <- NA
 return(x)
})

# Convert to long format for ggplot
wvs_long_selected <- wvs_data %>%
 select(Focus, all_of(selected_vars)) %>%
 pivot_longer(cols = -Focus, names_to = "Attribute", values_to = "Value")

# Create boxplot
ggplot(wvs_long_selected, aes(x = Focus, y = Value, fill = Focus)) +
 geom_boxplot() +
 facet_wrap(~Attribute, scales = "free_y") +
 scale_fill_manual(values = c("JOR" = "#A7BED3", "Others_Country" = "#FFDAC1")) +
 theme_minimal() +
 labs(
  title = "Boxplot of TFamily, VFamily (Jordan vs Other Countries)",
```

```
   x = "Country Group",
   y = "Response Value"
  )

# ------------------------------------------------
# Comparison of Participant Responses using T-test and present as a graph
# ------------------------------------------------
t_test_results <- lapply(predictor_vars, function(var) {
  formula <- as.formula(paste(var, "~ Focus"))
  test <- t.test(formula, data = wvs_data)

  data.frame(
    Variable = var,
    Mean_JOR = mean(wvs_data[[var]][wvs_data$Focus == "JOR"], na.rm = TRUE),
    Mean_Others = mean(wvs_data[[var]][wvs_data$Focus == "Others_Country"], na.rm = TRUE),
    p_value = test$p.value,
    Significant = ifelse(test$p.value < 0.05, "Yes", "No")
  )
})

# Combine into one data frame
t_test_summary <- do.call(rbind, t_test_results)

ggplot(t_test_summary, aes(x = reorder(Variable, -Mean_JOR))) +
  geom_bar(aes(y = Mean_JOR, fill = "Jordan"), stat = "identity", alpha = 0.7, width = 0.4, position
= position_nudge(x = -0.2)) +
  geom_bar(aes(y = Mean_Others, fill = "Others Countries"), stat = "identity", alpha = 0.7, width =
0.4, position = position_nudge(x = 0.2)) +
  geom_text(aes(y = pmax(Mean_JOR, Mean_Others) + 0.1, label = ifelse(Significant == "Yes", "*",
"")), size = 5) +
  scale_fill_manual(values = c("Jordan" = "#4E79A7", "Others Countries" = "#F28E2B")) +
  coord_flip() +
  labs(
    title = "Comparison of Participant Responses using T-Test (Jordan vs Other Countries)",
    x = "Variable", y = "Mean Score",
    fill = "Group"
  ) +
  theme_minimal()

print(t_test_summary)


# ------------------------------------------------
# Convert categorical nominal to factor
# ------------------------------------------------

# Convert numeric categorical variables into properly labeled factors
# to allow R to treat them as categorical and create dummy variables in regression
wvs_data <- wvs_data %>%
  mutate(
    PIAB = factor(PIAB, levels = 1:4,
```

```
          labels = c("Economic Growth", "Defense", "Jobs and Communities", "Beauty")),

  Employment = factor(Employment, levels = 1:8,
                      labels = c("Full time", "Part time", "Self employed", "Retired",
                                 "Spouse/not employed", "Student", "Unemployed", "Other")),

  MF = factor(MF, levels = 1:2, labels = c("Male", "Female"))
 )

# ---------------------------
# Comparison of Categorical Nominal Variables Between Jordan and Other Countries using Mode
# ---------------------------
# Function to get mode
get_mode <- function(x) {
 ux <- na.omit(unique(x))
 ux[which.max(tabulate(match(x, ux)))]
}

nominal_vars <- c("PIAB", "Employment", "MF")

# Recompute the mode for each nominal variable by group (after labeling)
mode_table <- wvs_data %>%
 group_by(Focus) %>%
 summarise(across(all_of(nominal_vars), get_mode))

mode_table

# ------------------------------------------------
# Plot the graph of distribution of categorical nominal variables
# ------------------------------------------------
nominal_long <- wvs_data %>%
 select(Focus, all_of(nominal_vars)) %>%
 pivot_longer(cols = -Focus, names_to = "Variable", values_to = "Response") %>%
 filter(!is.na(Response)) %>%
 group_by(Focus, Variable, Response) %>%
 summarise(Count = n(), .groups = "drop") %>%
 group_by(Variable, Focus) %>%
 mutate(Percent = Count / sum(Count) * 100)

