JuneJin, 31994695

June Jin - 31994695

2024-14-04

Introduction

My pdf is over 14~15 pages excluding appendix, but most pages have a large portion of code and graphs. It's close to 10 pages without large portion of data and code. Especially, in Question 3 - A, I tried to hide the output when i read the excel file, but I couldn't hide the output. Therefore, there are some wasted pages.

Question 1 - a

Overall Data

Upon examining the data in cvbase, it's evident that the dataframe consists of 40,000 rows and 52 columns. The data types include both integers and characters.

1. **Binary Input for Conditions:** Users input '1' for the conditions that apply to them and 'NA' for those that do not. For instance, in fields like "Employment status" or "Corona Proximity," where there are optional selections, users would input '1' for their applicable condition and 'NA' for others.
2. **Numeric Input within a Restricted Range:** Users input numeric values that appropriately represent their status or condition within a predefined range. For instance, if a feature allows values between 1 and 5, users would input a number that reflects their situation within that range.

For the chr data type, it is used in two main contexts:

1. **String Input for rankOrdLife:** String values are used to represent rankings, particularly in contexts like "rankOrdLife."
2. **String Input for Country Representation:** String values are used to represent countries. For example, users might input "South Korea" to represent South Korea or any other country name to denote their country.

Distribution of numerical attributes

Using `summary(cvbase)` , I could obtain the mean, median, maximum, and minimum values.

Variety of non-numerical (text) attributes

1. `rankOrdLife` ranges from 'A' to 'F', and there are also NA values present.
2. It is evident that there are a total of 110 different countries represented in the dataset. Furthermore, I could find there are some blank in the Country columns.

I could see there are lots of NA data in the original data(cvbase). Most NAs appear in employstatus 1 to 10, because the respondents were only choosing the items that fit them. Coronaclose also had a lot of NAs for the same reason.

In cvbase, missing values serve two purposes:

1. **No Need to Select:** This indicates that users don’t need to select a particular option because they have already chosen “1” in another section. For example, in the “employstatus” field, users need to choose only a few variable among 10 selections based on their employment status. Therefore, if they have already selected some option that is depends on their employment status, there should be some NA values to signify that the other options are not applicable.
2. **Truth of Missing Values:** These NA values represent truly missing data, indicating instances where the information is genuinely unavailable or not provided.

Question 1 - b

- Initially, in the cvbase dataset [1:10], where NA values are present, it is feasible to replace them with 0 without encountering any issues (code).
- However, for attributes like Isolation offline/offline, Loneliness, Life Satisfaction, etc., replacing NA with 0 could significantly impact the existing numerical values. Therefore, NA values can be replaced with the median to mitigate substantial effects on the original data.
- For attributes like Rank Order Life, where data is represented as characters (e.g., A, B, …, F) with NA values, it may be necessary to convert alphabetic characters to numeric equivalents.
- If NA values exist in the coded_country variable, it is advisable to exclude them during data analysis to avoid potential issues.

Question 2 - a

Since my focused country is Italy, I only extracted the coded_country from cvbase and made one data. After that, I filtered one data just because coded_country was not Italy. Now, I will compare those two data.

```
##
##  Welch Two Sample t-test
##
## data:  Q2_sample_for_italy and Q2_sample_not_italy
## t = 2.591, df = 67469, p-value = 0.009572
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.00613339 0.04423764
## sample estimates:
## mean of x mean of y
##  1.915912  1.890727
```

Explaining about the t-test

This t-test compares the association between the Italy data and non-Italy (others) data. Here, with a p-value of 0.009572, which is less than the significance level of 0.05, we can reject the null hypothesis. Rejecting the null hypothesis indicates that there is a statistically significant difference in the means between the two groups.

Therefore, we can conclude that there is a statistically significant difference between the means of the two groups, implying an association between Italy and non-Italy (others) data.

Comparing the data of biggest gap between Italy and Non-Italy (Others)

In the given Italian data, the proportion of students is relatively high at 0.2425, which shows a difference of about 0.0425 compared to the student proportion in the non-Italy group.

Moreover, the retirement rate stands at 0.1579, nearly double that of the non-Italy group, which is at 0.08.

Among regular workers, the proportion working over 40 hours per week is 0.1692. However, in non-Italy, this rate is significantly higher at 0.2753, indicating that Italy has a relatively lower proportion of people working over 40 hours compared to other countries.

The rate of volunteering in the past week is 0.0169, half of that observed in non-Italy. This suggests that volunteer activities in Italy are relatively limited.

Additionally, it was observed that in Italy, there are more people contacting others online than meeting them in person. While this trend is also evident in non-Italy, Italy has a higher number of online contacts.

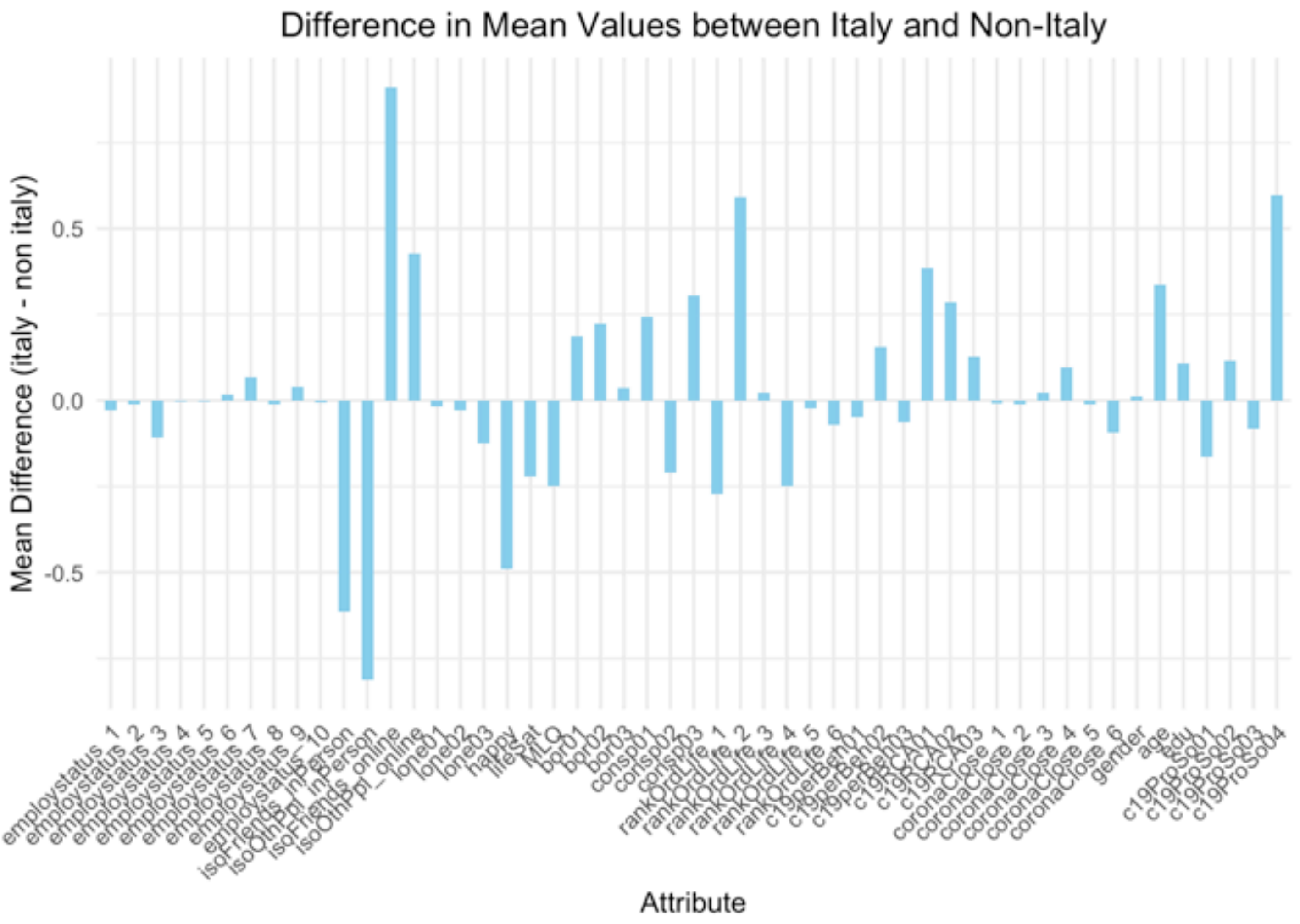
The average score for conspiracy among Italian citizens is notably high, showing a similar preference to the non-Italy group.

In Italy, the order of priorities in life is Victory - Beauty - Achievement - Empathy - Friendship - Love. This ranking is consistent with the priorities observed in non-Italy.

For finding the big difference between those two, I tried to make the graph, between italy and non-italy.

I could find the attribute which has the most biggest gap between italy and non-italy

```
comparison_plot
```



Areas where Italy exhibits higher values than Non-Italy:

1. `isoFriends_online` (Online Communication with Friends and Family):

Italians tend to have a higher frequency of online communication with friends and family, averaging approximately 5.924 contacts per week. This is higher than the average contact frequency of about 4.384 for citizens of other countries. This suggests that Italians are more open to maintaining social relationships and communication through digital means.

2. `c19ProSo04` (Willingness for Personal Sacrifice to Prevent COVID-19 Spread):

Italians demonstrate a greater willingness to make personal sacrifices to prevent the spread of the COVID-19 virus. With an average score of 1.865, Italians exhibit a higher willingness compared to the average score of 1.269 among citizens of other countries. This indicates that Italian society shows a tendency to be more proactive in making personal sacrifices for public safety.

3. `rankOrdLife` (Priority in Life):

Italians prioritize victory highly when evaluating various aspects of life. The average score of 4.7 assigned to the item representing victory (rank 3) indicates a significant emphasis on competition and success. Conversely, the average score of 2.11 assigned to the item representing love (rank 5) suggests a relatively lower importance placed on love. This reflects the cultural values in Italian society that emphasize competition and achievement.

4. `isoOthPpl_online` (Online Interaction with Others):

Italians engage in more frequent online interaction with others compared to citizens of other countries. With approximately 3.243 contacts per week, Italians have more online contacts than the average of about 2.816 contacts for citizens of other countries. This indicates that Italian society actively adopts digital platforms for communication and connection.

5. `c19RCA01` (Will to Receive Mandatory Vaccination):

Italians demonstrate a stronger support for mandatory vaccination once a vaccine is developed. With an average score of 1.6, Italians exhibit a higher willingness compared to the average score of 1.262 among citizens of other countries. This suggests that Italian society supports mandatory vaccination for disease prevention.

