Dissertation R Markdown

Alexandre Dore

2024-07-30

Settup and General Pre-Processing.

Load packages.

```
summary(cars)
```

```
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

```
# Load packages
library(DescTools)
```

```
## Warning: package 'DescTools' was built under R version 4.2.3
```

```
library(bfsMaps)
library("sf")
```

```
## Warning: package 'sf' was built under R version 4.2.3
```

```
## Linking to GEOS 3.11.0, GDAL 3.5.3, PROJ 9.1.0; sf_use_s2() is TRUE
```

```
library(tidyverse)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.3
```

```
## Warning: package 'tidyr' was built under R version 4.2.3
```

```
## Warning: package 'readr' was built under R version 4.2.3
```

```
## Warning: package 'dplyr' was built under R version 4.2.3
```

```
## Warning: package 'stringr' was built under R version 4.2.3
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2

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

```
library("SMMT")
library(xml2)
```

```
## Warning: package 'xml2' was built under R version 4.2.3
```

```
library(fixest)
```

```
## Warning: package 'fixest' was built under R version 4.2.3
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(lmtest)
```

```
## Loading required package: zoo
##
## Attaching package: 'zoo'
##
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(orcutt)
library(sandwich)
```

```
## Warning: package 'sandwich' was built under R version 4.2.3
```

Load all data sets.

```r
# Load all data sets:

# Load the nationality data for 2010 and 2021
data_10 <- read.csv("Dis 2010 data nationality csv.csv")
data_21 <- read.csv("Dis 2021 data nationality csv.csv")

# Load the referendum results data
FaceBan <- read.csv("Face ban referendum relevant data.csv")
Minaret <- read.csv("Minaret referendum relevant data.csv")


# Load the covariates data, as well as that for sub-group
# analyses

# Loading the population density data
Pop_density_09 <- read.csv("clean 2009 pop-density.csv")
Pop_density_20 <- read.csv("clean 2020 pop-density.csv")

# Loading the social help data
Social_help_09 <- read.csv("clean 2009 social help.csv")
Social_help_20 <- read.csv("clean 2020 social help.csv")

# Loading the Income data
Average_income_10 <- read.csv("clean 2010 income data.csv")
Average_income_20 <- read.csv("clean 2020 income data.csv", na.strings = c("",
    "NA", "#VALUE!"))
# Assign NAs to the missing values from the original data
# file

# Loading the SVP support data
SVP_support_07 <- read.csv("UDC parliament 2007.csv", na.strings = c("",
    "NA", "#VALUE!"))  # Assign NAs
SVP_support_19 <- read.csv("UDC parliament 2019.csv")

# Rename the columns to correct error made in the Excel
# data processing
names(SVP_support_07)[names(SVP_support_07) == "SPD_votes_07"] <- "SVP_votes_07"
names(SVP_support_19)[names(SVP_support_19) == "SPD_votes_19"] <- "SVP_votes_19"

# Loading the district-level crime data
crime_20 <- read.csv("2020 dis crime actual dataset.csv")

# Loading the linguistic regions data
Region_22 <- read.csv("4 Linguisitic regions 2022 Dis.csv")

# Loading the municipality type data
Municipality_type_24 <- read.csv("Municipality types.csv")
```

Investigate whether there are any NAs in the imported covariate data sets (there are none in the outcome, demographic, crime, regional/municipality type ones)

```r
summary(Pop_density_09)
```

```
##  Municipality_Number Total_population   Municipality_total_km2_area
##  Min.   :   1        Length:2624        Min.   :  0.31
##  1st Qu.:1206        Class :character   1st Qu.:  3.96
##  Median :3584        Mode  :character   Median :  7.34
##  Mean   :3382                           Mean   : 15.24
##  3rd Qu.:5429                           3rd Qu.: 14.42
##  Max.   :6810                           Max.   :283.98
```

```
summary(Pop_density_20)
```

```
##  Municipality_Number Total_population   Municipality_total_km2_area
##  Min.   :   1        Length:2198        Min.   :  0.310
##  1st Qu.:1086        Class :character   1st Qu.:  4.370
##  Median :3340        Mode  :character   Median :  8.135
##  Mean   :3307                           Mean   : 18.192
##  3rd Qu.:5415                           3rd Qu.: 16.550
##  Max.   :6810                           Max.   :438.610
```

```
summary(Social_help_09)
```

```
##  Municipality_Number Number_of_recipients Total_population
##  Min.   :   1        Length:2624          Length:2624
##  1st Qu.:1206        Class :character     Class :character
##  Median :3584        Mode  :character     Mode  :character
##  Mean   :3382
##  3rd Qu.:5429
##  Max.   :6810
```

```
summary(Social_help_20)
```

```
##  Municipality_Number Number_of_recipients Total_population
##  Min.   :   1        Length:2198          Length:2198
##  1st Qu.:1086        Class :character     Class :character
##  Median :3340        Mode  :character     Mode  :character
##  Mean   :3307
##  3rd Qu.:5415
##  Max.   :6810
```

```
summary(Average_income_10)
```

```
##  Municipality_Number Number_of_taxpayers Municipality_total_revenue.
##  Min.   :   1        Min.   :    12      Min.   :5.657e+05
##  1st Qu.:1188        1st Qu.:   213      1st Qu.:1.501e+07
##  Median :3572        Median :   525      Median :3.910e+07
##  Mean   :3377        Mean   :  1446      Mean   :1.157e+08
##  3rd Qu.:5452        3rd Qu.:  1317      3rd Qu.:1.006e+08
##  Max.   :6810        Max.   :195121      Max.   :1.615e+10
```

```
summary(Average_income_20)
```

```
##  Municipality_Number Number_of_taxpayers Municipality_total_revenue
##  Min.   :   1        Min.   :    28      Min.   :1.279e+06
##  1st Qu.:1086        1st Qu.:   360      1st Qu.:2.763e+07
##  Median :3340        Median :   762      Median :6.190e+07
##  Mean   :3307        Mean   :  1840      Mean   :1.591e+08
##  3rd Qu.:5415        3rd Qu.:  1778      3rd Qu.:1.455e+08
##  Max.   :6810        Max.   :202999      Max.   :1.917e+10
##                      NA's   :75          NA's   :75
```

```
summary(SVP_support_07)
```

```
##    bfs_nr_old   total_votes_07      SVP_votes_07
##  Min.   :   1   Length:2712        Min.   :    0.0
##  1st Qu.:1323   Class :character   1st Qu.:   39.0
##  Median :3642   Mode  :character   Median :  102.0
##  Mean   :3443                      Mean   :  257.4
##  3rd Qu.:5465                      3rd Qu.:  268.0
##  Max.   :6806                      Max.   :24421.0
##                                    NA's   :57
```

```
summary(SVP_support_19)
```

```
##    bfs_nr_old   total_votes_19      SVP_votes_19
##  Min.   :   1   Length:2206        Min.   :    0.0
##  1st Qu.:1093   Class :character   1st Qu.:   66.5
##  Median :3342   Mode  :character   Median :  149.0
##  Mean   :3312                      Mean   :  283.8
##  3rd Qu.:5413                      3rd Qu.:  339.0
##  Max.   :6810                      Max.   :15143.0
##                                    NA's   :3
```

```
summary(crime_20)
```

```
##    District          Total_property_related_crimes Violent_offences
##  Length:143          Min.   : 0.00                 Min.   : 0.000
##  Class :character    1st Qu.:12.70                 1st Qu.: 2.800
##  Mode  :character    Median :17.30                 Median : 4.100
##                      Mean   :20.55                 Mean   : 4.373
##                      3rd Qu.:24.20                 3rd Qu.: 5.300
##                      Max.   :96.80                 Max.   :14.500
##  Total_criminal_offenses
##  Min.   :  8.50
##  1st Qu.: 25.15
##  Median : 32.10
##  Mean   : 36.45
##  3rd Qu.: 41.00
##  Max.   :144.00
```

```
# There is around 30% NAs in the social help data sets, 75 in the
# 2020 average income, and 57 and 3 in the 2007 and 2019 SVP supports, respectively
```

```r
# Given the consequent volume of NAs in the social help data sets,
# we do not consider imputation as it would likely not be representative of the
# actual values.We decide to exclude these variables from the models as using
# them would imply cutting the data set size in half.

# The SVP and income missing values will be imputed once the data has been
# updated to the 2024 boundaries, for higher similarity among the municipalities
# used for imputation, considering municipality type
```

Load data for mapping tables (MT) **Below is an Important Comment for the Reader wanting to run the Code**

```r
# Download and extract the 2024 municipality inventory form a defined online
# source - this only needs to be done once, as the data is available locally afterwards
# path_inventory_xml <- download_municipality_inventory(path = getwd())

# Download the 2024 municipality inventory
Municipality_inventory_file <- read_xml("eCH0071_240101.xml")

# import_CH_municipality_inventory imports the Swiss municipality inventory
# from the raw XML resource into R as a tibble.
mutations_object <- import_CH_municipality_inventory(Municipality_inventory_file)
mutations        <- mutations_object$mutations
```

Create all the relevant old and new states before creating the mapping tables.

```r
# Create all the relevant old and new states before creating the mapping tables

# The spatial reference (SR) of the nationality data, the 2024-01-01,
# will form the new state of all other data sets as it is the latest in time,
# hence that with the least municipalities.
new_state_2024 <- as.Date("2024-01-01")

# Define an old state for the 2022 spatial referenced data
old_state_2022 <- as.Date("2022-05-01")

# Vote data: Define 2009 and 2021 old states corresponding to the spatial references
# of the referendum results.
old_state_vote_2009 <- as.Date("2009-12-30") # Setting the SR a day earlier than FSO;
# there seems do be an issue otherwise, (removes 428 municipalities from "Minaret")
old_state_vote_2021 <- as.Date("2021-01-01")

# Non-vote covariates: Define 2009, 2010, and 2020 old states.
old_state_cov2009 <- as.Date("2009-10-25")
old_state_cov2010 <- as.Date("2010-11-21")
old_state_cov2020 <- as.Date("2020-10-18")

# SVP-covariates: Define 2007 and 2019 old states.
old_state_SVP2007 <- as.Date("2007-12-30")
# Again, setting 1 day earlier than 31/12 to avoid issues.
old_state_SVP2019 <- as.Date("2019-01-01")
```

Create two MT-making functions. This is to account for the merger of municipalities forming Verzasca in 2020, not included in the "Swiss Municipal Data Merger Tool".

```r
# Set up the pre-2020 MT-making function
clean_mapping_table_maker_pre_2020 <- function(mutations, old_state, new_state) {
  # Create a mapping table
  mapping_object <- map_old_to_new_state(mutations, old_state, new_state)
  mapping_table <- mapping_object$mapped

  # Define lakes to be removed
  lakes_to_remove <- c("Zürichsee (ZH)", "Zürichsee (SZ)", "Zürichsee (SG)",
                "Thunersee", "Brienzersee", "Bielersee (BE)", "Lac de Bienne (NE)",
                "Lac de Neuchâtel (BE)", "Lac de Neuchâtel (FR)",
                "Lac de Neuchâtel (VD)", "Lac de Neuchâtel (NE)", "Baldeggersee",
                "Sempachersee", "Hallwilersee (LU)", "Hallwilersee (AG)",
                "Zugersee (LU)", "Zugersee (SZ)", "Zugersee (ZG)",
                "Vierwaldstättersee (LU)", "Vierwaldstättersee (UR)",
                "Vierwaldstättersee (SZ)", "Vierwaldstättersee (OW)",
                "Vierwaldstättersee (NW)", "Sihlsee", "Sarnersee", "Walensee (GL)",
                "Walensee (SG)", "Aegerisee", "Lac de la Gruyère", "Murtensee (FR)",
                "Lac de Morat (VD)", "Bodensee (SH)", "Bodensee (SG)", "Bodensee (TG)",
                "Lago di Lugano (o.Camp.)", "Lago Maggiore", "Lac de Joux",
                "Lac Léman (VD)", "Lac Léman (VS)", "Lac Léman (GE)")

  # Remove the lakes from the mapping table
  mapping_table <- mapping_table[!mapping_table$name_old %in% lakes_to_remove, ]

  # Handle the special case for the last lake with the same name as a municipality,
  # as well as erroneous municipalities
  mapping_table <- mapping_table[!mapping_table$bfs_nr_old %in% c(9040, 2391,
                                                                  5391, 5394), ]

  # Remove the original Verzasca row (without valid old BFS numbers)
  mapping_table <- mapping_table[!mapping_table$bfs_nr_new == 5399
                                | !is.na(mapping_table$bfs_nr_old), ]

  # Manually add the municipalities merging to form Verzasca in 2020
  verzasca_mergers <- data.frame(
    bfs_nr_new = rep(5399, 4),
    name_new = rep("Verzasca", 4),
    bfs_nr_old = c(5095, 5129, 5102, 5135),
    name_old = c("Brione (Verzasca)", "Sonogno", "Corippo", "Vogorno")
  )

  # Combine the manually added mergers with the existing mapping table
  mapping_table <- rbind(mapping_table, verzasca_mergers)

  # Correct NA entries for unchanged municipalities whose "old data" is missing
  na_indices <- which(is.na(mapping_table$bfs_nr_old))
  mapping_table[na_indices, "bfs_nr_old"] <- mapping_table[na_indices, "bfs_nr_new"]
  mapping_table[na_indices, "name_old"] <- mapping_table[na_indices, "name_new"]

  return(mapping_table)
}
```

```r
# Set up the MT-making function for 2020 and after
clean_mapping_table_maker_from_2020 <- function(mutations, old_state, new_state) {
  # Create a mapping table
  mapping_object <- map_old_to_new_state(mutations, old_state, new_state)
  mapping_table <- mapping_object$mapped

  # Define lakes to be removed
  lakes_to_remove <- c("Zürichsee (ZH)", "Zürichsee (SZ)", "Zürichsee (SG)",
                "Thunersee", "Brienzersee", "Bielersee (BE)", "Lac de Bienne (NE)",
                "Lac de Neuchâtel (BE)", "Lac de Neuchâtel (FR)",
                "Lac de Neuchâtel (VD)", "Lac de Neuchâtel (NE)", "Baldeggersee",
                "Sempachersee", "Hallwilersee (LU)", "Hallwilersee (AG)",
                "Zugersee (LU)", "Zugersee (SZ)", "Zugersee (ZG)",
                "Vierwaldstättersee (LU)", "Vierwaldstättersee (UR)",
                "Vierwaldstättersee (SZ)", "Vierwaldstättersee (OW)",
                "Vierwaldstättersee (NW)", "Sihlsee", "Sarnersee", "Walensee (GL)",
                "Walensee (SG)", "Aegerisee", "Lac de la Gruyère", "Murtensee (FR)",
                "Lac de Morat (VD)", "Bodensee (SH)", "Bodensee (SG)", "Bodensee (TG)",
                "Lago di Lugano (o.Camp.)", "Lago Maggiore", "Lac de Joux",
                "Lac Léman (VD)", "Lac Léman (VS)", "Lac Léman (GE)")

  # Remove the lakes from the mapping table
  mapping_table <- mapping_table[!mapping_table$name_old %in% lakes_to_remove, ]

  # Handle the special case for the last lake with the same name as a municipality,
  # as well as 3 erroneous municipalities
  mapping_table <- mapping_table[!mapping_table$bfs_nr_old %in% c(9040, 2391,
                                                                 5391, 5394), ]

  # Correct NA entries for unchanged municipalities whose "old data" is missing
  na_indices <- which(is.na(mapping_table$bfs_nr_old))
  mapping_table[na_indices, "bfs_nr_old"] <- mapping_table[na_indices,
                                                          "bfs_nr_new"]
  mapping_table[na_indices, "name_old"] <- mapping_table[na_indices, "name_new"]

  return(mapping_table)
}
```