# Plot a faceted bar chart with side-by-side percentages
ggplot(nominal_long, aes(x = Response, y = Percent, fill = Focus)) +
 geom_bar(stat = "identity", position = "dodge") +
 facet_wrap(~Variable, scales = "free_x") +
 scale_fill_manual(values = c("JOR" = "#4E5B6E", "Others_Country" = "#E6B800")) +
 labs(
  title = "Distribution of Categorical Nominal Variables (Jordan vs Other Countries)",
  x = "Response Category",
  y = "Percentage of Respondents"
 ) +
 theme_minimal() +
```

```
 theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

Q2(b).

```
# ------------------------------------------------
# Q2B. Linear and Stepwise Regression for Jordan (JOR)
# ------------------------------------------------
wvs_data <- read.csv("WVSExtract_34035958.csv", stringsAsFactors = FALSE)

wvs_data <- wvs_data %>%
 mutate(Focus = recode(Country,
                "JOR" = "JOR",
                .default = "Others_Country") %>%
        factor(levels = c("JOR", "Others_Country")))

# Filter dataset for the focus country: Jordan (JOR)
wvs_jordan <- wvs_data %>% filter(Focus == "JOR")

# Convert categorical variables with numeric codes into factors with labels
wvs_jordan <- wvs_jordan %>%
 mutate(
  PIAB = factor(PIAB, levels = 1:4,
           labels = c("Economic Growth", "Defense", "Jobs and Communities", "Beauty")),

  Employment = factor(Employment, levels = 1:8,
                labels = c("Full time", "Part time", "Self employed", "Retired",
                       "Spouse/not employed", "Student", "Unemployed", "Other")),

  MF = factor(MF, levels = 1:2, labels = c("Male", "Female"))
 )

# Remove the redundant Country column
wvs_jor <- wvs_jordan %>% select(-Country)

# ------------------------------------------------
# 2. Define response variables and predictors
# ------------------------------------------------
# List of all confidence variables (dependent variables)
target_vars <- c("CUniversities", "CArmedForces", "CPress", "CTelevision",
           "CUnions", "CCourts", "CPParties", "CCivilService",
           "CMajCompanies", "CEnvOrg")

# All independent variables (predictors)
predictors <- paste(c("TPeople", "TFamily", "TNeighbourhood", "TKnow", "TMeet",
                "VFamily", "VFriends",
                "HSatLife", "HSatFin", "HFood", "HMedicine",
                "EEquality", "ECompetition", "SSecure", "SJob",
                "PIAB", "STFaith", "STRight", "STWorld",
                "PMobile", "PEmail", "PFriends",
                "PDemImp", "PDemCurrent", "PSatisfied",
                "MF", "Age", "Edu", "Employment"), collapse = " + ")
```

```
# ------------------------------------------------
# 3. Fit linear and stepwise regression models
# ------------------------------------------------
# Initialize lists to store models
lm_models <- list()
stepwise_models <- list()

# Loop through each dependent variable and fit models
for (var in target_vars) {
  # Create formula dynamically
  f <- as.formula(paste(var, "~", predictors))

  # Fit full linear model
  model <- lm(f, data = na.omit(wvs_jor))
  lm_models[[var]] <- model

  # Fit stepwise model
  step_model <- step(model, direction = "both", trace = 0)
  stepwise_models[[var]] <- step_model
}