Areas where other countries exhibit higher values compared to Italy:

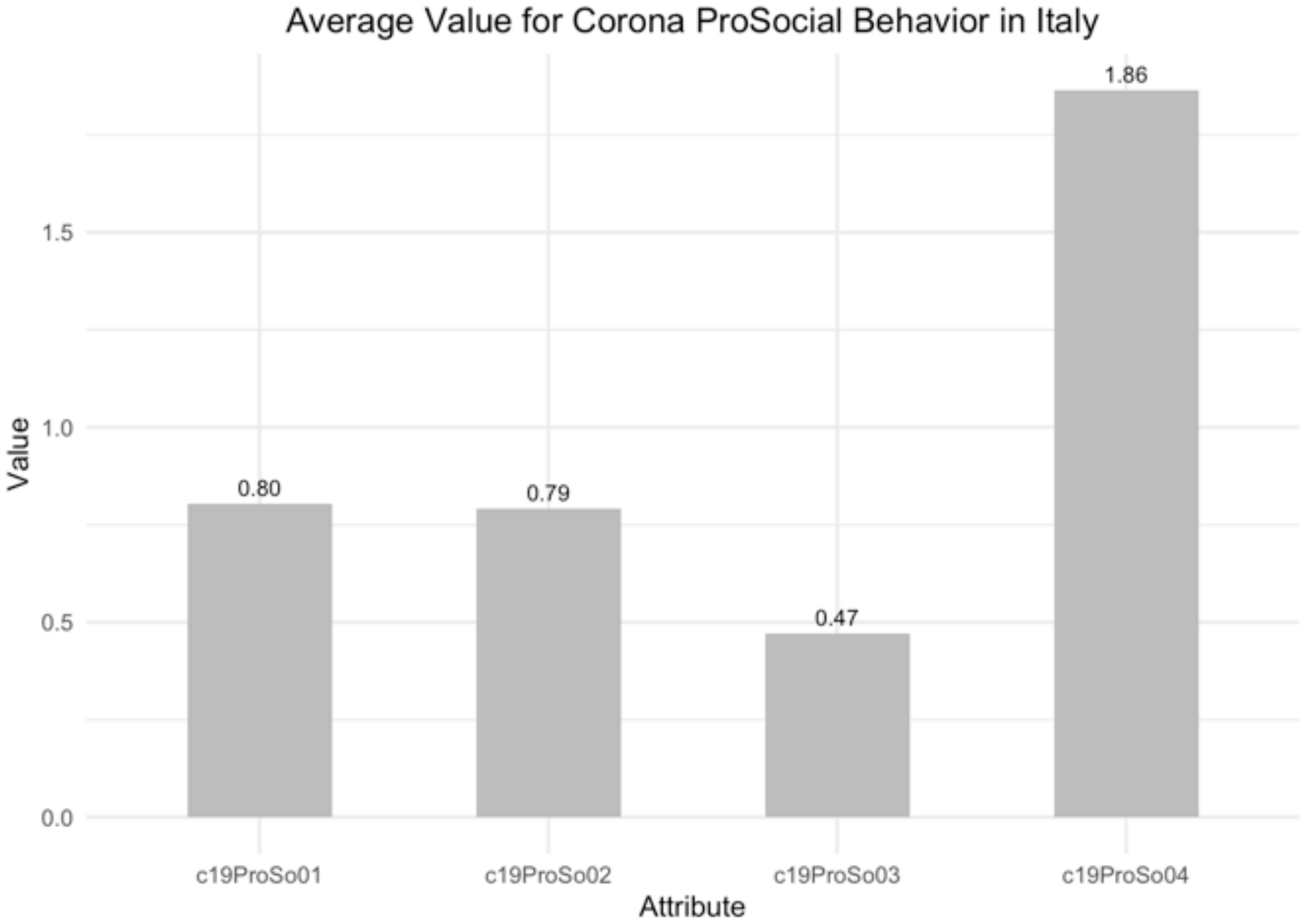
1. `isoOthPpl_inPerson` (Face-to-Face Interaction with Others):

Citizens of other countries engage in more face-to-face contact with others compared to Italians. Italians have approximately 1.151 contacts per week, whereas citizens of other countries have about 1.963 contacts per week. This indicates that citizens of other countries have a higher frequency of in-person communication with others compared to Italians.

2. `isoFriends_inPerson` (Face-to-Face Interaction with Friends and Family):

Italians have approximately 1.461 face-to-face contacts per week with friends and family, whereas citizens of other countries have about 2.075 contacts per week. This suggests that Italians have a lower frequency of face-to-face communication with friends and family compared to citizens of other countries.

Question 2 - b



In the results from Italy:

`c19ProSo01` : “I am willing to help others who suffer from coronavirus.” showed an average score close to “somewhat agree” at 0.8. Therefore, it can be observed that Italian citizens are somewhat willing to help those suffering from the coronavirus.

`c19ProSo02` : “I am willing to make donations to help others that suffer from coronavirus.” yielded an average score of 0.79, which is nearly identical to c19ProSo01. Thus, it appears that Italian citizens are inclined to donate to those affected by the coronavirus.

`c19ProSo03` : “I am willing to protect vulnerable groups from coronavirus even at my own expense.” produced an average score of 0.47, indicating a somewhat neutral stance. This part seems somewhat puzzling in its relation to c19ProSo02. While there’s a willingness to donate, there seems to be a lower score for actively protecting vulnerable groups, even at personal expense.

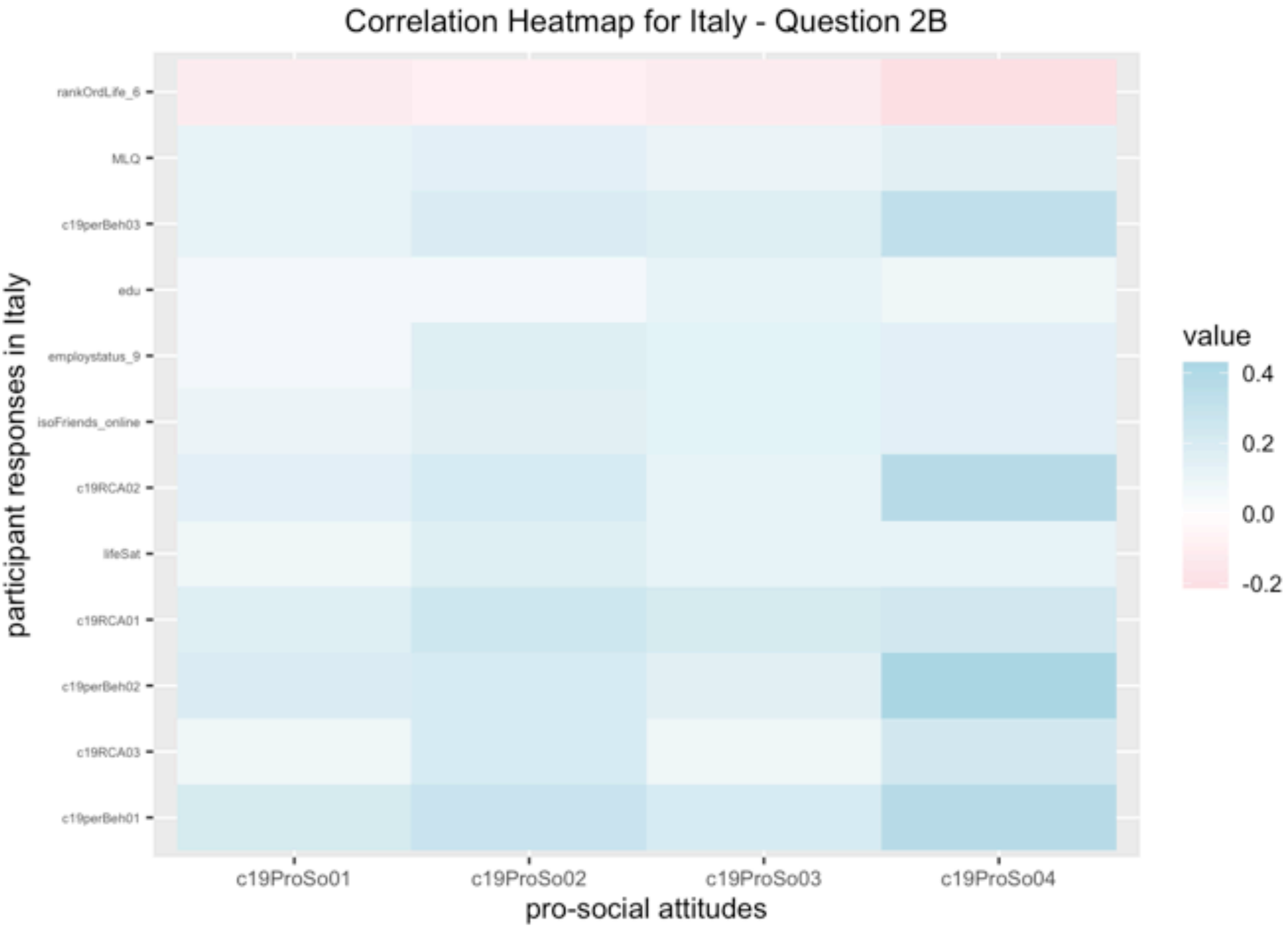
`c19ProSo04` : “I am willing to make personal sacrifices to prevent the spread of coronavirus.” recorded the highest score at 1.86, approaching “Agree”. Thus, it suggests that Italian citizens exhibit a positive attitude towards making personal sacrifices (such as taking the vaccine) to prevent the spread of coronavirus.

The processing of how to get the top attributes (maximum 20 attributes) from the Italy Dataset.

Compare correlation between prosocial attitude and responses, and will extract attributes that have biggest absolute number.

```
##                            Attribute      Value
## c19perBeh01             c19perBeh01  0.2169431
## c19RCA03                   c19RCA03  0.2124965
## c19perBeh02             c19perBeh02  0.1855465
## c19RCA01                   c19RCA01  0.1718641
## lifeSat                     lifeSat  0.1707562
## c19RCA02                   c19RCA02  0.1418563
## isoFriends_online isoFriends_online  0.1352253
## employstatus_9       employstatus_9  0.1339946
## edu                             edu  0.1266057
## c19perBeh03             c19perBeh03  0.1248271
## MLQ                             MLQ  0.1225972
## rankOrdLife_6         rankOrdLife_6 -0.1209064
```

This heatmap shows the correlation between the attributes that I assumed its related and prosocial attitudes of my selected couontry, Italy.