Create all necessary MTs.

```r
# MT to be used on the 2022 data
mapping_table_2022 <- clean_mapping_table_maker_from_2020(mutations,
                                                         old_state_2022,
                                                         new_state_2024)

# MT to be used on Minaret data
mapping_table_vote2009 <- clean_mapping_table_maker_pre_2020(mutations,
                                                            old_state_vote_2009,
                                                            new_state_2024)

# # MT to be used on Face Ban data
```

```r
mapping_table_vote2021 <- clean_mapping_table_maker_from_2020(mutations,
                                                              old_state_vote_2021,
                                                              new_state_2024)

# MT for non-vote covariates

# MT 2009 covariates
mapping_table_cov2009 <- clean_mapping_table_maker_pre_2020(mutations,
                                                            old_state_cov2009,
                                                            new_state_2024)

# MT for 2010 covariates
mapping_table_cov2010 <- clean_mapping_table_maker_pre_2020(mutations,
                                                            old_state_cov2010,
                                                            new_state_2024)

# MT for 2020 covariates
mapping_table_cov2020 <- clean_mapping_table_maker_from_2020(mutations,
                                                             old_state_cov2020,
                                                             new_state_2024)

# MT for SVP-covariates

# MT for 2007 Parliamentary Election
mapping_table_SVP2007 <- clean_mapping_table_maker_pre_2020(mutations,
                                                            old_state_SVP2007,
                                                            new_state_2024)

# MT for 2019 Parliamentary Election
mapping_table_SVP2019 <- clean_mapping_table_maker_pre_2020(mutations,
                                                            old_state_SVP2019,
                                                            new_state_2024)


# Investigate the presence of NAs in the MTs
sum(is.na(mapping_table_2022))
```

```
## [1] 0
```

```r
sum(is.na(mapping_table_vote2009))
```

```
## [1] 0
```

```r
sum(is.na(mapping_table_vote2021))
```

```
## [1] 0
```

```r
sum(is.na(mapping_table_cov2009))
```

```
## [1] 0
```

```r
sum(is.na(mapping_table_cov2010))
```

```
## [1] 0
```

```r
sum(is.na(mapping_table_SVP2007))
```

```
## [1] 0
```

```r
sum(is.na(mapping_table_SVP2019))
```

```
## [1] 0
```

```r
# No NA at all.
```

For relevant data sets, change the name of the Municipality_Number column to bfs_nr_old in order to use the left_join and the MT:

```r
# Nationality data:
colnames(data_10)[colnames(data_10) == "bfs_nr_new"] <- "bfs_nr_old"
colnames(data_21)[colnames(data_21) == "bfs_nr_new"] <- "bfs_nr_old"

# Linguistic region data:
colnames(Region_22)[colnames(Region_22) == "bfs_nr_new"] <- "bfs_nr_old"

# Referendum results:
colnames(Minaret)[colnames(Minaret) == "Municipality.number"] <- "bfs_nr_old"
colnames(FaceBan)[colnames(FaceBan) == "Municipality.number"] <- "bfs_nr_old"

# Population density
colnames(Pop_density_09)[colnames(Pop_density_09) ==
                           "Municipality_Number"] <- "bfs_nr_old"
colnames(Pop_density_20)[colnames(Pop_density_20) ==
                           "Municipality_Number"] <- "bfs_nr_old"

# Social help
colnames(Social_help_09)[colnames(Social_help_09) ==
                           "Municipality_Number"] <- "bfs_nr_old"
colnames(Social_help_20)[colnames(Social_help_20) ==
                           "Municipality_Number"] <- "bfs_nr_old"

# Income
colnames(Average_income_10)[colnames(Average_income_10) ==
                              "Municipality_Number"] <- "bfs_nr_old"
colnames(Average_income_20)[colnames(Average_income_20) ==
                              "Municipality_Number"] <- "bfs_nr_old"

# No need to do anything for the SVP-vote data as this has been handled
# in Excel directly.
```

End of Set Up Section

From here on, specifics of data sets require we abandon the grouped approach. Each data set will be cleaned in turn, and its municipality borders updated to the SR of the nationality data (2024). For ease of navigation through the code, operations on each data set will be indexed.

As it serves as base for the entire analysis, I start with the cleaning and formatting the nationality data:

Part 1: Cleaning and Formatting the Nationality Data

Part 1.1: Cleaning the 2021 Nationality Data

```
# Remove unnecessary columns and rename the first two to more understandable titles
data_21 <- subset(data_21, select = -c(X, X.2))
colnames(data_21)[1] <- "Area"
colnames(data_21)[2] <- "Area_Population_21"

# Remove the second row from the data set because it only contains 0's
data_21 <- data_21[-2, ]

# Reset row names
rownames(data_21) <- NULL

# Calculate total foreigners per area by summing up population of each nationality
# except Switzerland (that is, all but the first three columns)
data_21$total_foreigners_21 <- rowSums(data_21[,-(1:3)], na.rm = TRUE)
#2244181 foreigners. This matches official data.


# Create a list of Arab countries
arab_countries <- c("Algeria", "Bahrain", "Comoros", "Djibouti", "Egypt", "Iraq",
                    "Jordan", "Kuwait", "Lebanon", "Libya", "Mauritania",
                    "Morocco", "Oman", "Palestine", "Qatar", "Saudi.Arabia",
                    "Somalia", "Sudan", "Syria", "Tunisia",
                    "United.Arab.Emirates", "Yemen")

# Create a list of Muslim (OIC member)
muslim_countries <- c("Afghanistan", "Albania", "Azerbaijan", "Bahrain",
                      "Bangladesh", "Benin", "United.Arab.Emirates", "Brunei",
                       "Burkina.Faso", "Algeria", "Djibouti", "Chad", "Indonesia",
                       "Morocco", "Côte.d.Ivoire", "Palestine", "Gabon", "Gambia",
                       "Guinea", "Guinea.Bissau", "Guyana", "Iraq", "Iran",
                      "Cameroon", "Qatar", "Kazakhstan", "Kyrgyzstan", "Comoros",
                      "Kuwait", "Libya", "Lebanon", "Maldives", "Malaysia",
                      "Mali", "Egypt", "Mauritania", "Mozambique", "Niger",
                      "Nigeria", "Uzbekistan", "Pakistan", "Senegal",
                      "Sierra.Leone", "Somalia", "Sudan", "Suriname",
                      "Syria", "Saudi.Arabia", "Tajikistan", "Togo", "Tunisia",
                       "Türkiye", "Turkmenistan", "Uganda", "Oman", "Jordan",
                       "Yemen")
```

```r
Turkey <- "Türkiye"


# Sum the number of nationals from each of the listed Arab countries
# listed for each municipality (row)
data_21$Arab_Pop_21 <- rowSums(data_21[arab_countries], na.rm = TRUE)

# Sum the number of nationals from each of the OIC member countries for each
# municipality (row)
data_21$Muslim_Pop_21 <- rowSums(data_21[muslim_countries], na.rm = TRUE)

# Sum the number of nationals from Turkey
data_21$Turkish_Pop_21 <- rowSums(data_21[Turkey], na.rm = TRUE)

# Create a new data frame with only the relevant columns
data_21_clean <- data_21 %>%
  select(Area, Area_Population_21, total_foreigners_21, Arab_Pop_21, Muslim_Pop_21 ,
         Turkish_Pop_21)
```

Part 1.2: Cleaning the 2010 Nationality Data

```r
# Make column names identical between both nationality data sets
colnames(data_10)[2] <- "Area_Population_10"

# Calculate total foreigners per area by summing up population of each
# nationality except Switzerland
data_10$total_foreigners_10 <- rowSums(data_10[,-(1:3)], na.rm = TRUE)
# 1766277 people, matches the official data.

# Sum the population for the Arab countries listed for each row
data_10$Arab_Pop_10 <- rowSums(data_10[arab_countries], na.rm = TRUE)

# Sum the population for the OIC member countries listed for each row
data_10$Muslim_Pop_10 <- rowSums(data_10[muslim_countries], na.rm = TRUE)


# Sum the number of nationals from Turkey
data_10$Turkish_Pop_10 <- rowSums(data_10[Turkey], na.rm = TRUE)

# Create a new data frame with relevant columns only
data_10_clean <- data_10 %>%
  select(Area, Area_Population_10, total_foreigners_10, Arab_Pop_10, Muslim_Pop_10
         , Turkish_Pop_10)
```

Part 1.3: Merging the 2021 and 2010 Nationality Data Frames by “Area” and Cleaning.

```r
merged_data <- full_join(data_21_clean, data_10_clean, by = "Area")
```

```r
# We then need to clean the merged data

# Downloading the register of Swiss districts
Swiss_Districts <- read.csv("District data .csv")

# Removing the Swiss level data in the first row of merged_data
merged_data <- subset(merged_data, Area != "Switzerland")

# Resetting row names to start from 1
rownames(merged_data) <- NULL

# Create 'Canton' and 'District' columns
merged_data <- merged_data %>%
  mutate(Canton = ifelse(grepl("^-", Area), sub("^-\\s*", "", Area), NA_character_),
         District = ifelse(grepl("^>>", Area), sub("^>>\\s*", "", Area),
                           NA_character_)) %>%
  fill(Canton, .direction = "down") %>%
  fill(District, .direction = "down")

# Reordering columns to have Canton 1st, District and, Area 3rd, then everything else
merged_data <- merged_data %>%
  select(Canton, District, Area, everything())

# Removing the rows representing the 26 cantons in the Area column
merged_data <- subset(merged_data, !grepl("^-\\s", merged_data$Area))

# Checking the dimensions of the new dataset
dim(merged_data) # All good, only the 26 cantons were removed.
```

```
## [1] 2288   13
```

```r
# Removing the rows containing Swiss districts, i.e.,
# those preceded by ">>" in the Area column
merged_data <- subset(merged_data, !grepl("^>>", merged_data$Area))

# Checking the dimensions of the new data set
dim(merged_data) # Success, only the 143 districts were removed.
```

```
## [1] 2145   13
```

```r
# Splitting municipality numbers and names into two columns
merged_data <- separate(merged_data, Area, into = c("bfs_nr_old",
                                                    "Municipality_Name"),
                        sep = " ", remove = FALSE)
```

```
## Warning: Expected 2 pieces. Additional pieces discarded in 301 rows [1, 2, 4, 6, 14, 16,
## 17, 18, 32, 57, 61, 85, 88, 90, 91, 93, 94, 101, 104, 107, ...].
```

```r
# Removing dots before the actual number in the new bfs_nr_old column
merged_data$bfs_nr_old <- gsub("\\.+\\b", "", merged_data$bfs_nr_old)
```

```r
# Removing the Area column
merged_data_final <- subset(merged_data, select = -Area)

# Removing leading zeros from the bfs_nr_old column
merged_data_final$bfs_nr_old <- gsub("^0+", "", merged_data_final$bfs_nr_old)

# Checking the structure of the updated dataset
str(merged_data_final)
```

```
## 'data.frame':    2145 obs. of  14 variables:
##  $ Canton             : chr  "Zürich" "Zürich" "Zürich" "Zürich" ...
##  $ District           : chr  "Bezirk Affoltern" "Bezirk Affoltern" "Bezirk Affoltern" "Bezirk Affolt
##  $ bfs_nr_old         : chr  "1" "2" "3" "4" ...
##  $ Municipality_Name  : chr  "Aeugst" "Affoltern" "Bonstetten" "Hausen" ...
##  $ Area_Population_21 : int  1988 12404 5632 3852 3780 1290 2445 641 5570 5760 ...
##  $ total_foreigners_21: num  295 3631 993 607 689 ...
##  $ Arab_Pop_21        : num  3 94 26 7 21 5 8 0 43 37 ...
##  $ Muslim_Pop_21      : num  12 253 68 39 68 16 18 0 62 76 ...
##  $ Turkish_Pop_21     : num  4 60 8 11 25 5 4 0 7 17 ...
##  $ Area_Population_10 : int  1824 11091 5159 3360 3410 918 1800 616 4243 4686 ...
##  $ total_foreigners_10: num  225 2832 657 391 451 ...
##  $ Arab_Pop_10        : num  1 35 3 5 8 0 0 0 10 10 ...
##  $ Muslim_Pop_10      : num  8 138 15 19 25 1 2 0 16 17 ...
##  $ Turkish_Pop_10     : num  1 54 8 11 3 0 0 0 4 5 ...
```