# ------------------------------------------------
# 4. Extract R-squared values for model comparison
# ------------------------------------------------
# Function to get R-squared
get_r_squared <- function(model) summary(model)$r.squared

# Function to extract significant predictors
get_significant_predictors <- function(model) {
  sig <- summary(model)$coefficients
  sig_vars <- rownames(sig)[sig[, 4] < 0.05]
  sig_vars[sig_vars != "(Intercept)"]
}

# R-squared tables
r_squared_df <- data.frame(
  CVariable = names(lm_models),
  R_squared = sapply(lm_models, get_r_squared)
)

# Data frame of R-squared values for stepwise models
stepwise_r_squared_df <- data.frame(
  CVariable = names(stepwise_models),
  R_squared = sapply(stepwise_models, get_r_squared)
)

# Predictor lists
lm_predictors_df <- data.frame(
  CVariable = names(lm_models),
  Best_Predictors = sapply(lm_models, function(m) paste(get_significant_predictors(m),
collapse = ", "))
```

```
)

# Data frame of R-squared values for stepwise models
stepwise_predictors_df <- data.frame(
  CVariable = names(stepwise_models),
  Best_Predictors = sapply(stepwise_models, function(m) paste(get_significant_predictors(m),
collapse = ", "))
)
# ------------------------------------------------
# 6. Count most common predictors across all models
# ------------------------------------------------
lm_top_predictors <- sort(table(unlist(sapply(lm_models, get_significant_predictors))),
decreasing = TRUE)
step_top_predictors <- sort(table(unlist(sapply(stepwise_models, get_significant_predictors))),
decreasing = TRUE)


# ------------------------------------------------
# 7. Display Results
# ------------------------------------------------
cat("----- Linear Regression R-squared Values (JOR) -----\n")
print(r_squared_df)

cat("----- Stepwise Regression R-squared Values (JOR) -----\n")
print(stepwise_r_squared_df)

cat("----- Significant Predictors from Linear Models (JOR) -----\n")
print(lm_predictors_df)

cat("----- Significant Predictors from Stepwise Model (JOR)s -----\n")
print(stepwise_predictors_df)

cat("----- Top 5 Predictors from Linear Models (JOR) -----\n")
print(head(lm_top_predictors, 5))

cat("----- Bottom 3 Predictors from Linear Models (JOR) -----\n")
print(tail(lm_top_predictors, 3))

cat("----- Top 5 Predictors from Stepwise Models (JOR) -----\n")
print(head(step_top_predictors, 5))

cat("----- Bottom 3 Predictors from Stepwise Models (JOR) -----\n")
print(tail(step_top_predictors, 3))

# -------------------------------
# correlation
# -------------------------------
# 1. Select only numeric columns
numeric_data <- wvs_jordan[, sapply(wvs_jordan, is.numeric)]

# 2. Compute correlation matrix
```

```
cor_matrix <- cor(numeric_data, use = "complete.obs")

# 3. Extract correlations between confidence variables (C*) and predictors
cor_confidence <- cor_matrix[grepl("^C", rownames(cor_matrix)), !grepl("^C",
colnames(cor_matrix))]

# 4. Plot the heatmap with values included
corrplot(cor_confidence,
      method = "color",
      col = colorRampPalette(c("red", "white", "blue"))(200),
      tl.col = "black", tl.srt = 45,
      addCoef.col = "black",
      number.cex = 0.6,      # Text size of the numbers
      title = "Correlation between Predictors and Confidence Variables (JOR)",
      mar = c(0, 0, 1, 0),
      cl.pos = "r")
```