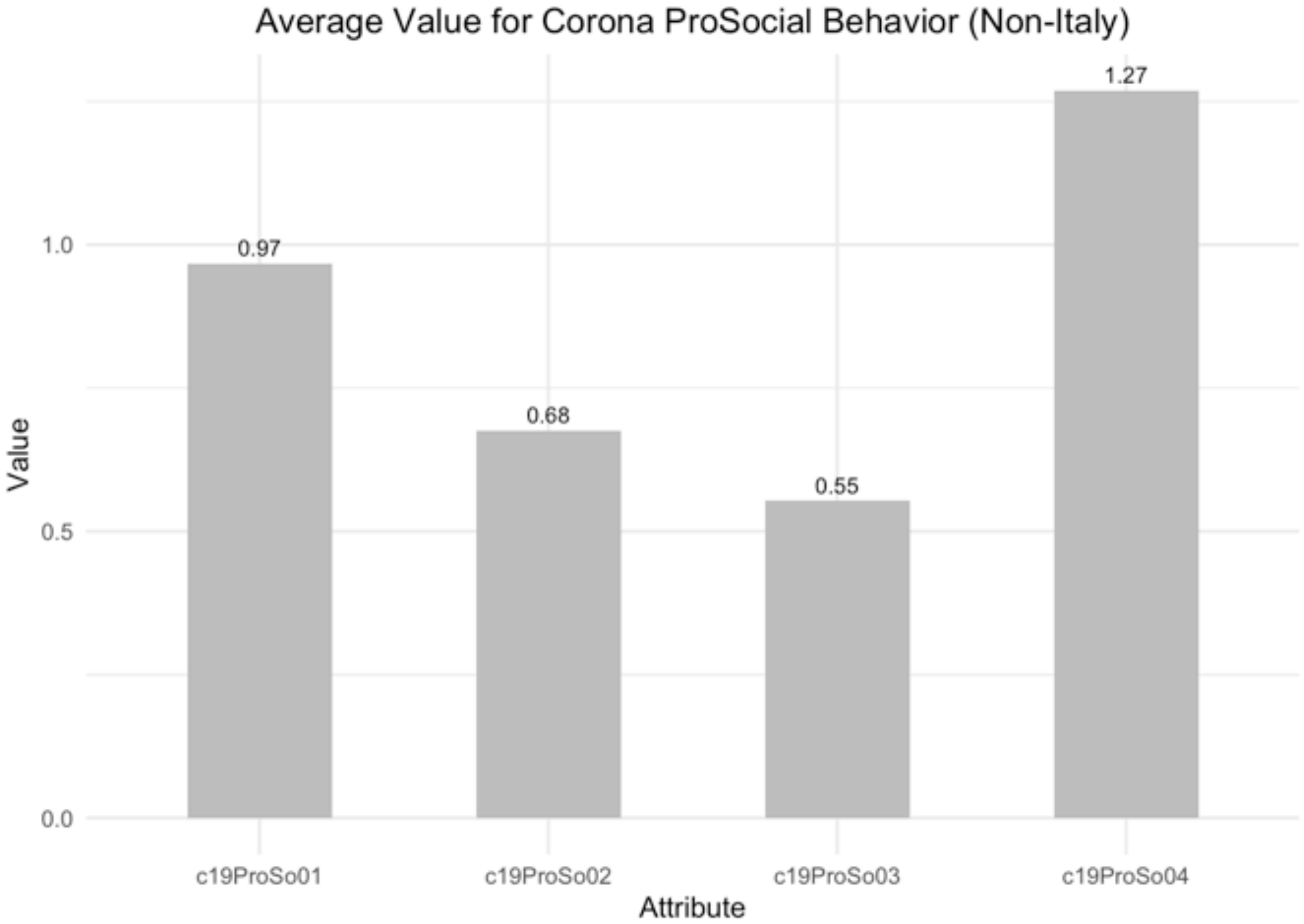


1. c19ProSo01 - Italy Attributes with strong relationships with `c19ProSo01` were found to be `c19ProSo03` , with a P-value smaller than 2e-16. Additionally, `c19ProSo02` and `c19ProSo04` showed values smaller than 0.001, indicating a significant influence as predictors. `MLQ` also exhibited a very small value with a P-value of 0.038. The Multiple R-squared score is 0.2622.

2. c19ProSo02 - Italy The attribute most strongly associated with c19ProSo02 was identified as `c19ProSo3` , with a P-value smaller than 2e-16, indicating it as the most powerful predictor. Moreover, `c19perBeh01` , `c19RCA03` , and `c19ProSo01` also showed significantly small P-values, indicating their strong attributes. The Multiple R-squared score is 0.471.
3. c19ProSo03 - Italy The attributes identified as the best predictors for c19ProSo03 were `c19ProSo01` and `c19ProSo02` , with P-values smaller than 2e-16, indicating them as the strongest attributes. `c19ProSo04` also exhibited a P-value of 2.45e-05, indicating it as a strong predictor. The Multiple R-squared score is 0.4879.
4. c19ProSo04 - Italy The attributes identified as the best predictors for c19ProSo04 were `c19perBeh02` , `c19RCA02` , `c19perBeh03` , `rankOrdLife_6` , `c19ProSo01` , and `c19ProSo03` . These exhibited sufficiently small P-values, and the Multiple R-squared score is 0.3363.

The reason for separately analyzing each attribute was due to the belief that the numerical values of Pro-Social Attributes would also affect themselves. Therefore, I divided them into four scenarios to consider all possibilities. Moreover, I do not believe that all 47 different attributes can be the strongest predictors. Hence, through correlation, I selected seven attributes with the largest absolute values, excluding duplicates, and optimized the number of attributes as much as possible, adding three Pro-Social Attributes for further optimization to find the best predictors.

Question 2 - c



In the results from non-Italy (others):

`c19ProSo01` : “I am willing to help others who suffer from coronavirus.” showed an average score close to “somewhat agree” at 0.8. Therefore, it can be observed that Non-Italian citizens are willing to help those suffering from the coronavirus.

`c19ProSo02` : “I am willing to make donations to help others that suffer from coronavirus.” yielded an average score of 0.79, which is nearly similar to `c19ProSo01` . Thus, it appears that Non-Italian citizens are inclined to donate to those affected by the coronavirus.

`c19ProSo03` : “I am willing to protect vulnerable groups from coronavirus even at my own expense.” produced an average score of 0.47, indicating a somewhat neutral stance. This part seems difficult to understand for me with relation to c19ProSo02.

While there’s a willingness to donate, there seems to be a lower score for actively protecting vulnerable groups, even at personal expense.

`c19ProSo04` : “I am willing to make personal sacrifices to prevent the spread of coronavirus.” recorded the highest score at 1.86, approaching “Agree”. Thus, it suggests that Non-Italian citizens exhibit a positive attitude towards making personal sacrifices (such as taking the vaccine) to prevent the spread of coronavirus.

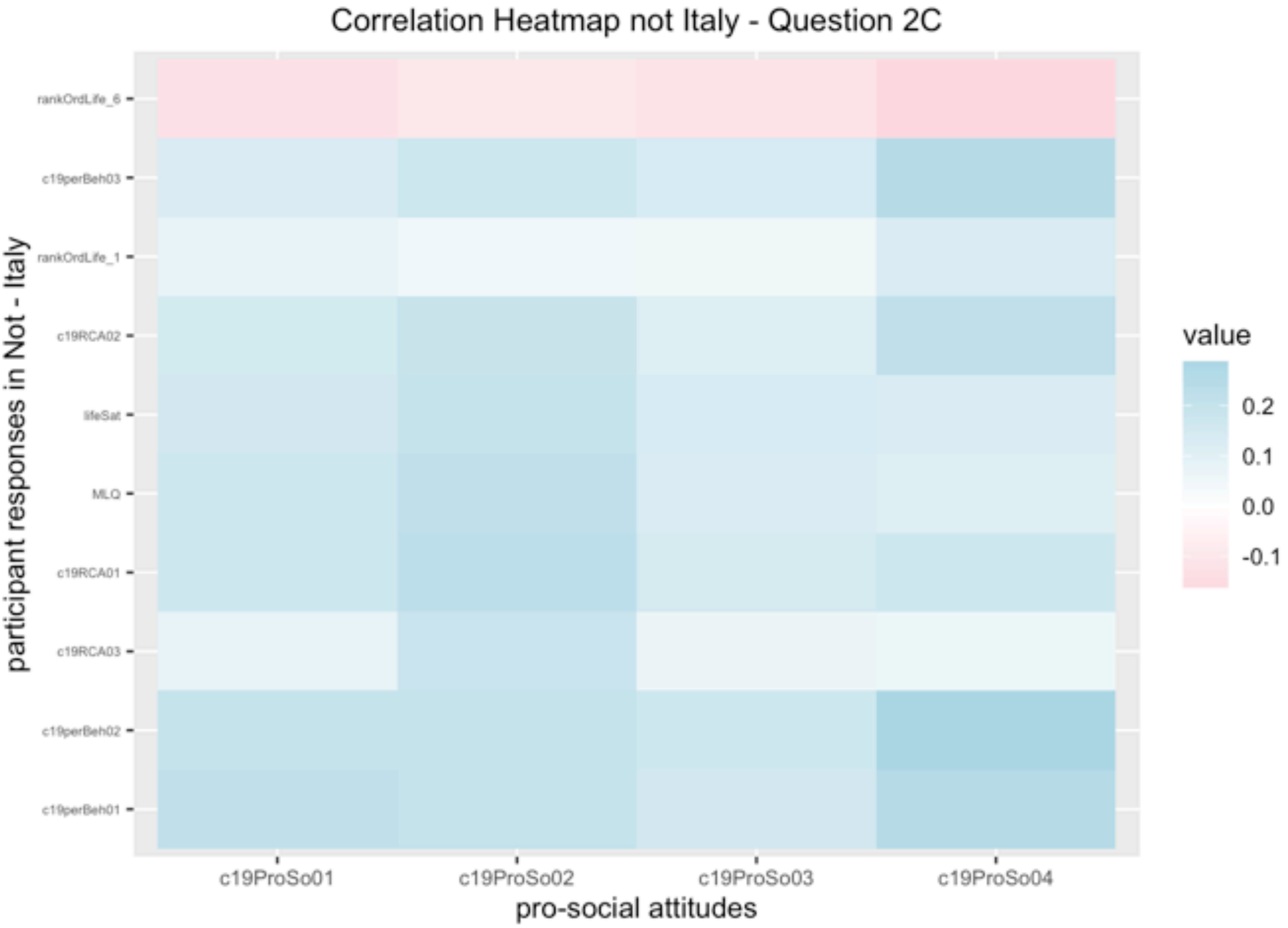
The processing of how to get the top attributes (maximum 20 attributes) from the Italy Dataset.

The progression is same with 2B.

```
##                     Attribute      Value
## c19perBeh01     c19perBeh01  0.2171493
## c19perBeh02     c19perBeh02  0.1972896
## c19RCA03           c19RCA03  0.1847939
## c19RCA01           c19RCA01  0.1651306
## MLQ                     MLQ  0.1645511
## lifeSat             lifeSat  0.1509148
## c19RCA02           c19RCA02  0.1491482
## rankOrdLife_1 rankOrdLife_1  0.1323238
## c19perBeh03     c19perBeh03  0.1288560
## rankOrdLife_6 rankOrdLife_6 -0.1224075
```

Therefore, we could get the optimized attributes for prediction.

This heatmap shows the correlation between the attributes that I assumed its related and prosocial attitudes of Non - Italy



1. c19ProSo01 - Non Italy (Others) Attributes with strong relationships with c19ProSo01 were found to be `c19perBeh01` , `c19perBeh03` , `c19ProSo02` , `c19ProSo03` , and `c19ProSo04` , with P-values smaller than 2e-16, indicating significance. Additionally, `c19RCA01` , `rankOrdLife_6` , and `c19perBeh02` showed relatively small values, suggesting a significant association with `c19ProSo01` . Furthermore, `c19RCA03` exhibited a value of 0.00333, indicating it as a reasonably good predictor. The Multiple R-squared is observed to be 0.3587.
2. c19ProSo02 - Non Italy (Others) The attributes identified as the best predictors for c19ProSo02 were `c19RCA03` , `c19RCA01` , `MLQ` , `lifeSat` , `c19ProSo01` , and `c19ProSo03` , all exhibiting P-values smaller than 2e-16, with a total of six receiving the smallest values. Following as good predictors were `c19PerBeh01` , `c19perBeh03` , `rankOrdLife_6` , and `c19ProSo04` , all receiving sufficiently small P-values and three asterisks, indicating their effectiveness. The Multiple R-squared is 0.3743.
3. c19ProSo03 - Non Italy (Others) The attributes identified as the best predictors for c19ProSo03 were `c19ProSo01` , `c19ProSo02` , and `c19ProSo04` , with P-values smaller than 2e-16, rated as the smallest values. Although `lifeSat` , `c19RCA02` , `rankOrdLife_1` , and `rankOrdLife_6` exhibited P-values larger than 2e-16, they still showed sufficiently small values, qualifying as good predictors. The overall Multiple R-squared value is 0.438, the highest among the four items.
4. c19ProSo04 - Non Italy (Others) The attributes identified as the best predictors for c19ProSo04 were `c19perBeh02` , `c19RCA03` , `c19RCA02` , `rankOrdLife_1` , `c19perBeh03` , `rankOrdLife_6` ,

`c19ProSo01` , and `c19ProSo03` , all exhibiting P-values smaller than 2e-16, rated as the best predictors. Furthermore, `c19RCA01` , `MLQ` , `lifeSat` , and `c19ProSo02` , although with P-values larger than 2e-16, were still rated as good predictors. The Multiple R-squared is 0.3176.