End of part 1 and the Nationality Data Cleaning

Part 2: Merge the district-level crime data with the Nationality Data

```r
merged_data_final$District <- trimws(merged_data_final$District)
crime_20$District <- trimws(crime_20$District)

# Start by defining long regex patterns using paste0
prefix_patterns_1 <- paste0(
  "^(Bezirk\\s+|Kanton\\s+|de\\s+|District\\s+|District\\sdu\\s+|",
  "Verwaltungskreis\\s+|Wahlkreis\\s+|Region\\s+|Arrondissement\\sadministratif\\s+|",
  "Distretto\\sdi\\s+|Bezirk\\s+|d'|du\\s+|l')"
)

prefix_patterns_2 <- paste0(
  "^(Bezirk\\s+|Kanton\\s+|District\\s+|Verwaltungskreis\\s+|Wahlkreis\\s+|Region\\s+|",
  "Kt.\\s+|Arrondissement\\sadministratif\\s+|Distretto\\sdi\\s+|d'|du\\s+|l')"
)

# Round 1
merged_data_final <- merged_data_final %>%
  mutate(District = str_remove_all(District, prefix_patterns_1),
         District = str_replace_all(District, c("See / District du Lac" = "See / Lac")))
```

```r
crime_20 <- crime_20 %>%
  mutate(District = str_remove_all(District, prefix_patterns_2),
         District = str_replace_all(District, c("Lavau0-Oron" = "Lavaux-Oron",
                                                "Appenzell i.Rh." =
                                                  "Appenzell Innerrhoden")))

# Round 2
merged_data_final <- merged_data_final %>%
  mutate(District = str_remove_all(District, prefix_patterns_1),
         District = str_replace_all(District, c("See / District du Lac" = "See / Lac")))

crime_20 <- crime_20 %>%
  mutate(District = str_remove_all(District, prefix_patterns_2),
         District = str_replace_all(District, c("Lavau0-Oron" = "Lavaux-Oron",
                                                "Appenzell i.Rh." =
                                                  "Appenzell Innerrhoden")))

# Round 3
merged_data_final <- merged_data_final %>%
  mutate(District = str_remove_all(District, prefix_patterns_1),
         District = str_replace_all(District, c("See / District du Lac" = "See / Lac")))

crime_20 <- crime_20 %>%
  mutate(District = str_remove_all(District, prefix_patterns_2),
         District = str_replace_all(District, c("Lavau0-Oron" = "Lavaux-Oron",
                                                "Appenzell i.Rh." =
                                                  "Appenzell Innerrhoden")))

# Create a mapping vector to make all remaining non-matching district
# match according to the crime_20 syntax
replacements <- c("la Broye" = "La Broye",
                  "la Glâne" = "La Glâne",
                  "la Gruyère" = "La Gruyère",
                  "la Sarine" = "La Sarine",
                  "la Veveyse" = "La Veveyse",
                  "Engiadina Bassa / Val Müstair" = "Engiadina Bassa/Val Müstair",
                  "Prättigau / Davos" = "Prättigau/Davos",
                  "la Broye-Vully" = "Broye-Vully",
                  "la Riviera-Pays-d'Enhaut" = "Riviera-Pays-d'Enhaut",
                  "Canton de Neuchâtel" = "Cant. de Neuchâtel",
                  "Canton de Genève" = "Cant. de Genève",
                  "des Franches-Montagnes" = "Les Franches-Montagnes")

# Correct district names
merged_data_final$District <- recode(merged_data_final$District, !!!replacements)

# Perform a left join to add crime rates to each municipality
merged_data_final <- left_join(merged_data_final, crime_20, by = "District")

# Check the first few rows of the updated dataset to confirm the join
head(merged_data_final)
```

```
##   Canton  District bfs_nr_old Municipality_Name Area_Population_21
```

```
## 1 Zürich Affoltern          1           Aeugst               1988
## 2 Zürich Affoltern          2         Affoltern              12404
## 3 Zürich Affoltern          3        Bonstetten               5632
## 4 Zürich Affoltern          4            Hausen               3852
## 5 Zürich Affoltern          5          Hedingen               3780
## 6 Zürich Affoltern          6            Kappel               1290
##   total_foreigners_21 Arab_Pop_21 Muslim_Pop_21 Turkish_Pop_21
## 1                 295           3            12              4
## 2                3631          94           253             60
## 3                 993          26            68              8
## 4                 607           7            39             11
## 5                 689          21            68             25
## 6                 212           5            16              5
##   Area_Population_10 total_foreigners_10 Arab_Pop_10 Muslim_Pop_10
## 1               1824                 225           1             8
## 2              11091                2832          35           138
## 3               5159                 657           3            15
## 4               3360                 391           5            19
## 5               3410                 451           8            25
## 6                918                 102           0             1
##   Turkish_Pop_10 Total_property_related_crimes Violent_offences
## 1              1                          21.3              4.7
## 2             54                          21.3              4.7
## 3              8                          21.3              4.7
## 4             11                          21.3              4.7
## 5              3                          21.3              4.7
## 6              0                          21.3              4.7
##   Total_criminal_offenses
## 1                    38.3
## 2                    38.3
## 3                    38.3
## 4                    38.3
## 5                    38.3
## 6                    38.3
```

```r
# Check NAs
sum(is.na(merged_data_final))
```

```
## [1] 0
```

```r
# No NAs
```

End part 2

Part 3: Add the Linguistic Regions of each Municipality, merging by bfs__nr__old

```r
# Merge the data sets by 'bfs_nr_old'
merged_data_final <- merge(x = Region_22[, c("bfs_nr_old", "Linguistic_region")],
```

```r
                              y = merged_data_final, by = "bfs_nr_old", all.y = TRUE)

# Reordering columns to make 'Linguistic_region' the first column
merged_data_final <- merged_data_final[c("Linguistic_region",
                                          setdiff(names(merged_data_final),
                                                  "Linguistic_region"))]

# View the top rows of the updated dataset and check NAs
head(merged_data_final)
```

```
##   Linguistic_region bfs_nr_old Canton  District Municipality_Name
## 1            German          1 Zürich Affoltern            Aeugst
## 2            German          2 Zürich Affoltern         Affoltern
## 3            German          3 Zürich Affoltern        Bonstetten
## 4            German          4 Zürich Affoltern            Hausen
## 5            German          5 Zürich Affoltern          Hedingen
## 6            German          6 Zürich Affoltern            Kappel
##   Area_Population_21 total_foreigners_21 Arab_Pop_21 Muslim_Pop_21
## 1               1988                 295           3            12
## 2              12404                3631          94           253
## 3               5632                 993          26            68
## 4               3852                 607           7            39
## 5               3780                 689          21            68
## 6               1290                 212           5            16
##   Turkish_Pop_21 Area_Population_10 total_foreigners_10 Arab_Pop_10
## 1              4               1824                 225           1
## 2             60              11091                2832          35
## 3              8               5159                 657           3
## 4             11               3360                 391           5
## 5             25               3410                 451           8
## 6              5                918                 102           0
##   Muslim_Pop_10 Turkish_Pop_10 Total_property_related_crimes Violent_offences
## 1             8              1                          21.3              4.7
## 2           138             54                          21.3              4.7
## 3            15              8                          21.3              4.7
## 4            19             11                          21.3              4.7
## 5            25              3                          21.3              4.7
## 6             1              0                          21.3              4.7
##   Total_criminal_offenses
## 1                    38.3
## 2                    38.3
## 3                    38.3
## 4                    38.3
## 5                    38.3
## 6                    38.3
```

```r
sum(is.na(merged_data_final))
```

```
## [1] 0
```

```r
# All good

# For efficiency in the next step,rename crime variables
colnames(merged_data_final)[colnames(merged_data_final)
                            == "Total_property_related_crimes"] <-
  "Property_related_crimes"
colnames(merged_data_final)[colnames(merged_data_final)
                            == "Total_criminal_offenses"] <-
  "Tot_criminal_offenses"

# Update merged_data_final to the 2024 spatial reference and calculate
# population shares and percentages

# Update merged_data_final using the 2024 MT:
merged_data_final24 <- left_join(mapping_table_2022, merged_data_final,
                                 by = "bfs_nr_old") %>%
  group_by(bfs_nr_new) %>%
  summarise(
    Linguistic_Region = first(Linguistic_region),# Preserve the first linguistic
    # region entry as all mergers are within such regions
    District = first(District), # Similarly for districts
    Canton = first(Canton),  # Similarly for canton
    Total_Population_21 = sum(Area_Population_21, na.rm = TRUE),
    Total_Foreigners_21 = sum(total_foreigners_21, na.rm = TRUE),
    Total_Arab_Population_21 = sum(Arab_Pop_21, na.rm = TRUE),
    Total_Muslim_Population_21 = sum(Muslim_Pop_21, na.rm = TRUE),
    Total_Turkish_Population_21 = sum(Turkish_Pop_21, na.rm = TRUE),
    Total_Population_10 = sum(Area_Population_10, na.rm = TRUE),
    Total_Foreigners_10 = sum(total_foreigners_10, na.rm = TRUE),
    Total_Arab_Population_10 = sum(Arab_Pop_10, na.rm = TRUE),
    Total_Muslim_Population_10 = sum(Muslim_Pop_10, na.rm = TRUE),
    Total_Turkish_Population_10 = sum(Turkish_Pop_10, na.rm = TRUE),
    Violent_offences = first(Violent_offences), # Crime is at district level
    Property_related_crimes = first(Property_related_crimes),# idem
    Tot_criminal_offenses = first(Tot_criminal_offenses), # idem
  ) %>%
  mutate(
    # Calculate shares for 2021
    Tot_foreigner_share_21 = (Total_Foreigners_21 / Total_Population_21) * 100,
    Arab_share_21 = (Total_Arab_Population_21 / Total_Population_21) * 100,
    Muslim_share_21 = (Total_Muslim_Population_21 / Total_Population_21) * 100,
    NM_Foreigner_share_21 = (Tot_foreigner_share_21 - Muslim_share_21),
    MNA_share_21 = (Muslim_share_21 - Arab_share_21),
    Turkish_share_21 = (Total_Turkish_Population_21 / Total_Population_21) * 100,

    # Calculate shares for 2010
    Tot_foreigner_share_10 = (Total_Foreigners_10 / Total_Population_10) * 100,
    Arab_share_10 = (Total_Arab_Population_10 / Total_Population_10) * 100,
    Muslim_share_10 = (Total_Muslim_Population_10 / Total_Population_10) * 100,
    NM_Foreigner_share_10 = (Tot_foreigner_share_10 - Muslim_share_10),
    MNA_share_10 = (Muslim_share_10 - Arab_share_10),
    Turkish_share_10 = (Total_Turkish_Population_10 / Total_Population_10) * 100,
```

```r
    # Calculate changes between 2021 and 2010
    Change_foreigners = Tot_foreigner_share_21 - Tot_foreigner_share_10,
    Change_arabs = Arab_share_21 - Arab_share_10,
    Change_Muslims = Muslim_share_21 - Muslim_share_10,
    Change_NM_foreigners = NM_Foreigner_share_21 - NM_Foreigner_share_10,
    Change_MNA = MNA_share_21 - MNA_share_10,
    Change_turkish = Turkish_share_21 - Turkish_share_10
  )%>%
  select(-starts_with("Total_"), -Canton, -Tot_foreigner_share_21, -Arab_share_21,
         -Muslim_share_21, -NM_Foreigner_share_21, -MNA_share_21,
         -Tot_foreigner_share_10, Turkish_share_21)


# Merge the above data set with the municipality types' data set
merged_data_final24 <- left_join(merged_data_final24, Municipality_type_24,
                                 by = "bfs_nr_new")

# Reorder the columns
merged_data_final24 <- merged_data_final24 %>%
  select(bfs_nr_new, Municipality_name, Municipality_types,
         District, Cantonal_affiliation, Linguistic_Region, everything(),
         -c(Violent_offences, Property_related_crimes, Tot_criminal_offenses),
         Violent_offences, Property_related_crimes, Tot_criminal_offenses)

# Rename columns to easier names:
colnames(merged_data_final24)[colnames(merged_data_final24) ==
                                "Municipality_types"] <- "Municipality_type"
colnames(merged_data_final24)[colnames(merged_data_final24) ==
                                "Cantonal_affiliation"] <- "Canton"


# Simplify the municipality type categories
merged_data_final24 <- merged_data_final24 %>%
  mutate(Municipality_type = case_when(
    Municipality_type %in% c("Urban municipality of a large conurbation ",
                             "Urban community in a medium-sized conurbation ",
                             "Small urban community or outside urban area ") ~ "Urban",
    Municipality_type %in% c("Low-density suburban community ",
                             "Medium-density suburban community ",
                             "High-density suburban community ") ~ "Semi-Urban",
    Municipality_type %in% c("Centrally located rural municipality ",
                             "Municipality in a rural center ",
                             "Rural outskirts ") ~ "Rural"))

# verify this worked:
unique(merged_data_final24$Municipality_type)
```

```
## [1] "Semi-Urban" "Urban"      "Rural"
```

```r
# All good.
```