Q2(c).

```
# ------------------------------
# Q2C. Linear and Stepwise Regression for Others_Country
# ------------------------------

wvs_data <- read.csv("WVSExtract_34035958.csv", stringsAsFactors = FALSE)

wvs_data <- wvs_data %>%
 mutate(Focus = recode(Country,
                "JOR" = "JOR",
                .default = "Others_Country") %>%
       factor(levels = c("JOR", "Others_Country")))


# Step 1: Filter dataset for other countries
wvs_others <- wvs_data %>% filter(Focus == "Others_Country")

# Convert numeric categorical variables into properly labeled factors
wvs_others <- wvs_others %>%
 mutate(
  PIAB = factor(PIAB, levels = 1:4,
           labels = c("Economic Growth", "Defense", "Jobs and Communities", "Beauty")),

  Employment = factor(Employment, levels = 1:8,
                labels = c("Full time", "Part time", "Self employed", "Retired",
                      "Spouse/not employed", "Student", "Unemployed", "Other")),


  MF = factor(MF, levels = 1:2, labels = c("Male", "Female"))
 )

# Drop the 'Country' column since we group all the countries into Others_Country
wvs_other <- wvs_others %>% select(-Country)
```

```
str(wvs_others)

# ------------------------------------------------
# 2. Define response variables and predictors
# ------------------------------------------------
target_vars <- c("CUniversities", "CArmedForces", "CPress", "CTelevision",
            "CUnions", "CCourts", "CPParties", "CCivilService",
            "CMajCompanies", "CEnvOrg")

# Shared predictors
predictors <- paste(c("TPeople", "TFamily", "TNeighbourhood", "TKnow", "TMeet",
                "VFamily", "VFriends",
                "HSatLife", "HSatFin", "HFood", "HMedicine",
                "EEquality", "ECompetition", "SSecure", "SJob",
                "PIAB", "STFaith", "STRight", "STWorld",
                "PMobile", "PEmail", "PFriends",
                "PDemImp", "PDemCurrent", "PSatisfied",
                "MF", "Age", "Edu", "Employment"), collapse = " + ")

# ------------------------------------------------
# 3. Fit linear and stepwise regression models
# ------------------------------------------------
# Initialize model storage
lm_models_others <- list()
stepwise_models_others <- list()

# Build full and stepwise models
for (var in target_vars) {
 formula_str <- as.formula(paste(var, "~", predictors))

 model <- lm(formula_str, data = na.omit(wvs_others))
 step_model <- step(model, direction = "both", trace = 0)

 lm_models_others[[var]] <- model
 stepwise_models_others[[var]] <- step_model
}

# ------------------------------------------------
# 4. Extract R-squared values for model comparison
# ------------------------------------------------

# R-squared
get_r_squared <- function(model) summary(model)$r.squared

r_squared_df <- data.frame(
 CVariable = names(lm_models_others),
 R_squared = sapply(lm_models_others, get_r_squared)
)
cat("----- Linear Regression R-squared Values (Others Country) -----\n")
print(r_squared_df)
```

```
stepwise_r_squared_df <- data.frame(
  CVariable = names(stepwise_models_others),
  R_squared = sapply(stepwise_models_others, get_r_squared)
)
cat("----- Stepwise Regression R-squared Values (Others Country) -----\n")
print(stepwise_r_squared_df)

# ------------------------------------------------
# 5. Extract significant predictors (p < 0.05)
# ------------------------------------------------
get_significant_predictors <- function(model) {
  coef_summary <- summary(model)$coefficients
  sig_vars <- rownames(coef_summary)[coef_summary[, 4] < 0.05]
  sig_vars[sig_vars != "(Intercept)"]
}

lm_predictors_df <- data.frame(
  CVariable = names(lm_models_others),
  Best_Predictors = sapply(lm_models_others, function(m) paste(get_significant_predictors(m),
collapse = ", "))
)

stepwise_predictors_df <- data.frame(
  CVariable = names(stepwise_models_others),
  Best_Predictors = sapply(stepwise_models_others, function(m)
paste(get_significant_predictors(m), collapse = ", "))
)


# ------------------------------------------------
# 6. Count frequency of predictors across all models
# ------------------------------------------------

lm_top_predictors <- sort(table(unlist(sapply(lm_models_others, get_significant_predictors))),
decreasing = TRUE)
cat("----- Top 5 Predictors from Linear Models (Others Country) -----\n")
print(head(lm_top_predictors, 5))

cat("----- Bottom 3 Predictors from Linear Models (Others Country) -----\n")
print(tail(lm_top_predictors, 3))

step_top_predictors <- sort(table(unlist(sapply(stepwise_models_others,
get_significant_predictors))), decreasing = TRUE)
cat("----- Top 5 Predictors from Stepwise Models (Others Country) -----\n")
print(head(step_top_predictors, 5))

cat("----- Bottom 3 Predictors from Stepwise Models (Others Country) -----\n")
print(tail(step_top_predictors, 3))

# ------------------------------
# correlation
```

```
# ------------------------------
# 1. Select only numeric columns
numeric_data <- wvs_other[, sapply(wvs_other, is.numeric)]

# 2. Compute correlation matrix
cor_matrix <- cor(numeric_data, use = "complete.obs")

# 3. Extract correlations between confidence variables (C*) and predictors
cor_confidence <- cor_matrix[grepl("^C", rownames(cor_matrix)), !grepl("^C",
colnames(cor_matrix))]

# 4. Plot the heatmap with values included
corrplot(cor_confidence,
      method = "color",
      col = colorRampPalette(c("red", "white", "blue"))(200),
      tl.col = "black", tl.srt = 45,
      addCoef.col = "black",
      number.cex = 0.6,      # Text size of the numbers
      title = "Correlation between Predictors and Confidence Variables (Other Country)",
      mar = c(0, 0, 1, 0),
      cl.pos = "r")
```