`Overall Comparing attributes compare to my focus country`

Overall, we compared strong predictors for Pro-Social Attributes in Italy's data with those in other countries' data. Generally, significantly discernible strong predictors were much more numerous in Non-Italy's data, consistently including Italy's strong predictors among them.

Question 3 - a

I've added the code for extracting external data for Question 3 to the appendix. I extracted the desired data from each data and attached each process to the appendix.

```
## New names:
## • `1996` -> `1996...3`
## • `1996` -> `1996...4`
## • `1996` -> `1996...5`
## • `1996` -> `1996...6`
## • `1996` -> `1996...7`
## • `1996` -> `1996...8`
## • `1998` -> `1998...9`
## • `1998` -> `1998...10`
## • `1998` -> `1998...11`
## • `1998` -> `1998...12`
## • `1998` -> `1998...13`
## • `1998` -> `1998...14`
## • `2000` -> `2000...15`
## • `2000` -> `2000...16`
## • `2000` -> `2000...17`
## • `2000` -> `2000...18`
## • `2000` -> `2000...19`
## • `2000` -> `2000...20`
## • `2002` -> `2002...21`
## • `2002` -> `2002...22`
## • `2002` -> `2002...23`
## • `2002` -> `2002...24`
## • `2002` -> `2002...25`
## • `2002` -> `2002...26`
## • `2003` -> `2003...27`
## • `2003` -> `2003...28`
## • `2003` -> `2003...29`
## • `2003` -> `2003...30`
## • `2003` -> `2003...31`
## • `2003` -> `2003...32`
## • `2004` -> `2004...33`
## • `2004` -> `2004...34`
## • `2004` -> `2004...35`
## • `2004` -> `2004...36`
## • `2004` -> `2004...37`
## • `2004` -> `2004...38`
## • `2005` -> `2005...39`
## • `2005` -> `2005...40`
## • `2005` -> `2005...41`
## • `2005` -> `2005...42`
## • `2005` -> `2005...43`
## • `2005` -> `2005...44`
## • `2006` -> `2006...45`
## • `2006` -> `2006...46`
## • `2006` -> `2006...47`
## • `2006` -> `2006...48`
## • `2006` -> `2006...49`
## • `2006` -> `2006...50`
## • `2007` -> `2007...51`
## • `2007` -> `2007...52`
## • `2007` -> `2007...53`
## • `2007` -> `2007...54`
## • `2007` -> `2007...55`
## • `2007` -> `2007...56`
## • `2008` -> `2008...57`
## • `2008` -> `2008...58`
## • `2008` -> `2008...59`
## • `2008` -> `2008...60`
## • `2008` -> `2008...61`
## • `2008` -> `2008...62`
## • `2009` -> `2009...63`
## • `2009` -> `2009...64`
## • `2009` -> `2009...65`
## • `2009` -> `2009...66`
## • `2009` -> `2009...67`
## • `2009` -> `2009...68`
## • `2010` -> `2010...69`
## • `2010` -> `2010...70`
## • `2010` -> `2010...71`
## • `2010` -> `2010...72`
## • `2010` -> `2010...73`
## • `2010` -> `2010...74`
## • `2011` -> `2011...75`
## • `2011` -> `2011...76`
## • `2011` -> `2011...77`
## • `2011` -> `2011...78`
## • `2011` -> `2011...79`
## • `2011` -> `2011...80`
## • `2012` -> `2012...81`
## • `2012` -> `2012...82`
## • `2012` -> `2012...83`
## • `2012` -> `2012...84`
## • `2012` -> `2012...85`
## • `2012` -> `2012...86`
## • `2013` -> `2013...87`
## • `2013` -> `2013...88`
## • `2013` -> `2013...89`
## • `2013` -> `2013...90`
## • `2013` -> `2013...91`
## • `2013` -> `2013...92`
## • `2014` -> `2014...93`
## • `2014` -> `2014...94`
## • `2014` -> `2014...95`
## • `2014` -> `2014...96`
## • `2014` -> `2014...97`
## • `2014` -> `2014...98`
## • `2015` -> `2015...99`
## • `2015` -> `2015...100`
## • `2015` -> `2015...101`
## • `2015` -> `2015...102`
## • `2015` -> `2015...103`
## • `2015` -> `2015...104`
## • `2016` -> `2016...105`
## • `2016` -> `2016...106`
## • `2016` -> `2016...107`
## • `2016` -> `2016...108`
## • `2016` -> `2016...109`
## • `2016` -> `2016...110`
## • `2017` -> `2017...111`
## • `2017` -> `2017...112`
## • `2017` -> `2017...113`
## • `2017` -> `2017...114`
## • `2017` -> `2017...115`
## • `2017` -> `2017...116`
## • `2018` -> `2018...117`
## • `2018` -> `2018...118`
## • `2018` -> `2018...119`
## • `2018` -> `2018...120`
## • `2018` -> `2018...121`
## • `2018` -> `2018...122`
## • `2019` -> `2019...123`
## • `2019` -> `2019...124`
## • `2019` -> `2019...125`
## • `2019` -> `2019...126`
## • `2019` -> `2019...127`
## • `2019` -> `2019...128`
## • `2020` -> `2020...129`
## • `2020` -> `2020...130`
## • `2020` -> `2020...131`
## • `2020` -> `2020...132`
## • `2020` -> `2020...133`
## • `2020` -> `2020...134`
## • `2021` -> `2021...135`
## • `2021` -> `2021...136`
## • `2021` -> `2021...137`
## • `2021` -> `2021...138`
## • `2021` -> `2021...139`
## • `2021` -> `2021...140`
## • `2022` -> `2022...141`
## • `2022` -> `2022...142`
## • `2022` -> `2022...143`
## • `2022` -> `2022...144`
## • `2022` -> `2022...145`
## • `2022` -> `2022...146`
```

```
# change rowname to coded_country
names(labour_datad)[names(labour_datad) == "labour_data$Country"] <- "coded_countr
y"

# politicsd, vaccinated, happyd, health_datad, labour_datad >> merge
merged_data <- left_join(politicsd, vaccinated, by = "coded_country") %>%
  left_join(gdpd, by = "coded_country") %>%
  left_join(happyd, by = "coded_country") %>%
  left_join(health_datad, by = "coded_country") %>%
  left_join(labour_datad, by = "coded_country")

write.csv(merged_data, "June Merged Data.csv", row.names = TRUE)
```

In the political part, I utilized an indicator representing the effectiveness of each country's government. This indicator ranges approximately from -2.5 (indicating weak effectiveness) to 2.5 (indicating strong effectiveness), providing valuable insights into the performance of governments.

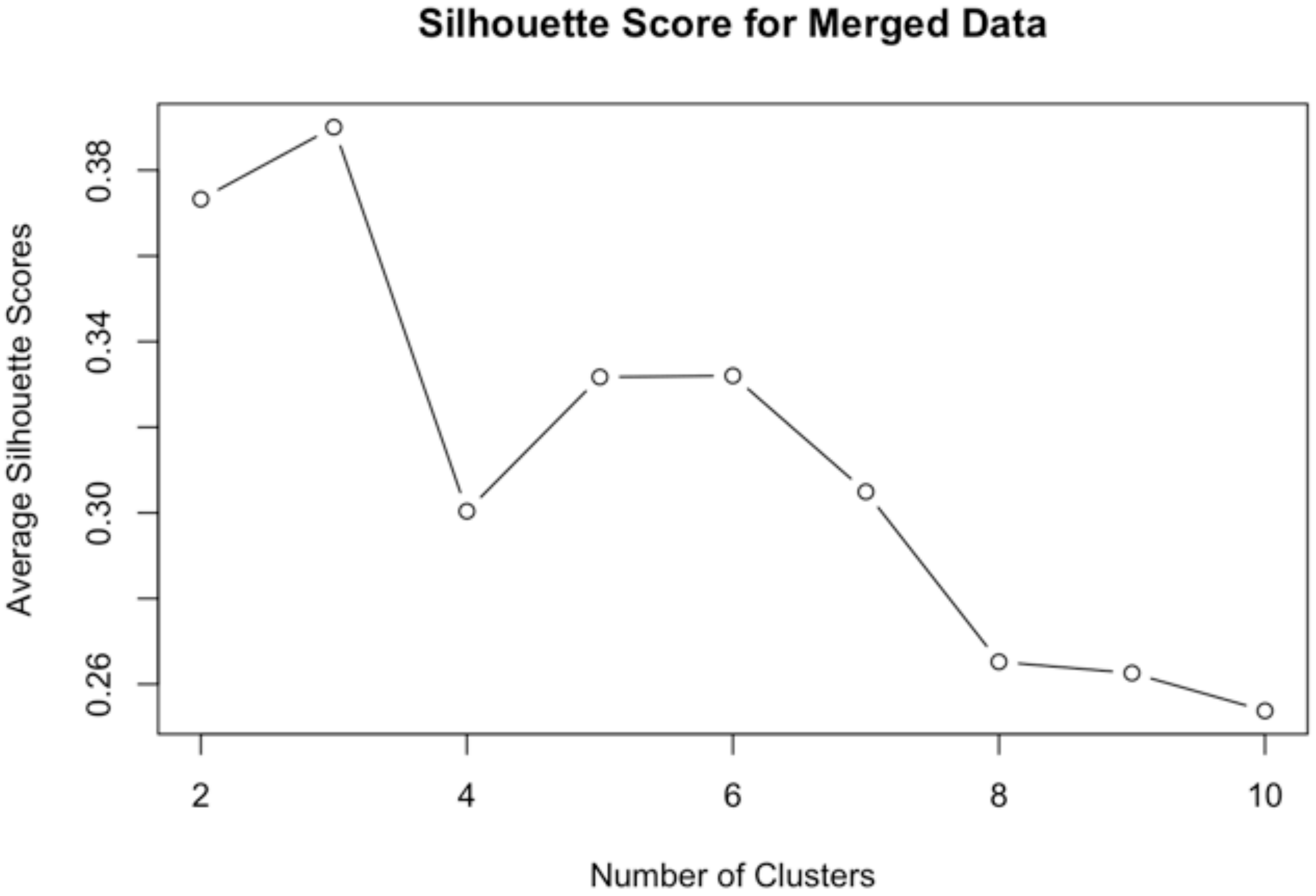
Additionally, I collected data on the vaccination rates of each country, represented as percentages. This aspect is crucial in understanding the vaccination status of each country, which is a key aspect of the response to COVID-19.

Furthermore, I gathered data on GDP per capita, reflecting the economic aspect of the dataset. GDP per capita serves as a measure of the average economic output per person in a country.

I also included data on the weekly average number of deaths, which is instrumental in evaluating the severity of the COVID-19 pandemic.