End part 3

Part 4: Cleaning and Formatting the Referendum Data:

```r
# Removing the Swiss level data in the first row from both data sets
Minaret <- subset(Minaret, Canton != " Suisse  ")
FaceBan <- subset(FaceBan, Canton != "Suisse")

# Resetting row names to start from 1
rownames(Minaret) <- NULL
rownames(FaceBan) <- NULL

# Simultaneously cleaning and updating both data sets to 2024 boundaries,
# as well as calculating referendums' percentage of yes votes and participation rate:

# Minaret:
Minaret_updated <- left_join(mapping_table_vote2009, Minaret, by = "bfs_nr_old") %>%
  mutate(Yes_cleaned = as.numeric(gsub("[[:punct:][:space:]]", "", Yes)),
         # Convert 'Yes' column to numeric, remove punctuation
         No_cleaned = as.numeric(gsub("[[:punct:][:space:]]", "", No)),
         # Same process repeated
         Registered.voters_cleaned = as.numeric(gsub("[[:punct:][:space:]]",
                                                     "", Registered.voters)),
         Votes.received_cleaned = as.numeric(gsub("[[:punct:][:space:]]", "",
                                                  Votes.received))) %>%
  group_by(bfs_nr_new) %>%
  summarise(Minaret_yes = sum(Yes_cleaned, na.rm = TRUE),
            Minaret_no = sum(No_cleaned, na.rm = TRUE),
            Minaret_registered_voters = sum(Registered.voters_cleaned, na.rm = TRUE),
            Minaret_votes_received = sum(Votes.received_cleaned, na.rm = TRUE),
            .groups = "drop") %>%
  mutate(Minaret_yes_percent = Minaret_yes / (Minaret_yes + Minaret_no) * 100,
         Minaret_participation = (Minaret_votes_received /
                                    Minaret_registered_voters)*100)


# Face ban:
FaceBan_updated <- left_join(mapping_table_vote2021, FaceBan, by = "bfs_nr_old") %>%
  mutate(FaceBan_yes = as.numeric(gsub("[[:punct:][:space:]]", "", Yes)),
         # Convert 'Yes' column to numeric, remove punctuation
         FaceBan_no = as.numeric(gsub("[[:punct:][:space:]]", "", No)),
         # Same process repeated
         FaceBan_registered.voters = as.numeric(gsub("[[:punct:][:space:]]", "",
                                                     Registered.voters)),
         FaceBan_votes.received = as.numeric(gsub("[[:punct:][:space:]]", "",
                                                  Votes.received))) %>%
  group_by(bfs_nr_new) %>%
  summarise(FaceBan_yes = sum(FaceBan_yes, na.rm = TRUE),
            FaceBan_no = sum(FaceBan_no, na.rm = TRUE),
            FaceBan_registered.voters = sum(FaceBan_registered.voters, na.rm = TRUE),
            FaceBan_votes.received = sum(FaceBan_votes.received, na.rm = TRUE),
            .groups = "drop") %>%
```

```r
  mutate(FaceBan_yes_percent = FaceBan_yes / (FaceBan_yes + FaceBan_no) * 100,
         FaceBan_participation = (FaceBan_votes.received /
                                     FaceBan_registered.voters) * 100)

# We get the right number of municipalities for both the 2009 and 2021 data sets.


# Investigate if there are any unexpected NAs in the updated data sets

# Starting with the Minaret data
sum(is.na(Minaret_updated)) # 12
```

```
## [1] 12
```

```r
# Finding where they are
Minaret_updated[is.na(Minaret_updated$bfs_nr_new), ] # 0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <dbl>, Minaret_yes <dbl>, Minaret_no <dbl>,
## #   Minaret_registered_voters <dbl>, Minaret_votes_received <dbl>,
## #   Minaret_yes_percent <dbl>, Minaret_participation <dbl>
```

```r
Minaret_updated[is.na(Minaret_updated$Minaret_yes), ] # 0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <dbl>, Minaret_yes <dbl>, Minaret_no <dbl>,
## #   Minaret_registered_voters <dbl>, Minaret_votes_received <dbl>,
## #   Minaret_yes_percent <dbl>, Minaret_participation <dbl>
```

```r
Minaret_updated[is.na(Minaret_updated$Minaret_no), ] # 0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <dbl>, Minaret_yes <dbl>, Minaret_no <dbl>,
## #   Minaret_registered_voters <dbl>, Minaret_votes_received <dbl>,
## #   Minaret_yes_percent <dbl>, Minaret_participation <dbl>
```

```r
Minaret_updated[is.na(Minaret_updated$Minaret_registered_voters), ] #0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <dbl>, Minaret_yes <dbl>, Minaret_no <dbl>,
## #   Minaret_registered_voters <dbl>, Minaret_votes_received <dbl>,
## #   Minaret_yes_percent <dbl>, Minaret_participation <dbl>
```

```r
Minaret_updated[is.na(Minaret_updated$Minaret_votes_received), ] # 0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <dbl>, Minaret_yes <dbl>, Minaret_no <dbl>,
## #   Minaret_registered_voters <dbl>, Minaret_votes_received <dbl>,
## #   Minaret_yes_percent <dbl>, Minaret_participation <dbl>
```

```r
Minaret_updated[is.na(Minaret_updated$Minaret_yes_percent), ] # 6
```

```
## # A tibble: 6 x 7
##   bfs_nr_new Minaret_yes Minaret_no Minaret_registered_voters
##        <dbl>       <dbl>      <dbl>                     <dbl>
## 1        389           0          0                         0
## 2        408           0          0                         0
## 3        422           0          0                         0
## 4        535           0          0                         0
## 5        868           0          0                         0
## 6        877           0          0                         0
## # i 3 more variables: Minaret_votes_received <dbl>, Minaret_yes_percent <dbl>,
## #   Minaret_participation <dbl>
```

```r
Minaret_updated[is.na(Minaret_updated$Minaret_participation), ] # 6
```

```
## # A tibble: 6 x 7
##   bfs_nr_new Minaret_yes Minaret_no Minaret_registered_voters
##        <dbl>       <dbl>      <dbl>                     <dbl>
## 1        389           0          0                         0
## 2        408           0          0                         0
## 3        422           0          0                         0
## 4        535           0          0                         0
## 5        868           0          0                         0
## 6        877           0          0                         0
## # i 3 more variables: Minaret_votes_received <dbl>, Minaret_yes_percent <dbl>,
## #   Minaret_participation <dbl>
```

```r
# These are 6 municipalities, whose rows are filled with zeros and 2 NAs at the end

# --> These 6 municipalities are all hamlets with between 50 and 500 citizens.
# We simply do not have data for them.


# Turning to the Face ban data
sum(is.na(FaceBan_updated)) #8
```

```
## [1] 8
```

```r
# Investigating where they are.
FaceBan_updated[is.na(FaceBan_updated$bfs_nr_new), ] # 0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <int>, FaceBan_yes <dbl>, FaceBan_no <dbl>,
## #   FaceBan_registered.voters <dbl>, FaceBan_votes.received <dbl>,
## #   FaceBan_yes_percent <dbl>, FaceBan_participation <dbl>
```

```r
FaceBan_updated[is.na(FaceBan_updated$FaceBan_yes), ] # 0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <int>, FaceBan_yes <dbl>, FaceBan_no <dbl>,
## #   FaceBan_registered.voters <dbl>, FaceBan_votes.received <dbl>,
## #   FaceBan_yes_percent <dbl>, FaceBan_participation <dbl>
```

```
FaceBan_updated[is.na(FaceBan_updated$FaceBan_no), ] # 0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <int>, FaceBan_yes <dbl>, FaceBan_no <dbl>,
## #   FaceBan_registered.voters <dbl>, FaceBan_votes.received <dbl>,
## #   FaceBan_yes_percent <dbl>, FaceBan_participation <dbl>
```

```
FaceBan_updated[is.na(FaceBan_updated$FaceBan_registered.voters), ] # 0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <int>, FaceBan_yes <dbl>, FaceBan_no <dbl>,
## #   FaceBan_registered.voters <dbl>, FaceBan_votes.received <dbl>,
## #   FaceBan_yes_percent <dbl>, FaceBan_participation <dbl>
```

```
FaceBan_updated[is.na(FaceBan_updated$FaceBan_votes.received), ] # 0
```

```
## # A tibble: 0 x 7
## # i 7 variables: bfs_nr_new <int>, FaceBan_yes <dbl>, FaceBan_no <dbl>,
## #   FaceBan_registered.voters <dbl>, FaceBan_votes.received <dbl>,
## #   FaceBan_yes_percent <dbl>, FaceBan_participation <dbl>
```

```
FaceBan_updated[is.na(FaceBan_updated$FaceBan_yes_percent), ] # 4
```

```
## # A tibble: 4 x 7
##   bfs_nr_new FaceBan_yes FaceBan_no FaceBan_registered.voters
##        <int>       <dbl>      <dbl>                     <dbl>
## 1        389           0          0                         0
## 2        408           0          0                         0
## 3        535           0          0                         0
## 4        877           0          0                         0
## # i 3 more variables: FaceBan_votes.received <dbl>, FaceBan_yes_percent <dbl>,
## #   FaceBan_participation <dbl>
```

```
FaceBan_updated[is.na(FaceBan_updated$FaceBan_participation), ] # 4
```

```
## # A tibble: 4 x 7
##   bfs_nr_new FaceBan_yes FaceBan_no FaceBan_registered.voters
##        <int>       <dbl>      <dbl>                     <dbl>
## 1        389           0          0                         0
## 2        408           0          0                         0
## 3        535           0          0                         0
## 4        877           0          0                         0
## # i 3 more variables: FaceBan_votes.received <dbl>, FaceBan_yes_percent <dbl>,
## #   FaceBan_participation <dbl>
```

```r
# Idem as above, representing 4/6 of the missing municipalities in the Minaret data


# Merging the referendum data sets on bfs_nr_new
Referendum_data <- merge(Minaret_updated, FaceBan_updated, by = "bfs_nr_new")

# Calculate the change in support
Referendum_data$Yes_percent_change <-
  Referendum_data$FaceBan_yes_percent - Referendum_data$Minaret_yes_percent

# Calculate the change in turnout
Referendum_data$Vote_turnout_change <-
  Referendum_data$FaceBan_participation - Referendum_data$Minaret_participation
```

End of part 4.

Part 5: Formatting the Population Density Data:

```r
# No additional cleaning is necessary for these data sets.

# Updating the 2009 data:
Pop_density_09_updated <- left_join(mapping_table_cov2009,
                                    Pop_density_09, by = "bfs_nr_old") %>%
  mutate(Total_population_cleaned = as.numeric(gsub("[[:punct:][:space:]]", "",
                                                    Total_population))) %>%
  #Convert 'Total_population' column to numeric, remove punctuation
  group_by(bfs_nr_new) %>%
  summarise(Total_pop_09 = sum(Total_population_cleaned, na.rm = TRUE),
            # calculate the total pop for each "new" municipality
            Municipality_area_09 = sum(Municipality_total_km2_area, na.rm = TRUE),
            # calculate the area for each "new" municipality
            .groups = "drop") %>%
  mutate(Population_density_09 = (Total_pop_09 / Municipality_area_09))


# Updating the 2020 data:
Pop_density_20_updated <- left_join(mapping_table_cov2020, Pop_density_20,
                                    by = "bfs_nr_old") %>%
  mutate(Total_population_cleaned = as.numeric(gsub("[[:punct:][:space:]]", "",
                                                    Total_population))) %>%
  #Convert 'Total_population' column to numeric, remove punctuation
  group_by(bfs_nr_new) %>%
  summarise(Total_pop_20 = sum(Total_population_cleaned, na.rm = TRUE),
            # calculate the total pop for each "new" municipality
            Municipality_area_20 = sum(Municipality_total_km2_area, na.rm = TRUE),
            # calculate the area for each "new" municipality
            .groups = "drop") %>%
  mutate(Population_density_20 = (Total_pop_20 / Municipality_area_20))

# We get 2145 municipalities, as desired.
```

```r
# Investigating if there are any unexpected NAs in the updated data sets:

# Starting with 2009:
sum(is.na(Pop_density_09_updated$bfs_nr_new)) # 0
```

```
## [1] 0
```

```r
sum(is.na(Pop_density_09_updated$Total_pop_09)) # 0
```

```
## [1] 0
```

```r
sum(is.na(Pop_density_09_updated$Municipality_area_09)) # 0
```

```
## [1] 0
```

```r
sum(is.na(Pop_density_09_updated$Population_density_09)) # 0
```

```
## [1] 0
```

```r
Pop_density_09_updated[is.na(Pop_density_09_updated$Population_density_09), ] # 0
```

```
## # A tibble: 0 x 4
## # i 4 variables: bfs_nr_new <dbl>, Total_pop_09 <dbl>,
## #   Municipality_area_09 <dbl>, Population_density_09 <dbl>
```

```r
# None.

# Turning to 2020:
sum(is.na(Pop_density_20_updated$bfs_nr_new))
```

```
## [1] 0
```

```r
sum(is.na(Pop_density_20_updated$Total_pop_20))
```

```
## [1] 0
```

```r
sum(is.na(Pop_density_20_updated$Municipality_area_20))
```

```
## [1] 0
```

```r
sum(is.na(Pop_density_20_updated$Population_density_20))
```

```
## [1] 0
```

```r
# None again. This reflects the lack of NA in both the MTs and original data.


# Merging the population density data sets on bfs_nr_new
Pop_density_merged <- merge(Pop_density_09_updated, Pop_density_20_updated,
                            by = "bfs_nr_new")

# Calculating the change in population density
Pop_density_merged$Pop_density_change <- Pop_density_merged$Population_density_20 -
  Pop_density_merged$Population_density_09

# Converting population density to hundreds of persons/km² for easier
# interpretation of model coefficients
Pop_density_merged$Population_density_09_in_hundreds <-
  Pop_density_merged$Population_density_09 / 100
Pop_density_merged$Population_density_20_in_hundreds <-
  Pop_density_merged$Population_density_20 / 100

# Calculating the change in population density in hundreds of persons/km²
Pop_density_merged$Pop_density_change_in_hundreds <-
  Pop_density_merged$Population_density_20_in_hundreds -
  Pop_density_merged$Population_density_09_in_hundreds
```

End of part 5

Part 6: Formatting the Social Help Data:

This data was not included in the models and is removed from the final dataset in section 9

```r
# To preserve the privacy of residents, the number of recipients
# is not available for very small municipalities.
# There, it is denoted by X. Accordingly, data cleaning is necessary prior to
# updating to the 2024 municipality boundaries.

# For 2009: removing punctuation and spaces in "Number_of_recipients" and
# "Total_population" columns and converting data to numeric
# (from character initially). This automatically makes the 917 "X" and the 1 "-" NAs.
Social_help_09$Number_of_recipients <- as.numeric(gsub("[[:punct:][:space:]]", "",
                                             Social_help_09$Number_of_recipients))
```

```
## Warning: NAs introduced by coercion
```

```r
Social_help_09$Total_population <- as.numeric(gsub("[[:punct:][:space:]]", "",
                                              Social_help_09$Total_population))

# Checking if NAs are as expected.
sum(is.na(Social_help_09$Number_of_recipients)) # 918 NAs = total number of X and -
```

```
## [1] 918
```

```r
# Looking at the type of municipality for which we do not have access to data:
mean(Social_help_09$Total_population[is.na(Social_help_09$Number_of_recipients)],
     na.rm = TRUE) # 477.62 residents: the villages with X's are indeed very small.
```

```
## [1] 477.6209
```

```r
# For 2020, clean the data similarly
Social_help_20$Number_of_recipients <- as.numeric(gsub("[[:punct:][:space:]]", "",
                                           Social_help_20$Number_of_recipients))
```

```
## Warning: NAs introduced by coercion
```

```r
Social_help_20$Total_population <- as.numeric(gsub("[[:punct:][:space:]]", "",
                                              Social_help_20$Total_population))

# Checking if NAs are as expected.
sum(is.na(Social_help_20$Number_of_recipients)) # 510 NAs
```

```
## [1] 510
```

```r
# Looking at NA municipality types.
mean(Social_help_20$Total_population[is.na(Social_help_20$Number_of_recipients)],
     na.rm = TRUE) # 575.2392: still very small municipalities, although
```

```
## [1] 575.2392
```

```r
# slightly larger. This could be because data in municipalities which have merged
# is kept secret anymore.
# The lower number of municipalities gives this credence, especially since the
# difference in NAs is approximates the number of mergers between periods.


# Updating the 2009 data:
Social_help_09_updated <- left_join(mapping_table_cov2009, Social_help_09,
                                    by = "bfs_nr_old") %>%
  group_by(bfs_nr_new) %>%
  summarise(Total_population_09 = sum(Total_population, na.rm = TRUE),
            # calculate the total pop for each "new" municipality
            Number_of_recipients_09 = sum(Number_of_recipients, na.rm = TRUE),
            # calculate the number of recipients for each "new" municipality
            .groups = "drop") %>%
  mutate(Percent_social_help_09 = (Number_of_recipients_09 / Total_population_09)*100)


# Updating the 2020 data:
Social_help_20_updated <- left_join(mapping_table_cov2020, Social_help_20,
                                    by = "bfs_nr_old") %>%
  group_by(bfs_nr_new) %>%
  summarise(Total_population_20 = sum(Total_population, na.rm = TRUE),
            # calculate the total pop for each "new" municipality
```

```
            Number_of_recipients_20 = sum(Number_of_recipients, na.rm = TRUE),
            # calculate the number of recipients for each "new" municipality
            .groups = "drop") %>%
  mutate(Percent_social_help_20 = (Number_of_recipients_20 /
                                     Total_population_20)*100)


# We now know that some NAs are inadvertently transformed to zeros.
# We investigate their possible presence:
sum(Social_help_09_updated$Number_of_recipients_09==0)
```

```
## [1] 578
```

```
# The initial 918 NA municipalities translate to 578 (new) ones with zero recipient.
sum(Social_help_09_updated$Percent_social_help_09==0)
```

```
## [1] 578
```

```
# And thus, zero percent of social help in these municipalities.

# We know these zeros are errors as no municipality had zero social help.
# Accordingly, we convert them back to NAs.
Social_help_09_updated <- Social_help_09_updated %>%
  mutate(
    Number_of_recipients_09 = if_else(Number_of_recipients_09 == 0, NA_real_,
                                      Number_of_recipients_09),
    Percent_social_help_09 = if_else(Percent_social_help_09 == 0, NA_real_,
                                     Percent_social_help_09)
  )

# Safety checks
Social_help_09_updated[is.na(Social_help_09_updated$Total_population_09), ]
```

```
## # A tibble: 0 x 4
## # i 4 variables: bfs_nr_new <dbl>, Total_population_09 <dbl>,
## #   Number_of_recipients_09 <dbl>, Percent_social_help_09 <dbl>
```

```
# Did not change unwanted columns
Social_help_09_updated[is.na(Social_help_09_updated$bfs_nr_new), ]
```

```
## # A tibble: 0 x 4
## # i 4 variables: bfs_nr_new <dbl>, Total_population_09 <dbl>,
## #   Number_of_recipients_09 <dbl>, Percent_social_help_09 <dbl>
```

```
# Did not change unwanted columns
sum(is.na(Social_help_09_updated$Number_of_recipients_09)) # 578
```

```
## [1] 578
```

```r
sum(is.na(Social_help_09_updated$Percent_social_help_09)) # 578
```

```
## [1] 578
```

```r
# Everything is in order.


# Going through the same process for the 2020 data
sum(Social_help_20_updated$Number_of_recipients_20==0)
```

```
## [1] 473
```

```r
# The initial 510 NA municipalities translate to 473
sum(Social_help_20_updated$Percent_social_help_20==0) # idem.
```

```
## [1] 473
```

```r
# Converting the zeros back to NAs.
Social_help_20_updated <- Social_help_20_updated %>%
  mutate(
    Number_of_recipients_20 = if_else(Number_of_recipients_20 == 0, NA_real_,
                                      Number_of_recipients_20),
    Percent_social_help_20 = if_else(Percent_social_help_20 == 0, NA_real_,
                                     Percent_social_help_20)
  )

# Safety checks
Social_help_20_updated[is.na(Social_help_20_updated$Total_population_20), ] # 0
```

```
## # A tibble: 0 x 4
## # i 4 variables: bfs_nr_new <int>, Total_population_20 <dbl>,
## #   Number_of_recipients_20 <dbl>, Percent_social_help_20 <dbl>
```

```r
Social_help_20_updated[is.na(Social_help_20_updated$bfs_nr_new), ] # 0
```

```
## # A tibble: 0 x 4
## # i 4 variables: bfs_nr_new <int>, Total_population_20 <dbl>,
## #   Number_of_recipients_20 <dbl>, Percent_social_help_20 <dbl>
```

```r
sum(is.na(Social_help_20_updated$Number_of_recipients_20)) # 473
```

```
## [1] 473
```

```r
sum(is.na(Social_help_20_updated$Percent_social_help_20)) # 473
```

```
## [1] 473
```

```r
# All good.


# Merging the social help data sets on bfs_nr_new
social_help_merged <- merge(Social_help_09_updated, Social_help_20_updated,
                            by = "bfs_nr_new")

# Calculating the change in social help
social_help_merged$Social_help_change <- social_help_merged$Percent_social_help_20 -
  social_help_merged$Percent_social_help_09
```