Q3(a).

```
# ------------------------------------------------
# Q3.Focus country vs cluster of similar countries.
# ------------------------------------------------
# Q3(a).

# Read the dataset
external_country_data <- read.csv("country_profile_variables.csv")

# List of countries included in the WVS Country (in our assignment)
assignment_countries <- c("Andorra", "Argentina", "Armenia", "Australia", "Bangladesh", "Bolivia",
"Brazil",
                          "Canada", "Chile", "China", "Colombia", "Cyprus", "Czech Republic", "Germany",
                          "Ecuador", "Egypt", "Ethiopia", "Great Britain", "Greece", "Guatemala", "Hong Kong",
                          "Indonesia", "India", "Iran", "Iraq", "Jordan", "Japan", "Kazakhstan", "Kenya",
                          "Kyrgyzstan", "South Korea", "Lebanon", "Libya", "Macedonia", "Morocco", "Moldova",
                          "Mexico", "Myanmar", "Mongolia", "Malaysia", "Nigeria", "Nicaragua", "Netherlands",
                          "New Zealand", "Pakistan", "Peru", "Philippines", "Puerto Rico", "Romania", "Russia",
                          "Singapore", "Serbia", "Slovakia", "Thailand", "Tajikistan", "Tunisia", "Turkey",
                          "Taiwan", "Ukraine", "Uruguay", "United States", "Uzbekistan", "Venezuela", "Viet
Nam", "Zimbabwe")

# Filter the external dataset to keep only the list of countries included in the WVS Country (in our
assignment)
filtered_data <- external_country_data %>%
```

```
 filter(country %in% assignment_countries)



# Select relevant columns (update these column names as per the dataset)
selected_data <- filtered_data %>%
 select(
  Country = country,
  `Population_thousands_2017` = Population.in.thousands..2017.,
  `GDP_per_capita` = `GDP.per.capita..current.US..`,
  `Fertility_rate` = `Fertility.rate..total..live.births.per.woman.`,
  `Health_exp_pct_GDP` = `Health..Total.expenditure....of.GDP.`,
  `Unemployment_rate` = `Unemployment....of.labour.force.`,
  `Internet_users_per_100` = `Individuals.using.the.Internet..per.100.inhabitants.`,
  `Edu_primary_gross_enrol` = Education..Primary.gross.enrol..ratio..f.m.per.100.pop..
  )

# ------------------------------------------------
# data cleaning
# ------------------------------------------------

# Convert string like "102.3/101.7" to numeric average
selected_data$Edu_primary_gross_enrol <-
sapply(strsplit(as.character(selected_data$Edu_primary_gross_enrol), "/"), function(x) {
 mean(as.numeric(x))
})

# Find and print rows with at least one column containing -99
rows_with_99 <- selected_data[apply(selected_data, 1, function(row) any(row == -99, na.rm =
TRUE)), ]
print(rows_with_99)

# Convert all -99 values in numeric columns to NA
selected_data[selected_data == -99] <- NA

# Drop rows with any NA values
selected_data_clean <- na.omit(selected_data)

# Check for remaining NA values in each column
na_counts <- sapply(selected_data_clean[, -1], function(col) sum(is.na(col)))
print(na_counts)

# Replace "..." strings with NA
selected_data_clean[selected_data_clean == "..."] <- NA

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

# ------------------------------------------------
```

```
# data normalization
# ------------------------------------------------
# Exclude the "Country" column for normalization
numeric_data <- selected_data_clean[, -1]
# Convert all columns to numeric
numeric_data <- data.frame(lapply(numeric_data, function(x) as.numeric(as.character(x))))

# Normalize using scale() — each column becomes mean=0, sd=1
normalized_data <- as.data.frame(scale(numeric_data))

# Add country names back for reference
normalized_data$Country <- selected_data_clean$Country

# View the normalized data
print(normalized_data)

# ------------------------------------------------
# hirerchical clustering
# ------------------------------------------------

# Set seed for reproducibility
set.seed(34035958)

# Compute Euclidean distance matrix
distance_matrix <- dist(normalized_data, method = "euclidean")

# Perform hierarchical clustering using Ward's method
hc <- hclust(distance_matrix, method = "ward.D2")

# Plot the dendrogram
plot(hc, labels = selected_data_clean$Country, main = "Dendrogram of Countries", xlab = "", sub
= "", cex = 0.7)

# Cut the tree to get 5 clusters
clusters <- cutree(hc, k = 5)

# Add cluster rectangles
rect.hclust(hc, k = 5, border = "red")

# Add cluster labels to the cleaned data
selected_data_clean$Cluster <- clusters

# Sort the data by cluster number
selected_data_sorted <- selected_data_clean[order(selected_data_clean$Cluster), ]

# View countries by cluster
print(selected_data_sorted[, c("Country", "Cluster")])
```