Lastly, I collected data related to the labor market impact of COVID-19, anticipating its significance. By employing a "left join" to combine these diverse datasets, I aimed to minimize missing data and constructed a comprehensive dataset named final_data.

```
plot(k,type='b',avg_sil,xlab='Number of Clusters', ylab='Average Silhouette Score
s', main=("Silhouette Score for Merged Data"))
```



```
ikfit <- kmeans(Q3_a_data[,2:7],3,nstart = 20)

table(actual = Q3_a_data$coded_country, fitted = ikfit$cluster)
```

```
##                    fitted
## actual              1 2 3
##   Australia         0 1 0
##   Austria           0 1 0
##   Belgium           0 1 0
##   Canada            0 1 0
##   Chile             1 0 0
##   Colombia          1 0 0
##   Costa Rica        1 0 0
##   Denmark           0 1 0
##   Estonia           1 0 0
##   Finland           0 1 0
##   France            0 1 0
##   Germany           0 1 0
##   Greece            1 0 0
##   Hungary           1 0 0
##   Iceland           0 1 0
##   Ireland           0 1 0
##   Israel            0 1 0
##   Italy             1 0 0
##   Japan             0 1 0
##   Latvia            1 0 0
##   Lithuania         1 0 0
##   Luxembourg        0 1 0
##   Mexico            1 0 0
##   Netherlands       0 1 0
##   New Zealand       0 1 0
##   Norway            0 1 0
##   Poland            1 0 0
##   Portugal          1 0 0
##   Slovenia          1 0 0
##   Spain             1 0 0
##   Sweden            0 1 0
##   Switzerland       0 1 0
##   United Kingdom 0 1 0
##   United States     0 0 1
```

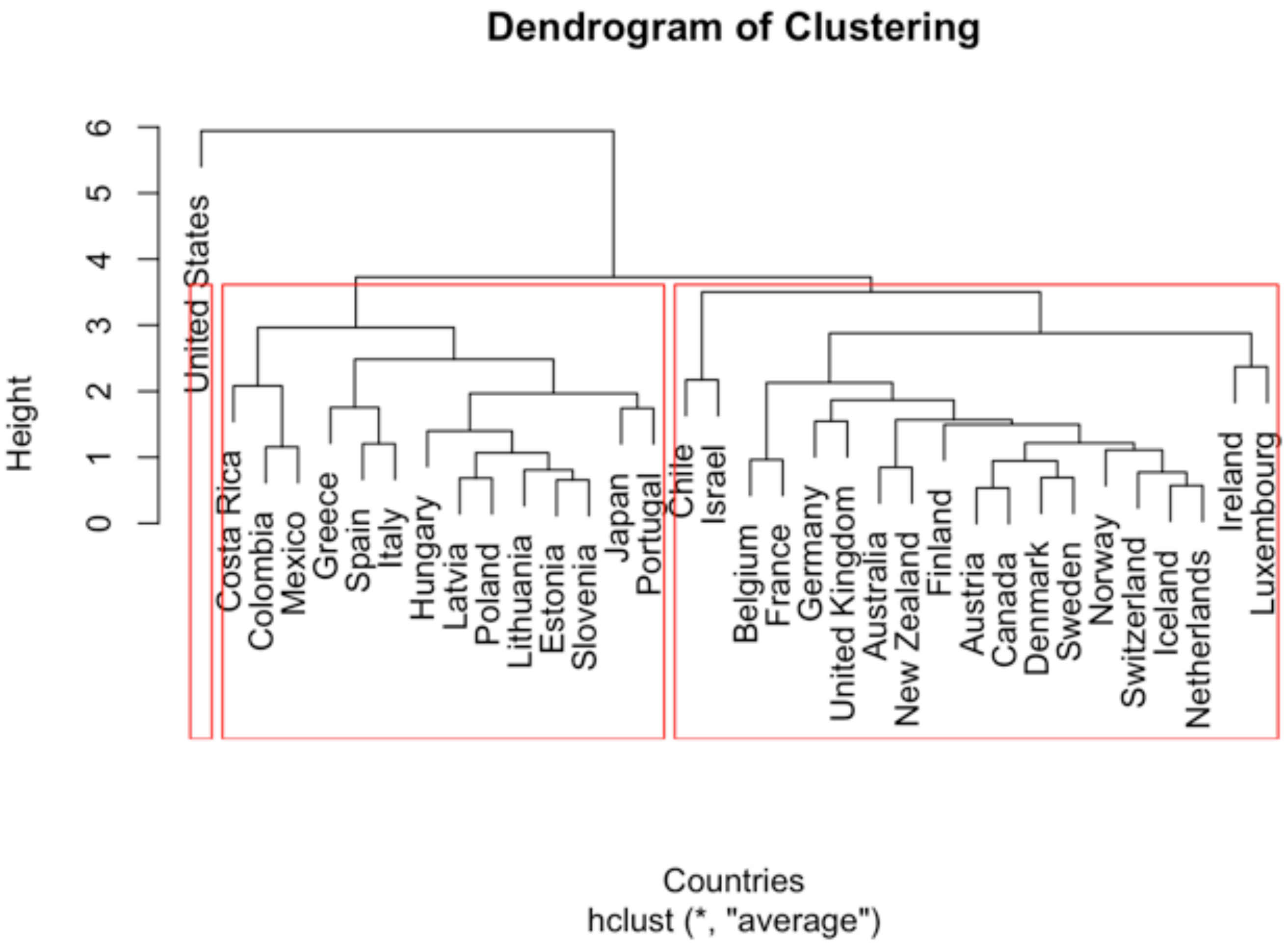
```
ikfit$cluster =as.factor(ikfit$cluster)
```

Answer for presentation.

I used silhouette score for finding the number of best clusters. In this case, max score recommends 3 clusters. So, I m going to use K-means clustering with 3 clusters. Finally, I could find that Chile, Colombia, Costa Rica, Greece, Hungary, Latvia, Lithuania, Mexico, Poland, Portugal, Slovenia, Spain are in the same clustering with Italy, which is my selected country.

However, I tried to make the dendrogram for comparing the answer would be same or not.

```
cb <- Q3_a_data

cb[,2:7] = scale(cb[2:7])

rownames(cb) = cb$coded_country

cfit = hclust(dist(cb[,2:7]), "average")

cut.cfit = cutree(cfit,k=3)

# visualize with plot
plot(cfit, main = "Dendrogram of Clustering", xlab = "Countries")
rect.hclust(cfit, k=3, border = "red")
```

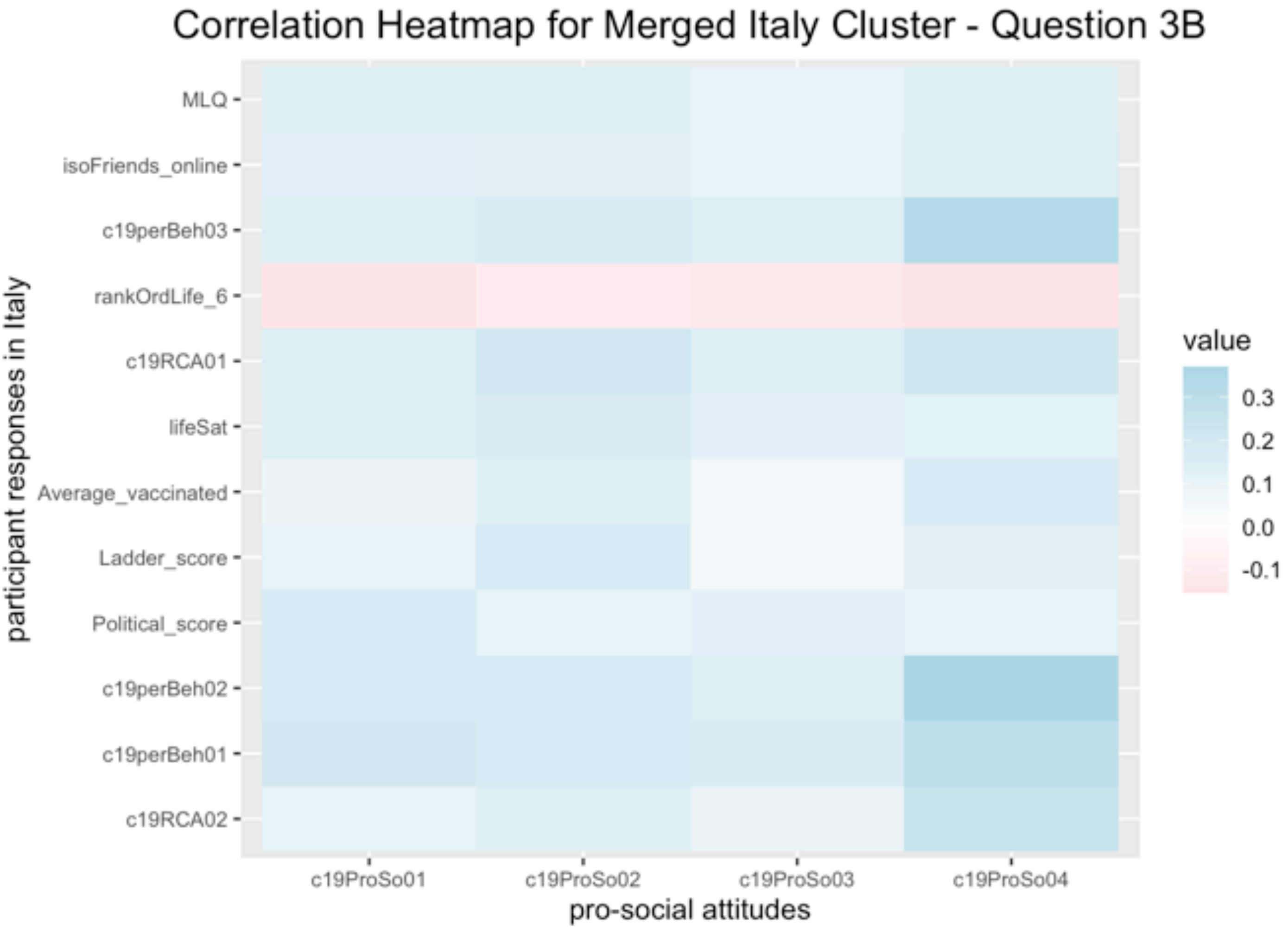


And, The Cluster Dendrogram's answer was slightly different.