End of part 6

Part 7: Formatting the Average Income per Taxpayer Data:

```r
# Some minor initial cleaning is necessary:

# For some reason, the third 2010 column name has a period at the end; correcting it:
colnames(Average_income_10)[colnames(Average_income_10) ==
                              "Municipality_total_revenue."] <-
  "Municipality_total_revenue"
sum(is.na(Average_income_20$Number_of_taxpayers))
```

```
## [1] 75
```

```r
# 75, remains minimal (will be identical for Municipality revenue)


# Updating the 2010 data:
Average_income_10_updated <- left_join(mapping_table_cov2010,
                                       Average_income_10, by = "bfs_nr_old") %>%
  group_by(bfs_nr_new) %>%
  summarise(Number_of_taxpayers_10 = sum(Number_of_taxpayers,
                                         na.rm = TRUE),
            # calculate the number of taxpayers for each "new" municipality
            Municipality_total_revenue_10 = sum(Municipality_total_revenue,
                                                na.rm = TRUE),
            # calculate the total revenue of taxpayers for each "new" municipality
            .groups = "drop") %>%
  mutate(Average_net_income_10 = (Municipality_total_revenue_10 /
                                    Number_of_taxpayers_10))


# Updating the 2020 data:
Average_income_20_updated <- left_join(mapping_table_cov2020,
                                       Average_income_20, by = "bfs_nr_old") %>%
  group_by(bfs_nr_new) %>%
  summarise(Number_of_taxpayers_20 = sum(Number_of_taxpayers, na.rm = TRUE),
            # calculate the number of taxpayers for each "new" municipality
```

```r
            Municipality_total_revenue_20 = sum(Municipality_total_revenue,
                                                na.rm = TRUE),
            # calculate the total revenue of taxpayers for each "new" municipality
            .groups = "drop") %>%
  mutate(Average_net_income_20 = (Municipality_total_revenue_20 /
                                  Number_of_taxpayers_20))



# Investigate zeros and NAs - 2010
sum(is.na(Average_income_10_updated))
```

```
## [1] 0
```

```r
sum(Average_income_20_updated==0)
```

```
## [1] NA
```

```r
# Nothing


# Investigate zeros and NAs - 2020
sum(is.na(Average_income_20_updated$bfs_nr_new)) # 0
```

```
## [1] 0
```

```r
sum(is.na(Average_income_20_updated$Number_of_taxpayers_20)) # 0
```

```
## [1] 0
```

```r
sum(Average_income_20_updated$Number_of_taxpayers_20==0) # 64 NAs transformed to 0
```

```
## [1] 64
```

```r
sum(is.na(Average_income_20_updated$Municipality_total_revenue_20)) # 0
```

```
## [1] 0
```

```r
sum(Average_income_20_updated$Municipality_total_revenue_20==0) # 64, idem
```

```
## [1] 64
```

```r
sum(is.na(Average_income_20_updated$Average_net_income_20)) # 64
```

```
## [1] 64
```

```r
# By precaution, we convert the zeros back to NAs:
Average_income_20_updated <- Average_income_20_updated %>%
  mutate(
    Number_of_taxpayers_20 = if_else(Number_of_taxpayers_20 == 0, NA_real_,
                                     Number_of_taxpayers_20),
    Municipality_total_revenue_20 = if_else(Municipality_total_revenue_20 == 0,
                                            NA_real_, Municipality_total_revenue_20)
  )

# Safety checks
sum(is.na(Average_income_20_updated$Number_of_taxpayers_20)) # 64
```

```
## [1] 64
```

```r
sum(is.na(Average_income_20_updated$Municipality_total_revenue_20)) # 64
```

```
## [1] 64
```

```r
# Successful NA conversion


# Merging the income data sets on bfs_nr_new
income_merged <- merge(Average_income_10_updated, Average_income_20_updated,
                       by = "bfs_nr_new")

# Calculating the change in income
income_merged$income_change <- income_merged$Average_net_income_20 -
  income_merged$Average_net_income_10

# Calculating the change in income in thousands as single francs
# are of reduced relecance as a unit of analysis.
income_merged$income_change_thousands <- income_merged$income_change/1000
```

End of part 7

Part 8: Formatting the SVP Support Data:

```r
# Some initial cleaning is required.

# Cleaning the 2007 data
SVP_support_07$total_votes_07 <- as.numeric(gsub("[[:punct:][:space:]]", "",
                                                 SVP_support_07$total_votes_07))
SVP_support_07$SVP_votes_07 <- as.numeric(gsub("[[:punct:][:space:]]", "",
                                               SVP_support_07$SVP_votes_07))
# NAs where tacit vote or no candidate

# Cleaning the 2019 data
SVP_support_19$total_votes_19 <- as.numeric(gsub("[[:punct:][:space:]]", "",
```

```r
                                                   SVP_support_19$total_votes_19))


# Updating the 2007 vote data:
SVP_support_07_updated <- left_join(mapping_table_SVP2007, SVP_support_07,
                                    by = "bfs_nr_old") %>%
  group_by(bfs_nr_new) %>%
  summarise(SVP_votes_07 = sum(SVP_votes_07, na.rm = TRUE),
            # calculate the number of SVP votes in each "new" municipality
            total_votes_07 = sum(total_votes_07, na.rm = TRUE),
            # calculate the total number of votes in each "new" municipality
            .groups = "drop") %>%
  mutate(SVP_percent_07 = (SVP_votes_07 / total_votes_07)*100)


# Updating the 2019 vote data:
SVP_support_19_updated <- left_join(mapping_table_SVP2019, SVP_support_19,
                                    by = "bfs_nr_old") %>%
  group_by(bfs_nr_new) %>%
  summarise(SVP_votes_19 = sum(SVP_votes_19, na.rm = TRUE),# idem
            total_votes_19 = sum(total_votes_19, na.rm = TRUE), # idem
            .groups = "drop") %>%
  mutate(SVP_percent_19 = (SVP_votes_19 / total_votes_19)*100)

# 2145 municipalities, as expected.


# Investigate zeros and NAs - 2007
sum(is.na(SVP_support_07_updated))
```

```
## [1] 18
```

```r
sum(SVP_support_07_updated==0)
```

```
## [1] NA
```

```r
# Looking into more detail
sum(is.na(SVP_support_07_updated$bfs_nr_new)) # 0
```

```
## [1] 0
```

```r
sum(is.na(SVP_support_07_updated$SVP_votes_07)) # 0
```

```
## [1] 0
```

```r
sum(SVP_support_07_updated$SVP_votes_07==0) # 65
```

```
## [1] 65
```

```r
sum(SVP_support_07_updated$total_votes_07==0) # 18 < 58
```

```
## [1] 18
```

```r
sum(is.na(SVP_support_07_updated$SVP_percent_07)) # 18
```

```
## [1] 18
```

```r
# likely because no candidates/tacit votes are more common in small municipalities
# and that these merged together, so there are now less municipalities with NAs


# Investigate zeros and NAs - 2019
sum(is.na(SVP_support_19_updated))
```

```
## [1] 5
```

```r
sum(SVP_support_19_updated==0)
```

```
## [1] NA
```

```r
# Looking into more detail
sum(is.na(SVP_support_19_updated$bfs_nr_new)) # 0
```

```
## [1] 0
```

```r
sum(is.na(SVP_support_19_updated$SVP_votes_19)) # 0
```

```
## [1] 0
```

```r
sum(SVP_support_19_updated$total_votes_19==0) # 5
```

```
## [1] 5
```

```r
sum(is.na(SVP_support_19_updated$SVP_percent_19)) # 5
```

```
## [1] 5
```

```r
SVP_support_19_updated[is.na(SVP_support_19_updated$total_votes_19), ]
```

```
## # A tibble: 0 x 4
## # i 4 variables: bfs_nr_new <dbl>, SVP_votes_19 <int>, total_votes_19 <dbl>,
## #   SVP_percent_19 <dbl>
```

```r
SVP_support_19_updated[is.na(SVP_support_19_updated$SVP_percent_19), ]
```

```
## # A tibble: 5 x 4
##   bfs_nr_new SVP_votes_19 total_votes_19 SVP_percent_19
##        <dbl>        <int>          <dbl>          <dbl>
## 1        389            0              0            NaN
## 2        408            0              0            NaN
## 3        422            0              0            NaN
## 4        535            0              0            NaN
## 5        877            0              0            NaN
```

```r
# Merging the SVP data sets on bfs_nr_new
SVP_merged <- merge(SVP_support_07_updated, SVP_support_19_updated,
                    by = "bfs_nr_new")

# Calculating the change in SVP support
SVP_merged$SVP_change <- SVP_merged$SVP_percent_19 - SVP_merged$SVP_percent_07
```

End of part 8

All the data sets have now been formatted appropriately. We now combine them:

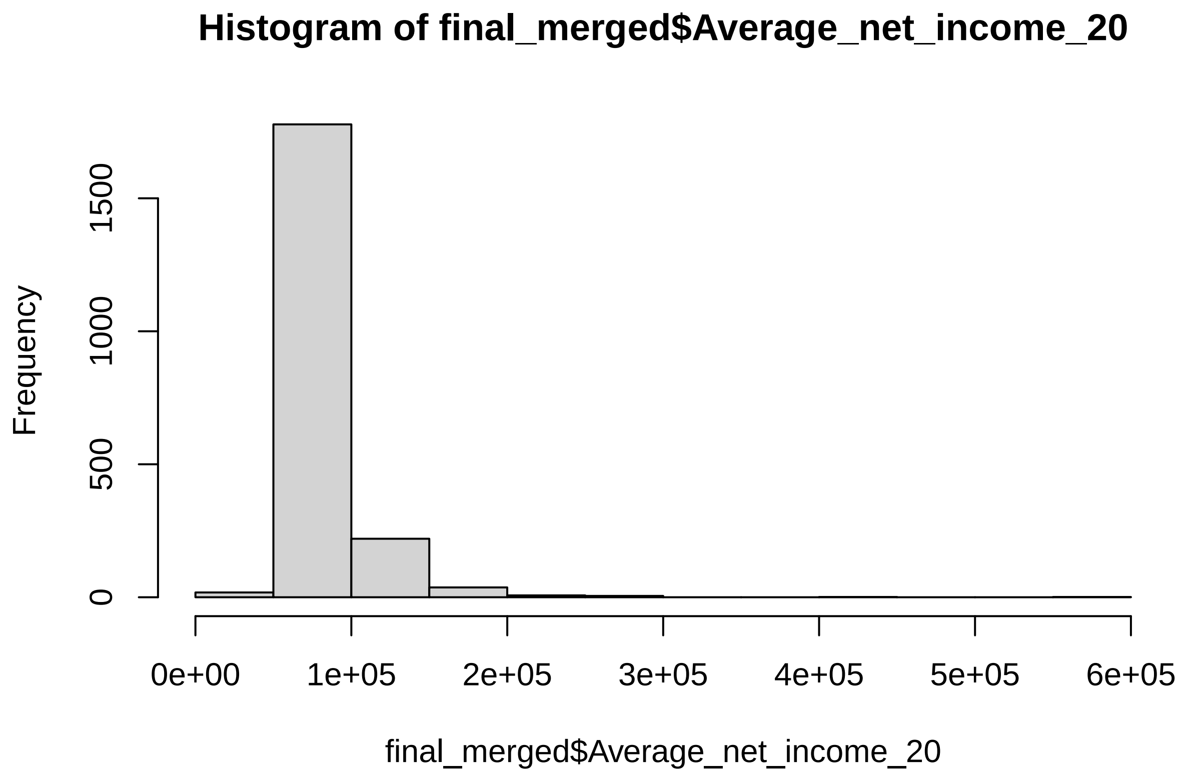
Part 9: Merging all data sets, Imputing NAs, and Removing Unnecessary Variables:

Merging the data sets

```r
# List of data sets to merge
datasets_list <- list(merged_data_final24, Referendum_data, income_merged,
                      social_help_merged, Pop_density_merged, SVP_merged)

# Use Reduce to merge all datasets on 'bfs_nr_new'
final_merged <- Reduce(function(x, y) merge(x, y, by = "bfs_nr_new",
                                            all = TRUE), datasets_list)
```