Q3(b).

```
# ------------------------------------------------
# Q3B. Linear and Stepwise Regression for Cluster 4 (Similar Countries to JOR)
# ------------------------------------------------
wvs_data <- read.csv("WVSExtract_34035958.csv", stringsAsFactors = FALSE)

# Filter data for countries in cluster 4 with JOR
cluster4_countries <- c("EGY", "BOL", "KEN", "NGA", "PAK", "ETH", "ZWE")
cluster4_data <- wvs_data %>% filter(Country %in% cluster4_countries)

# Convert numeric categorical variables into properly labeled factors
# This allows R to treat them as categorical and create dummy variables in regression
cluster4_data <- cluster4_data %>%
 mutate(
  PIAB = factor(PIAB, levels = 1:4,
          labels = c("Economic Growth", "Defense", "Jobs and Communities", "Beauty")),
  Employment = factor(Employment, levels = 1:8,
                labels = c("Full time", "Part time", "Self employed", "Retired",
                       "Spouse/not employed", "Student", "Unemployed", "Other")),
  MF = factor(MF, levels = 1:2, labels = c("Male", "Female"))
 )
cluster4_data_clean <- cluster4_data %>% select(-Country)

# ------------------------------------------------
# 2. Define response variables and predictors
# ------------------------------------------------
# List of the target variables
outcomes <- c("CUniversities", "CArmedForces", "CPress", "CTelevision", "CUnions",
         "CCourts", "CPParties", "CCivilService", "CMajCompanies", "CEnvOrg")

# Independent variables
predictors <- c("TPeople", "TFamily", "TNeighbourhood", "TKnow", "TMeet",
           "VFamily", "VFriends",
           "HSatLife", "HSatFin", "HFood", "HMedicine",
           "EEquality", "ECompetition", "SSecure", "SJob",
           "PIAB", "STFaith", "STRight", "STWorld",
           "PMobile", "PEmail", "PFriends",
           "PDemImp", "PDemCurrent", "PSatisfied",
           "MF", "Age", "Edu", "Employment")

# Full lm models
lm_models_cluster4 <- list()
for (outcome in outcomes) {
 formula_text <- paste(outcome, "~", paste(predictors, collapse = " + "))
 lm_models_cluster4[[outcome]] <- lm(as.formula(formula_text), data =
na.omit(cluster4_data_clean))
}

# Stepwise models
step_models_cluster4 <- list()
for (outcome in outcomes) {
```

```
  step_models_cluster4[[outcome]] <- step(lm_models_cluster4[[outcome]], direction = "both",
trace = 0)
}

cat("----- Full Linear Model Summaries (Cluster 4) -----\n")
for (outcome in outcomes) {
  cat("\n---", outcome, "---\n")
  print(summary(lm_models_cluster4[[outcome]]))
}

cat("\n----- Stepwise Linear Model Summaries (Cluster 4) -----\n")
for (outcome in outcomes) {
  cat("\n---", outcome, "---\n")
  print(summary(step_models_cluster4[[outcome]]))
}

# -------------------------------
# Extract and Print R-squared
# -------------------------------

get_r_squared <- function(model) summary(model)$r.squared