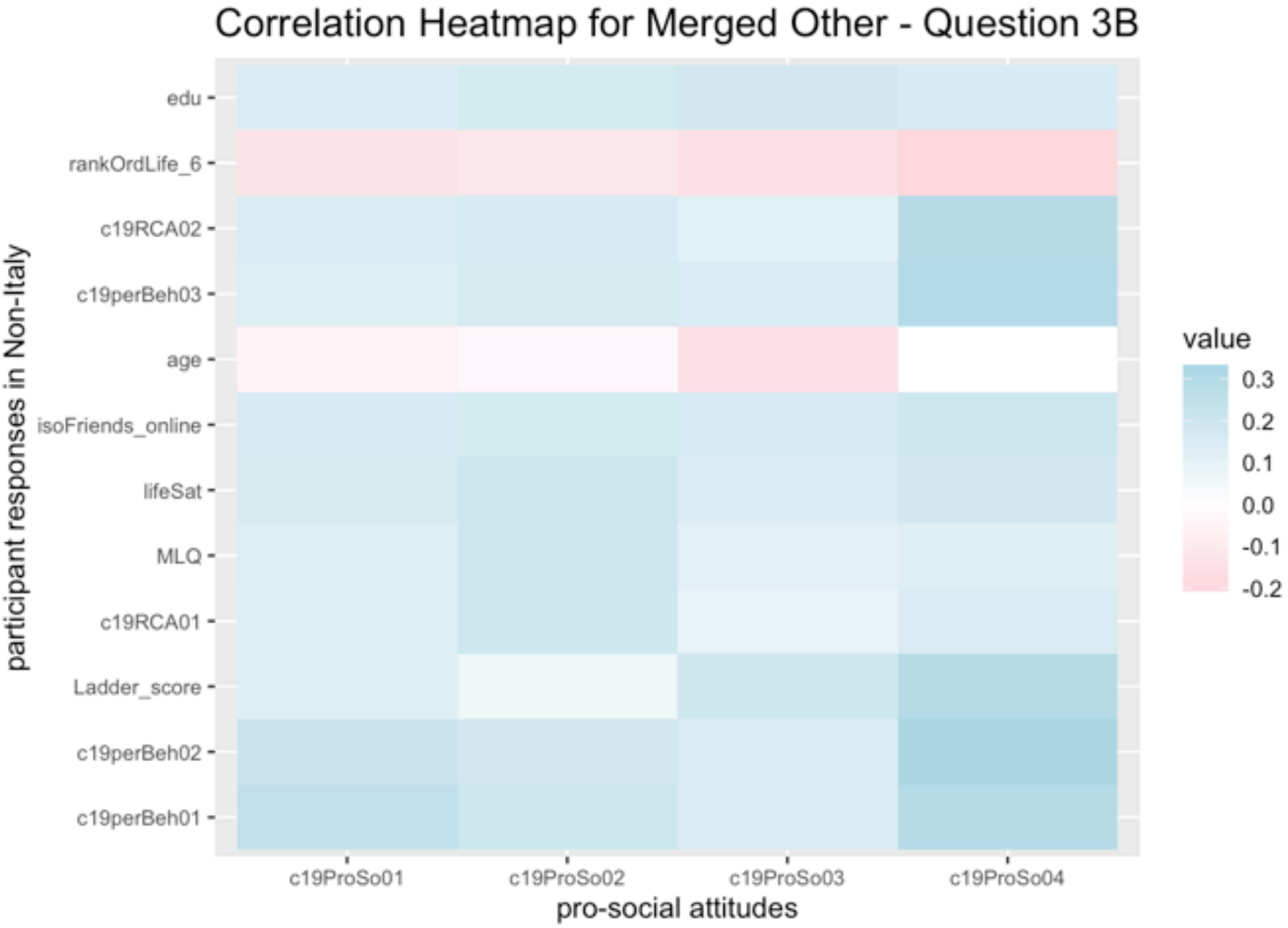
The similar countries were Spain, Greece, Hungary, Lativa, Poland, Lithuania, Estonia, Japan(it was not in the same clustering when I did ikfit), and Portugal.

Question 3 - b

```
heatmap_sa_italy_3B
```



```
heatmap_sa_other_3B
```



```
top_5_values_df_other_q3$Attribute
```

```
##  [1] "c19perBeh01"       "c19perBeh02"       "Ladder_score"
##  [4] "c19RCA01"          "MLQ"               "lifeSat"
##  [7] "isoFriends_online" "age"               "c19perBeh03"
## [10] "c19RCA02"          "rankOrdLife_6"     "edu"
```

```
top_5_values_df_italy_q3$Attribute
```

```
##  [1] "c19RCA02"          "c19perBeh01"         "c19perBeh02"
##  [4] "Political_score"   "Ladder_score"        "Average_vaccinated"
##  [7] "lifeSat"           "c19RCA01"            "rankOrdLife_6"
## [10] "c19perBeh03"       "isoFriends_online"   "MLQ"
```

Through this analysis, we were able to identify strong predictors in the merged dataset for both Italy Cluster countries and Other countries.

1. **c19ProSo01 - Merged Italy cluster** Attributes strongly associated with c19ProSo01 were identified as `c19ProSo02` , `c19ProSo03` with a P-value smaller than 2e-16. Additionally, `c19perBeh01` and `Ladder_score` , `MLQ` , `rankOrdLife_06` exhibited values smaller than 0.001, indicating some

degree of influence as predictors. The Multiple R-squared score is 0.3225.

- It is noticeable that the linear progression summary for c19ProSo01 in Question 2B shows a similarity in values. I could find new attributes in here, which is Ladder_score.

2. **c19ProSo02 - Merged Italy cluster** The attribute identified as the strongest predictor for c19ProSo02 was `c19ProSo01` , `c19ProSo03` , `Ladder_score` , `lifeSat` with a P-value smaller than 2e-16. The Multiple R-squared score is 0.3274.
    - A difference was observed compared to the linear progression summary for c19ProSo02 in Question 2B. In Italy data, c19RCA03 was considered, which was not identified as a strong predictor in the Merged Data. It was also noted that the P-values for c19ProSo01, 03 were quite similar. I could find new attributes in here, which is Ladder_score.
3. **c19ProSo03 - Merged Italy cluster** The attributes identified as the best predictors for c19ProSo03 were `c19ProSo01` and `c19ProSo02` , `c19ProSo04` with P-values smaller than 2e-16, rated as the strongest attributes. `Ladder_score` , `age` , `c19RCA02` also exhibited, indicating it as a strong predictor. The Multiple R-squared score is 0.3983.
    - The data for the linear progression summary of c19ProSo03 showed almost identical results.
4. **c19ProSo04 - Merged Italy cluster** The attributes identified as the best predictors for c19ProSo04 were `c19ProSo01` , `c19ProSo03` , `c19perBeh02` , `Ladder_score` , `c19perBeh03` with P-values smaller than 2e-16, rated as the strongest attributes. `c19perBeh01` , `c19RCA02` , `rankOrdLife_6` , `edu` also exhibited, indicating it as a strong predictor. The Multiple R-squared score is 0.3126.
    - Similarity was observed compared to the linear progression summary for c19ProSo04 in Question 2B. I could find new attributes in here, which is Ladder_score.
5. **c19ProSo01 - Merged Non Italy (Others)** Attributes strongly associated with c19ProSo01 were identified as `c19perBeh01` , `c19ProSo02` , `c19ProSo03` , and `c19ProSo04` , `isoFriends_online` , `c19perBeh03` , `rankOrdLife_6` . The Multiple R-squared score is 0.3384.
    - A similarity was observed with the linear progression summary for c19ProSo01 in Question 2C. A notable difference was the decreased P-value for `MLQ` compared to the 2C Others Data.
6. **c19ProSo02 - Merged Non Italy (Others)** The attribute identified as the strongest predictor for c19ProSo02 was `c19ProSo01` , and `c19ProSo03` , `c19ProSo04` . Additionally, good predictors included `c19PerBeh01` , `c19RCA01` , `isoFriends_online` , `LifeSat` , `MLQ` , and `c19perBeh03` , `edu` . The Multiple R-squared score is 0.3098.
    - A similarity was observed with the linear progression summary for c19ProSo02 in Question 2C. One notable difference was the inclusion of `rankOrdLife_6` as a strong predictor in 2C Others Data, which was not observed in Merged Others Data.
7. **c19ProSo03 - Merged Non Italy (Others)** The attributes identified as the best predictors for c19ProSo03 were `c19ProSo01` , `c19ProSo02` , `c19ProSo04` , and `age` , `c19perBeh02` with P-values smaller than 2e-16, rated as the strongest attributes. Additionally, `c19RCA02` , `rankOrdLife_6` , and `edu` exhibited sufficiently small P-values, indicating them as good predictors. The Multiple R-squared value is 0.4272.
    - A similarity was observed with the linear progression summary for c19ProSo03 in Question 2C. One notable difference was the inclusion of `rankOrdLife_1` as a strong predictor in 2C Others Data, which was not observed in Merged Others Data. Additionally, in 3B Others, `age`

emerged as a strong predictor, marking a significant difference.

8. **c19ProSo04 - Merged Non Italy (Others)** The attributes identified as the best predictors for c19ProSo04 were `c19perBeh02` , `isoFriends_online` , `c19RCA02` , `c19perBeh03` , `Ladder_score` , `rankOrdlife_6` , `c19ProSo01` , and `c19ProSo03` . These attributes exhibited P-values smaller than 2e-16, indicating them as the best predictors. Additionally, `c19perBeh01` , `age` , `edu` , `c19ProSo02` were also considered good predictors. The Multiple R-squared value is 0.3588.
    - A similarity was observed with the linear progression summary for c19ProSo04 in Question 2C. One notable difference was the absence of `MLQ` as a strong predictor in 2C Others Data, while in 3B Merged Others Data, `edu` and `age` , `ladder_score` emerged as new strong predictors.

Overall Comparison of Strong Predictors between Italy and Others Data

In summary, we were able to compare the strong predictor attributes for Pro-Social Attributes in Italy data with those in others' data. Overall, it was observed that the number of significant strong predictors is notably higher in Non-Italy data compared to Italy data. However, there was nothing much big difference that within these predictors, except of `ladder score` . the attributes from other data consistently include strong predictors present in Italy Clustering countries data.

This suggests a degree of consistency in the predictors across different data sets, with Italy data sharing common strong predictors with others' data despite having fewer significant predictors overall.

Appendix

Question 1 - A

```
dim(cvbase)

# Counting the number of NA
na_counts <- colSums(is.na(cvbase_filter_country))



str(cvbase)
```

Question 1 - B

```
 Pre-processing session based on column type
preprocessed_data <- function(data) {
  for (i in 1:ncol(data)) {
    column <- data[[i]]
    if (i %in% 1:10) {
      # the NA value in column [1:10] will be 0
      column[is.na(column)] <- 0
    } else if (i %in% 11:26) {
      # the NA value in column [11:24] will be median of their value
      median_val <- median(column, na.rm = TRUE)
      column[is.na(column)] <- median_val
    } else if (i %in% 33:38) {
      # the NA value in column [33:38] will be median of their value
      median_val <- median(column, na.rm = TRUE)
      column[is.na(column)] <- median_val
    } else if (i %in% 39:44) {
      # the NA value in column [39:44] will be 0
      column[is.na(column)] <- 0
    } else if (i %in% 44:47) {
      # the NA value in column [45:47] will be median of their value
      median_val <- median(column, na.rm = TRUE)
      column[is.na(column)] <- median_val
    } else if (i %in% 49:52) {
      # the NA value in column [49:52] will be median of their value
      median_val <- median(column, na.rm = TRUE)
      column[is.na(column)] <- median_val
    }
    data[[i]] <- column
  }
  return(data)
}

# Pre-processing done for int
cvbase_updated <- preprocessed_data(cvbase_filter_country)

# Change Chr(A~F) to 1~6 int
cvbase_updated[27:32] <- lapply(cvbase_updated[27:32], function(x) ifelse(is.na(
x), NA, match(x, LETTERS)))

preprocessed_data2 <- function(data) {
  # make it NA to median which column is [27:32]
  data[, 27:32] <- lapply(data[, 27:32], function(x) {
    median_val <- median(x, na.rm = TRUE)
    replace(x, is.na(x), median_val)
  })

  return(data)
}

cvbase_final <- preprocessed_data2(cvbase_updated)
```