Imputing NAs

```r
# We impute the NAs to avoid losing information in 2020 income data
hist(final_merged$Average_net_income_20)
```




```r
# The histogram is extremely skewed. To minimise the influence
# of outliers, we impute based on the median average income of municipalities
# of the same type and which are in the same canton, maximising similarities.

# Calculate median income for each combination of Municipality_type and Canton
median_incomes_20 <- final_merged %>%
  group_by(Municipality_type, Canton) %>%
  summarise(Median_Income_20 = median(Average_net_income_20, na.rm = TRUE),
            .groups = 'drop')

# Join the median incomes back to the original dataset for 2020 data
final_merged <- final_merged %>%
  left_join(median_incomes_20, by = c("Municipality_type", "Canton"))

# Impute missing Average_net_income_20 values using Median_Income_20
final_merged <- final_merged %>%
  mutate(Imputed_average_net_income_20 = ifelse(is.na(Average_net_income_20),
                                          Median_Income_20, Average_net_income_20))

# Remove the Median_Income_20 column as no longer needed
final_merged <- select(final_merged, -Median_Income_20)

# Verify that the process worked:
sum(is.na(final_merged$Imputed_average_net_income_20)) # no NAs
```

```
## [1] 0
```

```r
sum(is.na(final_merged$Average_net_income_20)) # The NAs are still there
```

```
## [1] 64
```

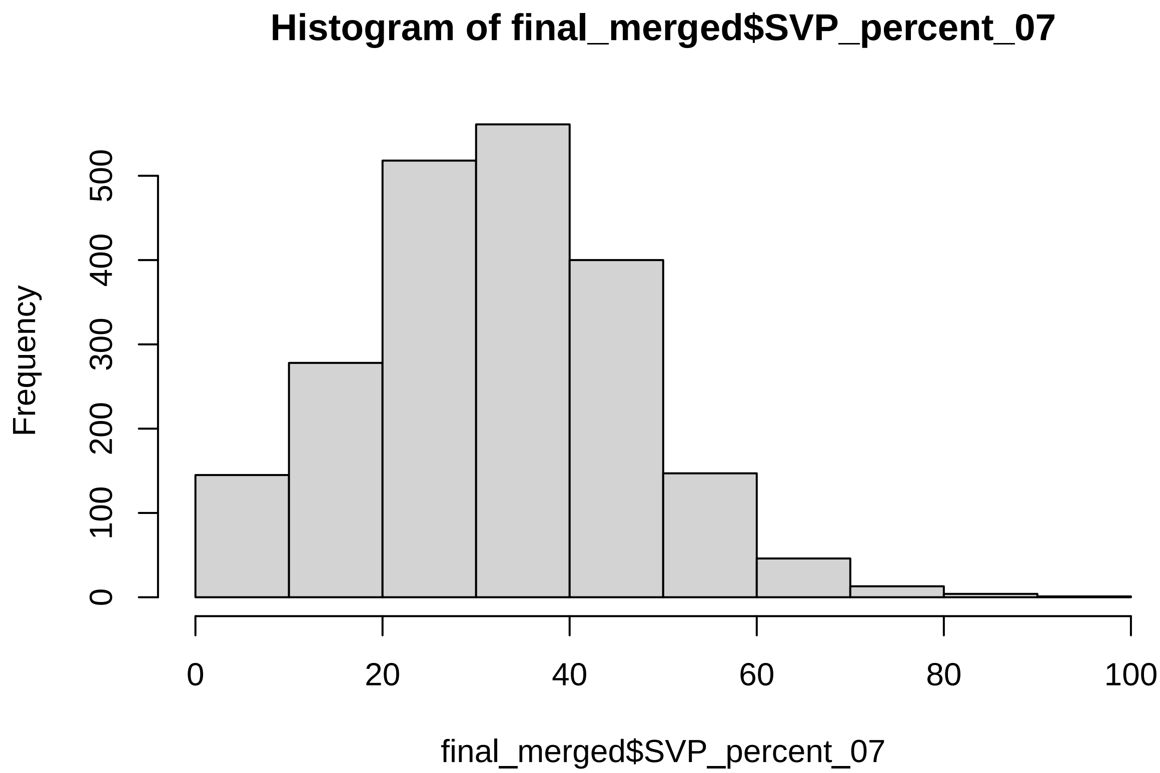
```r
# original average income column



# Calculating row-wise mean for average income between the 2 periods, in anticipation
# of creating sub-group of municipalities with higher and lower incomes

# First for the imputed data, then not imputed
final_merged$Average_net_income_global <-
  rowMeans(final_merged[, c("Average_net_income_10", "Imputed_average_net_income_20")])

final_merged$Average_net_income_global_no_impute <-
  rowMeans(final_merged[, c("Average_net_income_10", "Average_net_income_20")])

# Calculate the median of the Average_net_income_global(s)
median_income <- median(final_merged$Average_net_income_global)

median_income_no_impute <-
  median(final_merged$Average_net_income_global_no_impute, na.rm = TRUE)

# Recode into a binary variable
final_merged$Income_Category <- ifelse(final_merged$Average_net_income_global >
                                        median_income, "rich", "poor")

final_merged$Income_Category_no_impute <-
  ifelse(final_merged$Average_net_income_global_no_impute >
           median_income_no_impute, "rich", "poor")


sum(is.na(final_merged$Income_Category_no_impute)) # 64
```

```
## [1] 64
```

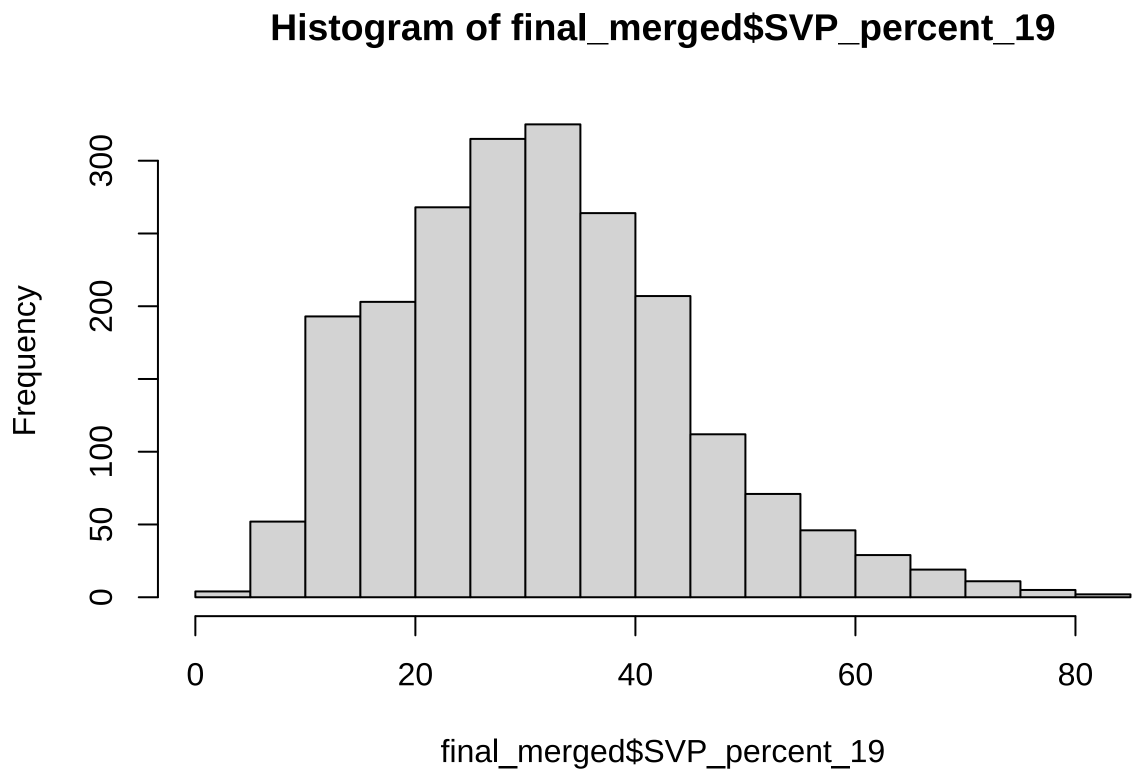
```r
sum(is.na(final_merged$Income_Category)) # 0
```

```
## [1] 0
```

```r
# All good to continue






# Similarly imputing for the SVP support data:
hist(final_merged$SVP_percent_07)
```




```
hist(final_merged$SVP_percent_19)
```




```
# The distributions are more or less normal. We use mean imputation.


# Calculate mean SVP support for each combination of Municipality_type and Canton
means_SVP_07 <- final_merged %>%
  group_by(Municipality_type, Canton) %>%
  summarise(Mean_SVP_07 = mean(SVP_percent_07, na.rm = TRUE), .groups = 'drop')
# All the municipalities in the Nidwald canton are NAs. After looking to impute
# those from its "sibling half-canton", Obwald, we find that there are only 7
# municipalities in Obwald. Accordingly, we impute cantonal means instead.

# Calculate mean SVP support for each Canton. For Nidwald, use Obwald's data
final_merged <- final_merged %>%
  group_by(Canton) %>%
  mutate(
    Mean_SVP_07 = mean(SVP_percent_07, na.rm = TRUE)
  ) %>%
  ungroup()

# Calculate Obwald's mean separately
mean_SVP_Obwald <- mean(final_merged$SVP_percent_07[final_merged$Canton == "Obwald"])

# Apply the Obwald mean to Nidwald
final_merged <- final_merged %>%
  mutate(
    Mean_SVP_07 = ifelse(Canton == "Nidwald" & is.na(Mean_SVP_07),
```

```r
                         mean_SVP_Obwald, Mean_SVP_07))


# Create a new column for imputed SVP_percent_07 values
final_merged <- final_merged %>%
  mutate(
    Imputed_SVP_percent_07 = ifelse(is.na(SVP_percent_07), Mean_SVP_07, SVP_percent_07)
  )%>%
  select(-Mean_SVP_07)  # Remove the temporary mean column after imputation

# Verify that the process worked
sum(is.na(final_merged$Imputed_SVP_percent_07))  # worked
```

```
## [1] 0
```

```r
# Repeat cantonal mean imputation for the 2019 SVP support data (3NAs)
final_merged <- final_merged %>%
  group_by(Canton) %>%
  mutate(
    Mean_SVP_19 = mean(SVP_percent_19, na.rm = TRUE)
  ) %>%
  ungroup()

sum(is.na(final_merged$Mean_SVP_19))
```

```
## [1] 0
```

```r
# Create a new column for imputed SVP_percent_19 values
final_merged <- final_merged %>%
  mutate(
    Imputed_SVP_percent_19 = ifelse(is.na(SVP_percent_19), Mean_SVP_19, SVP_percent_19)
  )%>%
  select(-Mean_SVP_19)  # Remove the temporary mean column after imputation

# Verify that the process worked
sum(is.na(final_merged$Imputed_SVP_percent_19))
```

```
## [1] 0
```

```r
# Re-calculate the change in SVP support account for imputations
final_merged$SVP_change_imputed <- final_merged$Imputed_SVP_percent_19 -
  final_merged$Imputed_SVP_percent_07

# Verify that no NA:
sum(is.na(final_merged$SVP_change_imputed))
```

```
## [1] 0
```

Removing Unwanted Variables

```r
# Remove specified columns from final_merged
final_merged <- select(final_merged, -c(Minaret_yes, Minaret_no,
                                        Minaret_registered_voters,
                                        Minaret_votes_received, FaceBan_yes,
                                        FaceBan_no, FaceBan_yes_percent,
                                        FaceBan_participation,
                                        FaceBan_registered.voters,
                                        FaceBan_votes.received,
                                        Number_of_taxpayers_10,
                                        Municipality_total_revenue_10,
                                        Number_of_taxpayers_20,
                                        Municipality_total_revenue_20,
                                        income_change, Total_population_09,
                                        Number_of_recipients_09,
                                        Percent_social_help_09,
                                        Total_population_20,
                                        Number_of_recipients_20,
                                        Percent_social_help_20, Social_help_change,
                                        Total_pop_09, Municipality_area_09,
                                        Population_density_09,
                                        Total_pop_20, Municipality_area_20,
                                        Population_density_20,
                                        Population_density_09_in_hundreds,
                                        Population_density_20_in_hundreds,
                                        Pop_density_change, SVP_votes_07,
                                        total_votes_07,SVP_votes_19, total_votes_19,
                                        SVP_percent_19, Average_net_income_10,
                                        Average_net_income_20, Turkish_share_21,
                                        SVP_percent_07, Average_net_income_global,
                                        income_change_thousands,
                                        Average_net_income_global_no_impute,
                                        Imputed_SVP_percent_19

))


sum(is.na(final_merged))
```

```
## [1] 106
```

```r
# There are 106 NAs, likely corresponding to the 6 municipalities voting within
# other ones, times the 4 vote variables (=24) + 18 NAs in the non-imputed SVP_change
# + 64 from the non-imputed Income categorical variable.

# We remove the 6 municipalities with no outcome variable, considering they would
# be automatically excluded from any model.

# Identify rows with NAs except, in SVP_change (keeping them for robustness)
rows_to_remove <- apply(final_merged, 1, function(row) {
  # Check if there are any NAs in the row, excluding specified columns
  any(is.na(row[-c(which(names(final_merged) %in%
                           c("SVP_change", "Income_Category_no_impute")))]))
})
```

```r
# Subset the data to remove the identified rows
final_merged <- final_merged[!rows_to_remove, ]

# Check for remaining NAs to confirm
sum(is.na(final_merged))
```

```
## [1] 70
```

```r
# The 6 municipalities removed were also had NAs in the 2 other columns, removing
# a total of 24 + 6 + 6 = 36 NAs.
# This is not surprising considering the reason why data sets lacked
# observations was inherently linked to municipalities' small size.
# We are left with 70 NAs to test if they have
# any effect on estimates as part of robustness checks. Expectations are of
# minimal impact.
```

End of Part 9 and Any Data Cleaning and Wrangling.