r_squared_lm <- sapply(lm_models_cluster4, get_r_squared)
r_squared_step <- sapply(step_models_cluster4, get_r_squared)

r_squared_df_lm <- data.frame(
  CVariable = names(r_squared_lm),
  R_squared = unname(r_squared_lm),
  row.names = NULL
)

r_squared_df_step <- data.frame(
  CVariable = names(r_squared_step),
  R_squared = unname(r_squared_step),
  row.names = NULL
)

cat("----- Linear Regression R-squared Values (Cluster 4) -----\n")
print(r_squared_df_lm)

cat("----- Stepwise Regression R-squared Values (Cluster 4) -----\n")
print(r_squared_df_step)


# -------------------------------
# Extract Significant Predictors
# -------------------------------

get_significant_predictors <- function(model) {
  coef_summary <- summary(model)$coefficients
  sig_vars <- rownames(coef_summary)[coef_summary[, 4] < 0.05]
```

```
  sig_vars <- sig_vars[sig_vars != "(Intercept)"]
  paste(sig_vars, collapse = ", ")
}

# Full models
best_lm_predictors_cluster4 <- sapply(lm_models_cluster4, get_significant_predictors)
best_lm_df_cluster4 <- data.frame(
  CVariable = names(best_lm_predictors_cluster4),
  Best_Predictors = unname(best_lm_predictors_cluster4),
  row.names = NULL
)

# Stepwise models
best_step_predictors_cluster4 <- sapply(step_models_cluster4, get_significant_predictors)
best_step_df_cluster4 <- data.frame(
  CVariable = names(best_step_predictors_cluster4),
  Best_Predictors = unname(best_step_predictors_cluster4),
  row.names = NULL
)

print(best_lm_df_cluster4)
print(best_step_df_cluster4)
# ------------------------------
# Top 5 and Bottom 3 Predictors from Full Linear Models
# ------------------------------

all_lm_predictors_cluster4 <- unlist(strsplit(best_lm_df_cluster4$Best_Predictors, ",\\s*"))
top_lm_predictor_counts <- sort(table(all_lm_predictors_cluster4), decreasing = TRUE)

top_5_lm <- head(top_lm_predictor_counts, 5)
bottom_3_lm <- tail(top_lm_predictor_counts, 3)

cat("----- Top 5 Best Predictors from Linear Regression (Cluster 4) -----\n")
print(top_5_lm)
cat("----- Bottom 3 Least Frequent Predictors from Linear Regression (Cluster 4) -----\n")
print(bottom_3_lm)

# ------------------------------
# Top 5 and Bottom 3 Predictors from Stepwise Models
# ------------------------------

all_stepwise_predictors_cluster4 <- unlist(strsplit(best_step_df_cluster4$Best_Predictors,
",\\s*"))
top_stepwise_predictor_counts <- sort(table(all_stepwise_predictors_cluster4), decreasing =
TRUE)

top_5_step <- head(top_stepwise_predictor_counts, 5)
bottom_3_step <- tail(top_stepwise_predictor_counts, 3)

cat("----- Top 5 Best Predictors from Stepwise Regression (Cluster 4) -----\n")
print(top_5_step)
```

```
cat("----- Bottom 3 Least Frequent Predictors from Stepwise Regression (Cluster 4) -----\n")
print(bottom_3_step)
```