Question 2 - A

```
# make copy of pre-processed data
Q2_sample <- cvbase_final

# extract data for selected country from Q2_sample
Q2_sample_for_italy = Q2_sample %>% filter(Q2_sample$coded_country == 'Italy')
Q2_sample_for_italy_save = Q2_sample_for_italy

# extract data for non-selected country from Q2 sample
Q2_sample_not_italy = Q2_sample %>% filter(Q2_sample$coded_country != 'Italy')
Q2_sample_not_italy_save = Q2_sample_not_italy

# Make all data type to integer except Coded_Country
Q2_sample_for_italy <- Q2_sample_for_italy %>%
  mutate_if(function(x) !is.character(x), as.integer)

# subset non-integer data for both table
Q2_sample_for_italy <- subset(Q2_sample_for_italy, select = -coded_country)
Q2_sample_not_italy <- subset(Q2_sample_not_italy, select = -coded_country)

# make average data table for Question 2
mean_data_italy <- colMeans(Q2_sample_for_italy)
mean_data_non_italy <- colMeans(Q2_sample_not_italy)

Q2_a_result <- t.test(Q2_sample_for_italy,Q2_sample_not_italy)

Q2_a_result

# trying to see difference between both table set.
mean_diff <- mean_data_italy - mean_data_non_italy

diff_table <- data.frame(
  Attribute = names(mean_diff),
  Mean_Difference = mean_diff)

diff_table$Attribute <- factor(diff_table$Attribute, levels = unique(diff_table$Attribute))

# ggplot2 graph
comparison_plot <- ggplot(diff_table, aes(x = Attribute, y = Mean_Difference)) +
  geom_bar(stat = "identity", fill = "skyblue", width = 0.5) +
  labs(title = "Difference in Mean Values between Italy and Non-Italy",
       x = "Attribute",
       y = "Mean Difference (italy - non italy)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1), plot.title = element_text(hjust = 0.5))

comparison_plot
```

Question 2 - B

```
c19_italy <-colMeans(Q2_sample_for_italy[48:51])

italy_table <- data.frame(
  Attribute = names(c19_italy),
  Mean_value = c19_italy)

Q2_b_table <- ggplot(italy_table, aes(x = Attribute, y = Mean_value)) +
  geom_bar(stat = "identity", fill = "grey", width = 0.5) +
  geom_text(aes(label = sprintf("%.2f", Mean_value)), vjust = -0.5, size = 3) +
  labs(title = "Average Value for Corona ProSocial Behavior in Italy",
       x = "Attribute", y = "Value") +
  theme_minimal() +
  theme(axis.text.x = element_text(), plot.title = element_text(hjust = 0.5))

Q2_b_table

# 4 of c19 columns
Q2_b_italy_c19 <- Q2_sample_for_italy[48:51]

# whole responses
Q2_b_italy_response <- Q2_sample_for_italy[1:47]

# getting the cor relationship between c19 and responses
Q2_italy_cor <- cor(Q2_b_italy_response, Q2_b_italy_c19)

# make the empty data frame for 2 a
top_5_values_df <- data.frame(Attribute = character(), Value = numeric(), stringsAsFactors = FALSE)

# find the attribute that has biggest value of attribute for cor relationship,
# each row 7 attributes
for (col_index in 1:ncol(Q2_italy_cor)) {
  col_values <- Q2_italy_cor[, col_index]
  abs_col_values <- abs(col_values)
  top_5_idx <- order(abs_col_values, decreasing = TRUE)[1:7]
  top_5_values <- col_values[top_5_idx]
  top_5_attribute_names <- rownames(Q2_italy_cor)[top_5_idx]

  # save 7 for each row, exclude same attributes
  for (i in 1:7) {
    if (!(top_5_attribute_names[i] %in% top_5_values_df$Attribute)) {
      top_5_values_df <- rbind(top_5_values_df, data.frame(Attribute = top_5_attribute_names[i], Value = top_5_values[i]))
    }
  }
}
```

```
# sort
top_5_values_df <- top_5_values_df[order(-abs(top_5_values_df$Value)),]
top_5_values_df

# pro_social_col
Q2_B_proso_for_italy <- colnames(Q2_sample_for_italy)[48:51]

# pro_social extract data
pro_social_for_italy <- Q2_sample_for_italy[, Q2_B_proso_for_italy]

# select which attributes is selected in data
selected_attributes <- top_5_values_df$Attribute

# calculate the correlation between data and pro-social attributes
corr_matrix_for_italy <- cor(Q2_sample_for_italy[, selected_attributes], pro_socia
l_for_italy)

# reframe the dataset for heatmap
corr_melt_for_italy <- melt(corr_matrix_for_italy)

# heatmap
heatmap_sa_for_italy <- ggplot(corr_melt_for_italy, aes(Var2, Var1)) +
  geom_tile(aes(fill = value)) +
  scale_fill_gradient2(low = "pink", high ="lightblue", mid = "white", midpoint =
0) +
  theme(axis.text.x = element_text(size = 8),
        axis.text.y = element_text(size = 5),
        plot.title = element_text(size = 12, hjust = 0.5)) +
  labs(title = "Correlation Heatmap for Italy - Question 2B") +
  xlab("pro-social attitudes") +
  ylab("participant responses in Italy")

# heatmap
heatmap_sa_for_italy

# Progression for making attribute format for lm()
predictors_formula_other <- paste(top_5_values_df$Attribute, collapse = " + ")

Q2_italy_so1 <- lm(paste("c19ProSo01 ~", predictors_formula_other, "+ c19ProSo02 +
c19ProSo03 + c19ProSo04"), data = Q2_sample_for_italy)

Q2_italy_so2 <- lm(paste("c19ProSo02 ~", predictors_formula_other, "+ c19ProSo01 +
c19ProSo03 + c19ProSo04"), data = Q2_sample_for_italy)

Q2_italy_so3 <- lm(paste("c19ProSo03 ~", predictors_formula_other, "+ c19ProSo01 +
c19ProSo02 + c19ProSo04"), data = Q2_sample_for_italy)

Q2_italy_so4 <- lm(paste("c19ProSo04 ~", predictors_formula_other, "+ c19ProSo01 +
c19ProSo02 + c19ProSo03"), data = Q2_sample_for_italy)
```

```
# Output
summary(Q2_italy_so1)

summary(Q2_italy_so2)

summary(Q2_italy_so3)

summary(Q2_italy_so4)
```

Question 2 - C

```
# C19 for Non-Italy
c19_not_italy <-colMeans(Q2_sample_not_italy[48:51])

# Making Non-Italy Data Frame
not_italy_table <- data.frame(
  Attribute = names(c19_not_italy),
  Mean_value = c19_not_italy)

# ggplot
q2_c_table <- ggplot(not_italy_table, aes(x = Attribute, y = Mean_value)) +
  geom_bar(stat = "identity", fill = "grey", width = 0.5) +
  geom_text(aes(label = sprintf("%.2f", Mean_value)), vjust = -0.5, size = 3) +
  labs(title = "Average Value for Corona ProSocial Behavior (Non-Italy)",
       x = "Attribute", y = "Value") +
  theme_minimal() +
  theme(axis.text.x = element_text(), plot.title = element_text(hjust = 0.5))

q2_c_table

# C19, 4 columns for Non-Italy Q2-c
Q2_c_other_c19 <- Q2_sample_not_italy[48:51]

# Responses from Non-Italy people
Q2_c_other_response <- Q2_sample_not_italy[1:47]

# Co-relationship for Non-Italy
Q2_non_italy_cor <- cor(Q2_c_other_response,Q2_c_other_c19)

# Making the Data Frame
top_5_values_df_other <- data.frame(Attribute = character(), Value = numeric(), st
ringsAsFactors = FALSE)



# find the attribute that has biggest value of attribute for cor relationship,
# each row 7 attributes
for (col_index in 1:ncol(Q2_non_italy_cor)) {
  col_values_other <- Q2_non_italy_cor[, col_index]
  abs_col_values <- abs(col_values_other)
  top_5_idx_other <- order(abs_col_values, decreasing = TRUE)[1:7]
  top_5_values_other <- col_values_other[top_5_idx_other]
  top_5_attribute_names_other <- rownames(Q2_non_italy_cor)[top_5_idx_other]

  # save 7 for each row, exclude same attributes
  for (i in 1:7) {
    if (!(top_5_attribute_names_other[i] %in% top_5_values_df_other$Attribute)) {
      top_5_values_df_other <- rbind(top_5_values_df_other, data.frame(Attribute =
top_5_attribute_names_other[i], Value = top_5_values_other[i]))
    }
  }
}

# sort
top_5_values_df_other <- top_5_values_df_other[order(-abs(top_5_values_df_other$Va
lue)),]

top_5_values_df_other

# trying to get column name of prosocial attribute
Q2_C_proso_not_italy <- colnames(Q2_sample_not_italy)[48:51]

# extract prosocial data
pro_social_not_italy <- Q2_sample_not_italy[, Q2_C_proso_not_italy]

# extract chosen attribute
selected_attributes_other <- top_5_values_df_other$Attribute

# find the correlation between prosocial attribute value and data
corr_matrix_not_italy <- cor(Q2_sample_not_italy[, selected_attributes_other], pro
_social_not_italy)

# reframe the data for heatmap
corr_melt_not_italy <- melt(corr_matrix_not_italy)

# heatmap
heatmap_sa_not_italy <- ggplot(corr_melt_not_italy, aes(Var2, Var1)) +
  geom_tile(aes(fill = value)) +
  scale_fill_gradient2(low = "pink", high ="lightblue", mid = "white", midpoint =
0) +
  theme(axis.text.x = element_text(size = 8),
        axis.text.y = element_text(size = 5),
        plot.title = element_text(size = 12, hjust = 0.5)) +
  labs(title = "Correlation Heatmap not Italy - Question 2C") +
  xlab("pro-social attitudes") +
  ylab("participant responses in Not - Italy")

heatmap_sa_not_italy

# Progression for making attribute format for lm()
```

```
predictors_formula_other <- paste(top_5_values_df_other$Attribute, collapse = " +
")

Q2_other_so1 <- lm(paste("c19ProSo01 ~", predictors_formula_other, "+ c19ProSo02 +
c19ProSo03 + c19ProSo04"), data = Q2_sample_not_italy)

Q2_other_so2 <- lm(paste("c19ProSo02 ~", predictors_formula_other, "+ c19ProSo01 +
c19ProSo03 + c19ProSo04"), data = Q2_sample_not_italy)

Q2_other_so3 <- lm(paste("c19ProSo03 ~", predictors_formula_other, "+ c19ProSo01 +
c19ProSo02 + c19ProSo04"), data = Q2_sample_not_italy)

Q2_other_so4 <- lm(paste("c19ProSo04 ~", predictors_formula_other, "+ c19ProSo01 +
c19ProSo02 + c19ProSo03"), data = Q2_sample_not_italy)

# Output
summary(Q2_other_so1)

summary(Q2_other_so2)

summary(Q2_other_so3)

summary(Q2_other_so4)
```

Question 3 - A

Political https://www.worldbank.org/en/publication/worldwide-governance-indicators (https://www.worldbank.org/en/publication/worldwide-governance-indicators)

Vaccinated & Average GDP per person https://ourworldindata.org/covid-vaccinations (https://ourworldindata.org/covid-vaccinations)

HAPPY - Ladder score (overall happeniess score) https://worldhappiness.report/ed/2021/#appendices-and-data (https://worldhappiness.report/ed/2021/#appendices-and-data)

Mortality https://stats.oecd.org/viewhtml.aspx?datasetcode=HEALTH_MORTALITY&lang=en (https://stats.oecd.org/viewhtml.aspx?datasetcode=HEALTH_MORTALITY&lang=en)