Part 10: Assessing Hypothesis H1 - Modelling for all Muslims

Part 10.1: Modelling for all Muslim

Set Up

```r
# Define thresholds for "No Change" as less than 0.05pp. variation in the shares
no_change_lower_bound <- -0.05
no_change_upper_bound <- 0.05

# Calculate medians for the changes outside the "No Change" category
negative_changes <- final_merged$Change_Muslims[final_merged$Change_Muslims <
                                                  no_change_lower_bound]
positive_changes <- final_merged$Change_Muslims[final_merged$Change_Muslims >
                                                  no_change_upper_bound]

median_negative_change <- median(negative_changes, na.rm = TRUE)
median_positive_change <- median(positive_changes, na.rm = TRUE)


# Recode the changes into a categorical variable
final_merged <- final_merged %>%
  mutate(Change_Muslim_Cat = case_when(
    Change_Muslims < median_negative_change ~ "High Decrease",
    Change_Muslims >= median_negative_change & Change_Muslims <
      no_change_lower_bound ~ "Low Decrease",
    Change_Muslims >= no_change_lower_bound & Change_Muslims <=
      no_change_upper_bound ~ "No Change",
```

```r
    Change_Muslims > no_change_upper_bound & Change_Muslims <=
      median_positive_change ~ "Low Increase",
    Change_Muslims > median_positive_change ~ "High Increase"
  ))

# Convert to factor, set "No Change" as reference category, and reorder other levels
final_merged$Change_Muslim_Cat <- factor(final_merged$Change_Muslim_Cat,
                                          levels = c("No Change", "High Increase",
                                                     "Low Increase", "Low Decrease",
                                                     "High Decrease"))
```

Main Specification

```r
# Main Specification
FD_Muslims_Final2 <- lm(Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
                          SVP_change_imputed + Minaret_yes_percent
                        + Muslim_share_10 + NM_Foreigner_share_10 + Violent_offences
                        + Municipality_type + Income_Category, data = final_merged)

summary(FD_Muslims_Final2)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change_imputed + Minaret_yes_percent + Muslim_share_10 +
##     NM_Foreigner_share_10 + Violent_offences + Municipality_type +
##     Income_Category, data = final_merged)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.3905  -3.2595   0.0024   3.5487  20.5457
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      27.39680    1.12409  24.372  < 2e-16 ***
## Change_Muslim_CatHigh Increase -0.98645    0.36860  -2.676 0.007503 **
## Change_Muslim_CatLow Increase  -0.70065    0.35855  -1.954 0.050822 .
## Change_Muslim_CatLow Decrease   0.03997    0.47337   0.084 0.932721
## Change_Muslim_CatHigh Decrease  1.84371    0.51365   3.589 0.000339 ***
## Change_NM_foreigners            0.09980    0.04046   2.467 0.013703 *
## SVP_change_imputed              0.06316    0.01231   5.131 3.14e-07 ***
## Minaret_yes_percent            -0.56481    0.01351 -41.801  < 2e-16 ***
## Muslim_share_10                -1.36013    0.15289  -8.896  < 2e-16 ***
## NM_Foreigner_share_10           0.04485    0.01838   2.439 0.014793 *
## Violent_offences                1.08252    0.07023  15.415  < 2e-16 ***
## Municipality_typeSemi-Urban    -0.83888    0.30444  -2.755 0.005911 **
## Municipality_typeUrban         -1.94618    0.40595  -4.794 1.75e-06 ***
## Income_Categoryrich            -1.84871    0.28173  -6.562 6.66e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.48 on 2111 degrees of freedom
## Multiple R-squared:  0.6015, Adjusted R-squared:  0.599
```

```
## F-statistic: 245.1 on 13 and 2111 DF,  p-value: < 2.2e-16
```

Conclusion: larger increases reduce Islamophobia more, while only large decreases increase it. These results support H1.

For comparison purposes, we also run a model with time-varying controls but no baseline, and one with no time-varying controls nor baselines. In the code, these two models will hereafter be referred to as the 1st and 2nd "simplified models", respectively. Finally, we include a vanilla model with only the change in Muslims as predictor.

1st Simplified Model: Time-Varying Controls but No Baselines

```
FD_Muslims_Final2_simplified1 <- lm(Yes_percent_change ~ Change_Muslim_Cat +
                                      Change_NM_foreigners + SVP_change_imputed +
                                      Violent_offences + Municipality_type +
                                      Income_Category,
                          data = final_merged)

summary(FD_Muslims_Final2_simplified1)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change_imputed + Violent_offences + Municipality_type +
##     Income_Category, data = final_merged)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.769  -5.139  -0.747   4.762  32.603
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -14.07801    0.57405 -24.524  < 2e-16 ***
## Change_Muslim_CatHigh Increase  -1.04258    0.50920  -2.047  0.04073 *
## Change_Muslim_CatLow Increase   -0.50154    0.49666  -1.010  0.31270
## Change_Muslim_CatLow Decrease    0.13926    0.65351   0.213  0.83128
## Change_Muslim_CatHigh Decrease  -0.16175    0.65748  -0.246  0.80570
## Change_NM_foreigners           -0.14782    0.05556  -2.661  0.00786 **
## SVP_change_imputed              0.03459    0.01705   2.029  0.04258 *
## Violent_offences                2.08890    0.09082  23.000  < 2e-16 ***
## Municipality_typeSemi-Urban    -1.25036    0.42427  -2.947  0.00324 **
## Municipality_typeUrban         -1.99291    0.49829  -4.000 6.57e-05 ***
## Income_Categoryrich             2.02134    0.37363   5.410 7.02e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.652 on 2114 degrees of freedom
## Multiple R-squared:  0.222,  Adjusted R-squared:  0.2183
## F-statistic: 60.33 on 10 and 2114 DF,  p-value: < 2.2e-16
```

Only high increases in Muslims are significant at the 5% level. There is considerably less significance than when including the baselines.

2nd Simplified Model: No Time-Varying Controls nor Baselines

```r
# No lagged model
FD_Muslims_Final2_simplified2 <- lm(Yes_percent_change ~ Change_Muslim_Cat
                               + Violent_offences + Municipality_type
                               + Income_Category,
                          data = final_merged)

summary(FD_Muslims_Final2_simplified2)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Violent_offences +
##     Municipality_type + Income_Category, data = final_merged)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.655  -5.213  -0.788   4.776  32.925
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   -14.21835    0.55409 -25.661  < 2e-16 ***
## Change_Muslim_CatHigh Increase  -1.15656    0.50855  -2.274   0.0231 *
## Change_Muslim_CatLow Increase   -0.56523    0.49730  -1.137   0.2558
## Change_Muslim_CatLow Decrease    0.11242    0.65469   0.172   0.8637
## Change_Muslim_CatHigh Decrease  -0.25758    0.65650  -0.392   0.6948
## Violent_offences                 2.04361    0.08935  22.872  < 2e-16 ***
## Municipality_typeSemi-Urban     -1.22119    0.42500  -2.873   0.0041 **
## Municipality_typeUrban          -2.02152    0.49843  -4.056 5.18e-05 ***
## Income_Categoryrich              1.92691    0.37027   5.204 2.14e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.668 on 2116 degrees of freedom
## Multiple R-squared:  0.218,  Adjusted R-squared:  0.2151
## F-statistic: 73.75 on 8 and 2116 DF,  p-value: < 2.2e-16
```

Only high increases in Muslims are significant at the 5% level. Interestingly, while the coefficients are similar to the vanilla model, the p-values are lower (for the change in Muslims).

Vanilla Model

```r
# No lagged model
FD_Muslims_Vanilla <- lm(Yes_percent_change ~ Change_Muslim_Cat,
                         data = final_merged)

summary(FD_Muslims_Vanilla)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat, data = final_merged)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -36.095  -6.326  -1.857   5.875  34.822
```

```
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    -5.4435     0.4379 -12.432   <2e-16 ***
## Change_Muslim_CatHigh Increase  -0.4360     0.5535  -0.788    0.431
## Change_Muslim_CatLow Increase   -0.4622     0.5535  -0.835    0.404
## Change_Muslim_CatLow Decrease    0.1212     0.7373   0.164    0.869
## Change_Muslim_CatHigh Decrease   0.3107     0.7373   0.421    0.674
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.658 on 2120 degrees of freedom
## Multiple R-squared:  0.001082,   Adjusted R-squared:  -0.0008023
## F-statistic: 0.5743 on 4 and 2120 DF,  p-value: 0.6813
```

All coefficients are positive and none is significant.

Part 10.2.2 - Assumptions, Model Diagnostics, and Robustness Checks:

1) No autocorrelation
2) Multicollinearity
3) Homoskedasticity
4) Models Without Imputation
5) Models Without Outliers
6) Tests for the linearity and normality assumptions are computed simultaneously for all models in section 12

```
dwtest(FD_Muslims_Final2)
```

1) Autocorrelation

```
##
##  Durbin-Watson test
##
## data:  FD_Muslims_Final2
## DW = 1.3713, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```
dwtest(FD_Muslims_Final2_simplified1)
```

```
##
##  Durbin-Watson test
##
## data:  FD_Muslims_Final2_simplified1
## DW = 0.89864, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
dwtest(FD_Muslims_Final2_simplified2)
```

```
##
##  Durbin-Watson test
##
## data:  FD_Muslims_Final2_simplified2
## DW = 0.88658, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
dwtest(FD_Muslims_Vanilla)
```

```
##
##  Durbin-Watson test
##
## data:  FD_Muslims_Vanilla
## DW = 0.6557, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

All p-values are well below the 0.1% level. We reject the null hypothesis of no autocorrelation. There is positive autocorrelation (Analyttica Datalab, 2021) for the all specifications (DW <2).This implies that the standard errors could be underestimated and variables considered significant when they are not.

We correct for the above using the Cochrane-Orcutt procedure

```r
# Applying the Cochrane-Orcutt procedure to address autocorrelation (Rdocumentation)
Corrected_FD_Muslims_Final2 <- cochrane.orcutt(FD_Muslims_Final2)
Corrected_FD_Muslims_Final2_simplified1 <- cochrane.orcutt(FD_Muslims_Final2_simplified1)
Corrected_FD_Muslims_Final2_simplified2 <-
  cochrane.orcutt(FD_Muslims_Final2_simplified2)
Corrected_FD_Muslims_Vanilla <- cochrane.orcutt(FD_Muslims_Vanilla)

# Summaries
summary(Corrected_FD_Muslims_Final2)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change_imputed + Minaret_yes_percent + Muslim_share_10 +
##     NM_Foreigner_share_10 + Violent_offences + Municipality_type +
##     Income_Category, data = final_merged)
##
##                               Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                  22.895088   1.205642  18.990 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase -0.918887   0.335177  -2.741 0.0061676 **
## Change_Muslim_CatLow Increase  -0.820376   0.318991  -2.572 0.0101854 *
## Change_Muslim_CatLow Decrease  -0.008921   0.416789  -0.021 0.9829253
## Change_Muslim_CatHigh Decrease  1.202854   0.451117   2.666 0.0077256 **
## Change_NM_foreigners           0.103957   0.037593   2.765 0.0057353 **
## SVP_change_imputed             0.072579   0.014422   5.032 5.256e-07 ***
## Minaret_yes_percent           -0.491295   0.014513 -33.852 < 2.2e-16 ***
## Muslim_share_10               -1.025140   0.148478  -6.904 6.650e-12 ***
## NM_Foreigner_share_10          0.015701   0.019087   0.823 0.4108060
## Violent_offences               0.969727   0.088313  10.981 < 2.2e-16 ***
```

```
## Municipality_typeSemi-Urban    -0.572846   0.307296  -1.864 0.0624392 .
## Municipality_typeUrban         -1.472187   0.395220  -3.725 0.0002005 ***
## Income_Categoryrich            -1.132022   0.282807  -4.003 6.476e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.1538 on 2114 degrees of freedom
## Multiple R-squared:  0.4458 ,  Adjusted R-squared:  0.4435
## F-statistic: 130.6 on 9 and 2114 DF,  p-value: < 6.349e-259
##
## Durbin-Watson statistic
## (original):    1.37125 , p-value: 1.549e-48
## (transformed): 2.20479 , p-value: 1e+00
```

```
summary(Corrected_FD_Muslims_Final2_simplified1)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change_imputed + Violent_offences + Municipality_type +
##     Income_Category, data = final_merged)
##
##                                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                   -11.229074   0.719218 -15.613 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase  -0.379492   0.370733  -1.024  0.306129
## Change_Muslim_CatLow Increase   -0.584024   0.352710  -1.656  0.097907 .
## Change_Muslim_CatLow Decrease    0.025425   0.458584   0.055  0.955790
## Change_Muslim_CatHigh Decrease  -0.031678   0.463712  -0.068  0.945543
## Change_NM_foreigners            0.017199   0.041874   0.411  0.681315
## SVP_change_imputed              0.048281   0.018601   2.596  0.009507 **
## Violent_offences                1.190681   0.124279   9.581 < 2.2e-16 ***
## Municipality_typeSemi-Urban     -0.180276   0.364207  -0.495  0.620664
## Municipality_typeUrban          -0.506911   0.419362  -1.209  0.226887
## Income_Categoryrich             1.419870   0.318811   4.454 8.883e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.2616 on 2117 degrees of freedom
## Multiple R-squared:  0.0558 ,  Adjusted R-squared:  0.0532
## F-statistic: 12.5 on 6 and 2117 DF,  p-value: < 2.383e-21
##
## Durbin-Watson statistic
## (original):    0.89864 , p-value: 2.05e-143
## (transformed): 2.42667 , p-value: 1e+00
```

```
summary(Corrected_FD_Muslims_Final2_simplified2)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Violent_offences +
##     Municipality_type + Income_Category, data = final_merged)
##
##                                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                   -11.037635   0.708019 -15.589 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase  -0.372719   0.371238  -1.004    0.3155
```

```
## Change_Muslim_CatLow Increase   -0.566406   0.353266  -1.603    0.1090
## Change_Muslim_CatLow Decrease    0.042479   0.459548   0.092    0.9264
## Change_Muslim_CatHigh Decrease   0.046092   0.463446   0.099    0.9208
## Violent_offences                 1.168060   0.123451   9.462 < 2.2e-16 ***
## Municipality_typeSemi-Urban     -0.253877   0.363879  -0.698    0.4854
## Municipality_typeUrban          -0.638684   0.417212  -1.531    0.1260
## Income_Categoryrich              1.352293   0.318093   4.251 2.218e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.269 on 2119 degrees of freedom
## Multiple R-squared:  0.0534 ,  Adjusted R-squared:  0.0516
## F-statistic: 14.9 on 4 and 2119 DF,  p-value: < 1.969e-21
##
## Durbin-Watson statistic
## (original):    0.88658 , p-value: 2.849e-146
## (transformed): 2.41925 , p-value: 1e+00
```

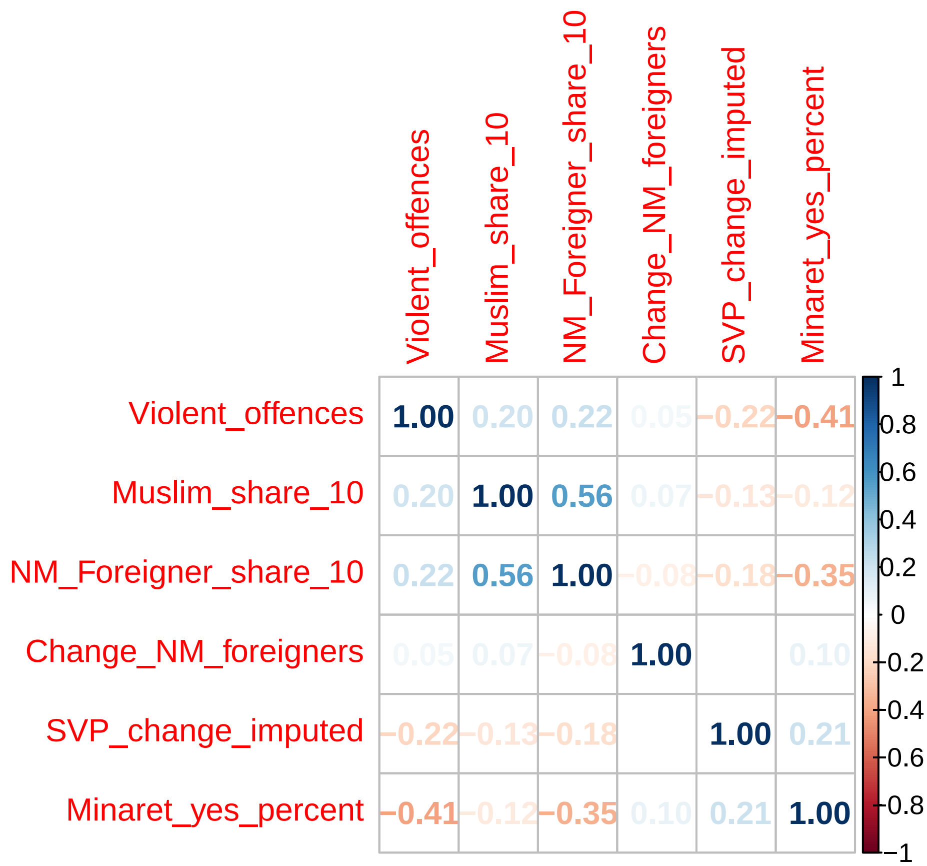
```
summary(Corrected_FD_Muslims_Vanilla)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat, data = final_merged)
##
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     -5.50844    0.49330 -11.167   <2e-16 ***
## Change_Muslim_CatHigh Increase -0.18830    0.36094  -0.522   0.6019
## Change_Muslim_CatLow Increase  -0.45147    0.34677  -1.302   0.1931
## Change_Muslim_CatLow Decrease   0.12202    0.45350   0.269   0.7879
## Change_Muslim_CatHigh Decrease  0.10673    0.45692   0.234   0.8153
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.4049 on 2122 degrees of freedom
## Multiple R-squared:  0.0016 ,  Adjusted R-squared:  0.0012
## F-statistic: 0.9 on 1 and 2122 DF,  p-value: < 4.87e-01
##
## Durbin-Watson statistic
## (original):    0.65570 , p-value: 2.311e-211
## (transformed): 2.50029 , p-value: 1e+00
```

- For the main specification, significance of the primary variable of interests is unchanged, but the scale of th estimates is slightly reduced.
- While the "simplified" models have reduced standard errors, the reduction is not sufficient to make the estimates s.s.. In fact, there is less significance.
- The vanilla model remains insignificant, although coefficients for all but high decreases have changed direction.