External sources for Q3 (After data cleaning and wrangling)

Kaggle. (n.d.). country_profile_variables.csv

	Country	Population_thousands_2017	GDP_per_capita	Fertility_rate	Health_exp_pct_GDP	Unemployment_rate	Internet_users_per_100	Edu_primary_gross_enrol
1	Andorra	77	39896.4	1.2	8.1	<NA>	13	NA
2	Argentina	44271	14564.5	2.3	4.8	6.5	256	110.00
3	Armenia	2930	3489.1	1.6	4.5	16.6	114	98.50
4	Australia	24451	51352.2	1.9	9.4	5.5	948	102.20
5	Bangladesh	164670	1207.9	2.2	2.8	4	151	120.55
6	Bolivia	11052	3076.8	3	6.3	3.8	231	97.10
7	Brazil	209288	8528.3	1.8	8.3	12.4	990	115.30
8	Canada	36624	43205.6	1.6	10.4	7.1	122	100.60
9	Chile	18055	13416.2	1.8	7.8	6.8	197	101.65
10	Hong Kong	7365	42431.0	1.2	NA	3.5	64	NA
11	China	1409517	8109.1	1.6	5.5	4.6	1080	104.15
12	Colombia	49066	6056.1	1.9	7.2	10.5	835	113.50
13	Cyprus	1180	21941.9	1.4	7.4	10.3	72	99.30
14	Czech Republic	10618	17561.7	1.5	7.4	3.9	53	99.75
15	Ecuador	16625	6205.1	2.6	9.2	5.8	2358	107.65
16	Egypt	97553	3452.3	3.4	5.6	11.5	156	103.95
17	Ethiopia	104957	602.8	4.6	4.9	5.7	148	102.05
18	Germany	82114	41686.2	1.4	11.3	4.2	116	104.95
19	Greece	11160	17788.0	1.3	8.1	23	374	97.60
20	Guatemala	16914	3903.5	3.2	6.2	2.4	290	101.80
21	India	1339180	1614.2	2.4	4.7	3.4	1052	108.95
22	Indonesia	263991	3346.5	2.4	2.8	5.8	1281	105.80
23	Iran	81163	5038.1	1.7	7.5	11.3	134	109.00
24	Iraq	38275	4509.0	4.6	5.5	16.1	72	NA
25	Japan	127484	34628.7	1.4	10.2	3	404	101.25
26	Jordan	9702	4940.1	3.6	7.5	13.4	113	97.35
27	Kazakhstan	18204	10312.1	2.7	4.4	5.6	82	108.95
28	Kenya	49700	1376.7	4.1	5.7	10.8	480	109.00
29	Kyrgyzstan	6045	1106.4	3.1	6.5	7.7	44	107.35
30	Lebanon	6082	8571.4	1.7	6.4	7	87	92.45
31	Libya	6375	5488.2	2.4	5.0	19.2	63	NA
32	Malaysia	31624	9768.4	2.1	4.2	3.3	1272	101.80
33	Mexico	129163	8980.9	2.3	6.3	4.1	1162	103.40
34	Mongolia	3076	3973.4	2.8	4.7	6.3	41	100.90
35	Morocco	35740	2919.3	2.6	5.9	10.4	207	114.65
36	Myanmar	53371	1161.5	2.3	2.3	0.8	321	99.65
37	Netherlands	17036	44332.1	1.7	10.9	5.6	40	104.70
38	New Zealand	4706	38294.3	2	11.0	5.5	199	99.35
39	Nicaragua	6218	2086.9	2.3	9.0	6.1	144	NA
40	Nigeria	190886	2714.5	5.7	3.7	5.4	361	93.65
40	Nigeria	190886	2714.5	5.7	3.7	5.4	361	93.65
41	Pakistan	197016	1410.4	3.7	2.6	5.9	140	92.45
42	Peru	32166	6069.1	2.5	5.5	5.3	685	101.70
43	Philippines	104918	2904.2	3	4.7	5.9	783	116.85
44	Puerto Rico	3663	27939.0	1.5	NA	12.8	126	89.75
45	South Korea	50982	27396.7	1.2	7.4	3.6	111	99.00
46	Moldova	4051	1591.4	1.3	10.3	5	35	92.40
47	Romania	19679	9120.7	1.5	5.6	7.1	104	89.75
48	Russia	143990	9243.3	1.7	7.1	5.8	235	100.50
49	Serbia	8791	5238.6	1.6	10.4	15.5	71	101.35
50	Singapore	5709	52239.0	1.2	4.9	2	293	NA
51	Slovakia	5448	16082.5	1.4	8.1	9.9	54	99.70
52	Tajikistan	8921	925.9	3.5	6.9	10.8	45	100.50
53	Thailand	69038	5814.8	1.5	4.1	0.6	611	102.65
54	Macedonia	2083	4836.1	1.5	6.5	27.3	110	93.20
55	Tunisia	11532	3660.9	2.3	7.0	14.6	96	114.15
56	Turkey	80745	9125.8	2.1	5.4	10.8	388	102.45
57	Ukraine	44223	2021.6	1.5	7.1	8.8	102	103.95
58	Great Britain	66182	44162.4	1.9	9.1	5	102	108.25
59	United States	324460	56053.8	1.9	17.1	4.9	1513	100.15
60	Uruguay	3457	15573.8	2	8.6	8.8	106	108.55
61	Uzbekistan	31911	2308.3	2.4	5.8	8.9	59	104.40
62	Venezuela	31977	11068.9	2.4	5.3	6.6	328	99.95
63	Viet Nam	95541	2067.9	2	7.1	2.2	616	108.85
64	Zimbabwe	16530	890.4	4	6.0	5	89	99.95