Short-Term Labour Market Statistics (/1000 people) https://stats.oecd.org/index.aspx?lang=en (https://stats.oecd.org/index.aspx?lang=en)

```
# Politics part : Government Performance Score
politics <- read_excel("DataforPolitics.xlsx", sheet = "GovernmentEffectiveness")

politicsd <- data.frame(politics$Data,politics[75])

names(politicsd)[names(politicsd) == "politics.Data"] <- "coded_country"
names(politicsd)[names(politicsd) == "X2011...75"] <- "Political_score"

# exclude the data if that data has N/A
```

```
politicsd <- politicsd[2:nrow(politicsd), ]

# exclude the data if that data has N/A
politicsd[politicsd == "#N/A"] <- NA
politicsd <- na.omit(politicsd)

politicsd$Political_score <- as.numeric(politicsd$Political_score)

# Vaccinated Rate + GDP per person Data Extracting.
covid_data <- read.csv("DataforCovid.csv")

# filter for 2021 years data
covid_data <- covid_data %>%
  filter(substr(as.character(date), 1, 4) == "2021")

# group by based on coded_country
vaccinated <- covid_data %>%
  group_by(covid_data$location) %>%
  summarize(Average_vaccinated = mean(total_vaccinations_per_hundred, na.rm = TRU
E))

# change name to row name to coded_country
names(vaccinated)[names(vaccinated) == "covid_data$location"] <- "coded_country"

# GDP
gdpd <- covid_data %>%
  group_by(covid_data$location) %>%
  summarize(gdp_per_person = mean(gdp_per_capita, na.rm=TRUE))

names(gdpd)[names(gdpd) == "covid_data$location"] <- "coded_country"


# HAPPY data set
happy <- read_excel("DataForHappy.xls")

# extracting ladder score from happy data
happyd <- data.frame(
  Country_name = happy$`Country name`,
  Ladder_score = happy$`Ladder score`
)
# change name to coded_country for merge in the future
names(happyd)[names(happyd) == "Country_name"] <- "coded_country"

# Average death rate per week
health <- read.csv("DataforHealth.csv")

# filtering 2021 years only
health_data <- health %>%
  filter(Year == 2021)
```

```
# group by coded country
health_datad <- health_data %>%
  group_by(health_data$Country) %>%
  summarize(Average_deaths_per_week = mean(Value, na.rm = TRUE))

# change rowname to coded_country
names(health_datad)[names(health_datad) == "health_data$Country"] <- "coded_countr
y"


# Short - Term labour rate
labour <- read.csv("DataForLabour.csv")

# filtering 2021 year
labour_data <- labour %>%
  filter(Time == 2021)

# group by coded country
labour_datad <- labour_data %>%
  group_by(labour_data$Country) %>%
  summarize(Average_labour_rate = mean(Value, na.rm = TRUE))

# change rowname to coded_country
names(labour_datad)[names(labour_datad) == "labour_data$Country"] <- "coded_countr
y"

# politicsd, vaccinated, happyd, health_datad, labour_datad >> merge
merged_data <- left_join(politicsd, vaccinated, by = "coded_country") %>%
  left_join(gdpd, by = "coded_country") %>%
  left_join(happyd, by = "coded_country") %>%
  left_join(health_datad, by = "coded_country") %>%
  left_join(labour_datad, by = "coded_country")

#
write.csv(merged_data, "June Merged Data.csv", row.names = TRUE)

Q3_a_data <- merged_data

# exclude the data if that data has N/A
Q3_a_data <- na.omit(Q3_a_data)

# make it numeric
Q3_a_data[,2:7] <- lapply(Q3_a_data[,2:7], as.numeric)

# normalization
Q3_a_data[,2:7] <- scale(Q3_a_data[,2:7])

# progression of getting silhouette score
i_silhouette_score <- function(k){
  km <- kmeans(Q3_a_data[,2:7], centers = k, nstart=25)
  ss <- silhouette(km$cluster, dist(Q3_a_data[,2:7]))
  mean(ss[,3])
}

k <- 2:10
avg_sil <- sapply(k,i_silhouette_score)

plot(k,type='b',avg_sil,xlab='Number of Clusters', ylab='Average Silhouette Scores', main=("Silhouette Score for Merged Data"))


ikfit <- kmeans(Q3_a_data[,2:7],3,nstart = 20)

table(actual = Q3_a_data$coded_country, fitted = ikfit$cluster)

ikfit$cluster =as.factor(ikfit$cluster)

cb <- Q3_a_data

cb[,2:7] = scale(cb[2:7])

rownames(cb) = cb$coded_country

cfit = hclust(dist(cb[,2:7]), "average")

cut.cfit = cutree(cfit,k=3)

plot(cfit, main = "Dendrogram of Clustering", xlab = "Countries")
rect.hclust(cfit, k=3, border = "red")

june <- read.csv("June Merged Data.csv")

june
```

Question 3 - B

```
# final merging with merged_data (extracted from myself) and finalized cvbase
Q3_b_data <- left_join(cvbase_final, merged_data, by = "coded_country")
Q3_b_data <- na.omit(Q3_b_data)


clustered_countries <- c('Italy', 'Chile', 'Colombia', 'Costa Rica', 'Greece', 'Hungary', 'Latvia', 'Lithuania', 'Mexico', 'Poland', 'Portugal', 'Slovenia', 'Spain')

Q3_b_data_for_italy <- Q3_b_data %>% filter(coded_country %in% clustered_countries)

Q3_b_data_non_italy <- Q3_b_data %>% filter(!coded_country %in% clustered_countrie
s)

Q3_b_data_for_italy <- Q3_b_data_for_italy %>% select(-coded_country)
Q3_b_data_non_italy <- Q3_b_data_non_italy %>% select(-coded_country)

# Responses from Non-Italy people
Q3_b_italy_response <- Q3_b_data_for_italy[,-c(48:51)]
Q3_b_other_response <- Q3_b_data_non_italy[,-c(48:51)]

# C19, 4 columns for Non-Italy Q2-c
Q3_b_italy_c19 <- Q3_b_data_for_italy[48:51]
Q3_b_other_c19 <- Q3_b_data_non_italy[48:51]

Q3_italy_cor <- cor(Q3_b_italy_response, Q3_b_italy_c19)
Q3_other_cor <- cor(Q3_b_other_response, Q3_b_other_c19)

# Making the Data Frame
top_5_values_df_italy_q3 <- data.frame(Attribute = character(), Value = numeric(),
stringsAsFactors = FALSE)
top_5_values_df_other_q3 <- data.frame(Attribute = character(), Value = numeric(),
stringsAsFactors = FALSE)

# find the attribute that has the biggest absolute value of attribute for cor rela
tionship, each row 7 attributes
for (col_index in 1:ncol(Q3_italy_cor)) {
  col_values_italy <- Q3_italy_cor[, col_index]
  abs_col_values <- abs(col_values_italy)
  top_5_idx_italy <- order(abs_col_values, decreasing = TRUE)[1:7]
  top_5_values_italy <- col_values_italy[top_5_idx_italy]
  top_5_attribute_names_italy <- rownames(Q3_italy_cor)[top_5_idx_italy]

  # save 7 for each row, exclude same attributes
  for (i in 1:7) {
    if (!(top_5_attribute_names_italy[i] %in% top_5_values_df_italy_q3$Attribute))
{
      top_5_values_df_italy_q3 <- rbind(top_5_values_df_italy_q3, data.frame(Attri
bute = top_5_attribute_names_italy[i], Value = top_5_values_italy[i]))
    }
  }
}

# sort
top_5_values_df_italy_q3 <- top_5_values_df_italy_q3[order(-abs(top_5_values_df_it
aly_q3$Value)),]


# Making the Data Frame
top_5_values_df_other_q3 <- data.frame(Attribute = character(), Value = numeric(),
stringsAsFactors = FALSE)
```

```r
# find the attribute that has the biggest absolute value of attribute for cor relationship, each row 7 attributes
for (col_index in 1:ncol(Q3_other_cor)) {
  col_values_other <- Q3_other_cor[, col_index]
  abs_col_values <- abs(col_values_other)
  top_5_idx_other <- order(abs_col_values, decreasing = TRUE)[1:7]
  top_5_values_other <- col_values_other[top_5_idx_other]
  top_5_attribute_names_other <- rownames(Q3_other_cor)[top_5_idx_other]

  # save 7 for each row, exclude same attributes
  for (i in 1:7) {
    if (!(top_5_attribute_names_other[i] %in% top_5_values_df_other_q3$Attribute)) {
      top_5_values_df_other_q3 <- rbind(top_5_values_df_other_q3, data.frame(Attribute = top_5_attribute_names_other[i], Value = top_5_values_other[i]))
    }
  }
}

# sort
top_5_values_df_other_q3 <- top_5_values_df_other_q3[order(-abs(top_5_values_df_other_q3$Value)),]

# get the colname of pro-social attribute
Q3_b_italy_c19 <- colnames(Q3_b_data_for_italy[48:51])

# extract pro_social data
pro_social_italy <- Q3_b_data_for_italy[, Q3_b_italy_c19]

# select the chosen attribute from the data
selected_attributes_italy <- top_5_values_df_italy_q3$Attribute

# calculate the correlation between pro-social attribute with data
corr_matrix_italy <- cor(Q3_b_data_for_italy[, selected_attributes_italy], pro_social_italy)

# reframe the data for heatmap
corr_melt_italy <- melt(corr_matrix_italy)

# heatmap
heatmap_sa_italy_3B <- ggplot(corr_melt_italy, aes(Var2, Var1)) +
  geom_tile(aes(fill = value)) +
  scale_fill_gradient2(low = "pink", high ="lightblue", mid = "white", midpoint = 0) +
  theme(axis.text.x = element_text(size = 8),
        axis.text.y = element_text(size = 8),
        plot.title = element_text(size = 15, hjust = 0.5)) +
  labs(title = "Correlation Heatmap for Merged Italy - Question 3B") +
  xlab("pro-social attitudes") +
  ylab("participant responses in Italy")
```

```
# Progression for making attribute format for lm()
predictors_formula_italy_q3 <- paste(top_5_values_df_other_q3$Attribute, collapse = " + ")

Q3_italy_so1 <- lm(paste("c19ProSo01 ~", predictors_formula_italy_q3, "+ c19ProSo02 + c19ProSo03 + c19ProSo04"), data = Q3_b_data_for_italy)

Q3_italy_so2 <- lm(paste("c19ProSo02 ~", predictors_formula_italy_q3, "+ c19ProSo01 + c19ProSo03 + c19ProSo04"), data = Q3_b_data_for_italy)

Q3_italy_so3 <- lm(paste("c19ProSo03 ~", predictors_formula_italy_q3, "+ c19ProSo01 + c19ProSo02 + c19ProSo04"), data = Q3_b_data_for_italy)

Q3_italy_so4 <- lm(paste("c19ProSo04 ~", predictors_formula_italy_q3, "+ c19ProSo01 + c19ProSo02 + c19ProSo03"), data = Q3_b_data_for_italy)

predictors_formula_other_q3 <- paste(top_5_values_df_other_q3$Attribute, collapse = " + ")

Q3_other_so1 <- lm(paste("c19ProSo01 ~", predictors_formula_other_q3, "+ c19ProSo02 + c19ProSo03 + c19ProSo04"), data = Q3_b_data)

Q3_other_so2 <- lm(paste("c19ProSo02 ~", predictors_formula_other_q3, "+ c19ProSo01 + c19ProSo03 + c19ProSo04"), data = Q3_b_data)

Q3_other_so3 <- lm(paste("c19ProSo03 ~", predictors_formula_other_q3, "+ c19ProSo01 + c19ProSo02 + c19ProSo04"), data = Q3_b_data)

Q3_other_so4 <- lm(paste("c19ProSo04 ~", predictors_formula_other_q3, "+ c19ProSo01 + c19ProSo02 + c19ProSo03"), data = Q3_b_data)
```