```
# For illustration purposes and high level insights, we create a confusion
# matrix for the numeric variables of the main specification
```

```r
# Subset the numeric variables
variables_of_interest <- c("Change_NM_foreigners", "Muslim_share_10",
                           "NM_Foreigner_share_10", "SVP_change_imputed",
                           "Minaret_yes_percent", "Violent_offences")

numeric_data <- final_merged[variables_of_interest]

# Calculate the correlation matrix for these variables
cor_matrix <- cor(numeric_data, use = "complete.obs")

testRes = cor.mtest(numeric_data, conf.level = 0.95)
# Plot the Pearson correlation matrix, leave insignificant correlations blank
corrplot(cor_matrix, method = 'number', order = 'hclust', p.mat = testRes$p, insig = 'blank')
```



2) Multicollinearity

```r
# There is moderate correlation between the NM foreigner and Muslim initial shares,
# as well as between the minaret yet percent and violent offenses. Other correlations
# are weak.

# VIF for more precise estimations
VIF(FD_Muslims_Final2)
```

```
##                          GVIF Df GVIF^(1/(2*Df))
## Change_Muslim_Cat    1.371622  4        1.040290
## Change_NM_foreigners 1.049020  1        1.024217
## SVP_change_imputed   1.110929  1        1.054006
```

```
## Minaret_yes_percent   1.501686  1        1.225433
## Muslim_share_10       2.054085  1        1.433208
## NM_Foreigner_share_10 1.809626  1        1.345224
## Violent_offences      1.281154  1        1.131881
## Municipality_type     1.877634  2        1.170584
## Income_Category       1.403949  1        1.184884
```

```
VIF(FD_Muslims_Final2_simplified1)
```

```
##                           GVIF Df GVIF^(1/(2*Df))
## Change_Muslim_Cat    1.102765  4        1.012303
## Change_NM_foreigners 1.014955  1        1.007450
## SVP_change_imputed   1.092945  1        1.045440
## Violent_offences     1.099243  1        1.048448
## Municipality_type    1.333218  2        1.074547
## Income_Category      1.266757  1        1.125503
```

```
VIF(FD_Muslims_Final2_simplified2)
```

```
##                        GVIF Df GVIF^(1/(2*Df))
## Change_Muslim_Cat 1.089149  4        1.010732
## Violent_offences  1.059464  1        1.029303
## Municipality_type 1.325984  2        1.073086
## Income_Category   1.238880  1        1.113050
```

```
# Multicollinearity is not problematic and does not require further investigation
# (below 4 for all variables)

# Note 1: we test on the original non-corrected models due to the inability of
# the VIF function to deal with Orcutt objects. However, the correlation between
# predictors remains unchanged in both cases, since the variables are identical.

# Note 2: there cannot be multicollinearity in the vanilla model given it has only
# one predictor variable. Accordingly, it is not tested.
```

```
# Breusch-Pagan Test
bptest(Corrected_FD_Muslims_Final2)
```

3) Homoskedasticity

```
##
##  studentized Breusch-Pagan test
##
## data:  Corrected_FD_Muslims_Final2
## BP = 65.663, df = 13, p-value = 4.995e-09
```

```
bptest(Corrected_FD_Muslims_Final2_simplified1)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  Corrected_FD_Muslims_Final2_simplified1
## BP = 39.328, df = 10, p-value = 2.224e-05
```

```
bptest(Corrected_FD_Muslims_Final2_simplified2)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  Corrected_FD_Muslims_Final2_simplified2
## BP = 40.624, df = 8, p-value = 2.451e-06
```

```
bptest(Corrected_FD_Muslims_Vanilla)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  Corrected_FD_Muslims_Vanilla
## BP = 10.132, df = 4, p-value = 0.03826
```

- We fail to reject the null hypothesis of no heteroskedasticity for the vanilla model (p=0.19)
- For all other models, the small p-values lead us to reject the H0 of no heteroscedasticity: variance is non-constant.

While this does not affect the value of the estimated coefficients, it implies the standard errors could be imprecisely calculated, leading to falsely significant p-values.

Because we are using wide format data and have already converted our model to account for autocorrelation, we cannot use functions coeftest and vcovHC, which directly produce a summary using robust standard errors. This is because they require objects of class lm, not orcutt.

A WLS version, as well as a version with HC1 std. errors of the main specification (non-orcutt object) will nonetheless be computed in section 12 to estimate the impact of heteroskedasticity on our coefficients.

4) Models without Imputation The only variables which were imputed are the change in SVP support and Municipality wealth status.

```
# Main Specification
FD_Muslims_Final2_no_imput <- lm(Yes_percent_change ~ Change_Muslim_Cat +
                                   Change_NM_foreigners +
                          SVP_change + Minaret_yes_percent + Muslim_share_10 +
                        NM_Foreigner_share_10 + Violent_offences
                        + Municipality_type + Income_Category_no_impute,
                        data = final_merged)
dwtest(FD_Muslims_Final2_no_imput)
```

```
##
##  Durbin-Watson test
```

```
##
## data:  FD_Muslims_Final2_no_imput
## DW = 1.3568, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```
Corrected_FD_Muslims_Final2_no_imput <- cochrane.orcutt(FD_Muslims_Final2_no_imput)
summary(Corrected_FD_Muslims_Final2_no_imput)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change + Minaret_yes_percent + Muslim_share_10 + NM_Foreigner_share_10 +
##     Violent_offences + Municipality_type + Income_Category_no_impute,
##     data = final_merged)
##
##                                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                    22.530852   1.210614  18.611 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase -0.842367   0.336477  -2.503 0.0123750 *
## Change_Muslim_CatLow Increase  -0.805473   0.319907  -2.518 0.0118840 *
## Change_Muslim_CatLow Decrease  -0.024952   0.413561  -0.060 0.9518948
## Change_Muslim_CatHigh Decrease  1.066788   0.446469   2.389 0.0169667 *
## Change_NM_foreigners            0.086337   0.037339   2.312 0.0208630 *
## SVP_change                      0.078078   0.015301   5.103 3.654e-07 ***
## Minaret_yes_percent            -0.483530   0.014689 -32.918 < 2.2e-16 ***
## Muslim_share_10                -0.935211   0.144577  -6.469 1.235e-10 ***
## NM_Foreigner_share_10           0.006471   0.018818   0.344 0.7309712
## Violent_offences                0.944286   0.088732  10.642 < 2.2e-16 ***
## Municipality_typeSemi-Urban    -0.550843   0.304559  -1.809 0.0706516 .
## Municipality_typeUrban         -1.424257   0.388260  -3.668 0.0002504 ***
## Income_Category_no_imputerich  -1.024590   0.278869  -3.674 0.0002448 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.0061 on 2044 degrees of freedom
## Multiple R-squared:  0.438 ,  Adjusted R-squared:  0.4355
## F-statistic: 122.3 on 9 and 2044 DF,  p-value: < 7.067e-244
##
## Durbin-Watson statistic
## (original):    1.35679 , p-value: 4.186e-49
## (transformed): 2.23641 , p-value: 1e+00
```

```
# Simplified Model 1: No Baseline Controls

FD_Muslims_Final2_simplified1_no_imput <- lm(Yes_percent_change ~
                                               Change_Muslim_Cat +
                                               Change_NM_foreigners +
                                               SVP_change + Violent_offences +
                                               Municipality_type +
                                               Income_Category_no_impute,
                                             data = final_merged)
dwtest(FD_Muslims_Final2_simplified1_no_imput)
```

```
##
##  Durbin-Watson test
```

```
##
## data:  FD_Muslims_Final2_simplified1_no_imput
## DW = 0.86221, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```
Corrected_FD_Muslims_Final2_simplified1_no_imput <-
  cochrane.orcutt(FD_Muslims_Final2_simplified1_no_imput)
summary(Corrected_FD_Muslims_Final2_simplified1_no_imput)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change + Violent_offences + Municipality_type + Income_Category_no_impute,
##     data = final_merged)
##
##                                  Estimate  Std. Error t value  Pr(>|t|)
## (Intercept)                   -10.7384987   0.7250784 -14.810 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase  -0.4893211   0.3691258  -1.326  0.185113
## Change_Muslim_CatLow Increase   -0.7368936   0.3506483  -2.102  0.035718 *
## Change_Muslim_CatLow Decrease   -0.1238508   0.4514220  -0.274  0.783838
## Change_Muslim_CatHigh Decrease  -0.2851114   0.4578800  -0.623  0.533566
## Change_NM_foreigners             0.0076103   0.0410704   0.185  0.853014
## SVP_change                       0.0532326   0.0195200   2.727  0.006445 **
## Violent_offences                 1.1028653   0.1229763   8.968 < 2.2e-16 ***
## Municipality_typeSemi-Urban     -0.0224290   0.3560429  -0.063  0.949777
## Municipality_typeUrban          -0.3695180   0.4077039  -0.906  0.364863
## Income_Category_no_imputerich    1.4494525   0.3089339   4.692 2.888e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.0377 on 2047 degrees of freedom
## Multiple R-squared:  0.0539 ,  Adjusted R-squared:  0.0511
## F-statistic: 11.6 on 6 and 2047 DF,  p-value: < 1.111e-19
##
## Durbin-Watson statistic
## (original):    0.86221 , p-value: 5.934e-148
## (transformed): 2.44827 , p-value: 1e+00
```

```
# Simplified Model 2: No Time-Varying nor Baseline Controls


FD_Muslims_Final2_simplified2_no_imput <- lm(Yes_percent_change ~ Change_Muslim_Cat
                               + Violent_offences + Municipality_type
                               + Income_Category_no_impute,
                          data = final_merged)
dwtest(FD_Muslims_Final2_simplified2_no_imput)
```

```
##
##  Durbin-Watson test
##
## data:  FD_Muslims_Final2_simplified2_no_imput
## DW = 0.84681, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
Corrected_FD_Muslims_Final2_simplified2_no_imput <-
  cochrane.orcutt(FD_Muslims_Final2_simplified2_no_imput)
summary(Corrected_FD_Muslims_Final2_simplified2_no_imput)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Violent_offences +
##     Municipality_type + Income_Category_no_impute, data = final_merged)
##
##                                   Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                     -10.580512   0.711126 -14.879 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase   -0.508736   0.367168  -1.386   0.16603
## Change_Muslim_CatLow Increase    -0.730772   0.349345  -2.092   0.03658 *
## Change_Muslim_CatLow Decrease    -0.102319   0.449115  -0.228   0.81981
## Change_Muslim_CatHigh Decrease   -0.232187   0.455790  -0.509   0.61051
## Violent_offences                  1.082745   0.121932   8.880 < 2.2e-16 ***
## Municipality_typeSemi-Urban      -0.084631   0.353740  -0.239   0.81094
## Municipality_typeUrban           -0.489657   0.402523  -1.216   0.22395
## Income_Category_no_imputerich     1.349146   0.306186   4.406 1.106e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.0306 on 2061 degrees of freedom
## Multiple R-squared:  0.0511 ,  Adjusted R-squared:  0.0492
## F-statistic: 13.8 on 4 and 2061 DF,  p-value: < 9.954e-20
##
## Durbin-Watson statistic
## (original):    0.84681 , p-value: 1.58e-152
## (transformed): 2.44114 , p-value: 1e+00
```

```r
# Vanilla model: the change in Muslim shares was not imputed
```

- For all models, any effect of the removal of imputed observations is small to the point it is not perceptible.

5) Models Without Outliers Main Specification Without Outliers

```r
# Calculating diagnostics
hat_values_Final2 <- hatvalues(FD_Muslims_Final2)
cooks_dist_Final2 <- cooks.distance(FD_Muslims_Final2)
std_res_Final2 <- rstandard(FD_Muslims_Final2)

# Identifying outliers, using standard thresholds
high_leverage_Final2 <- which(hat_values_Final2 > 2 * mean(hat_values_Final2))
high_cooks_Final2 <- which(cooks_dist_Final2 > 4/(nrow(final_merged)-
                                                    length(coef(FD_Muslims_Final2))))
high_std_res_Final2 <- which(abs(std_res_Final2) > 3)
outliers_Final2 <- unique(c(high_leverage_Final2, high_cooks_Final2,
                            high_std_res_Final2))



# Refitting the model without outliers
final_merged_no_outliers_main <- final_merged[-outliers_Final2, ]
```

```r
FD_Muslims_Final2_no_outliers <- lm(Yes_percent_change ~ Change_Muslim_Cat +
                                      Change_NM_foreigners + SVP_change_imputed +
                                      Minaret_yes_percent + Muslim_share_10 +
                                      NM_Foreigner_share_10 + Violent_offences +
                                      Municipality_type + Income_Category,
                                    data = final_merged_no_outliers_main)

dwtest(FD_Muslims_Final2_no_outliers)
```

```
##
##  Durbin-Watson test
##
## data:  FD_Muslims_Final2_no_outliers
## DW = 1.4255, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
Corrected_FD_Muslims_Final2_no_outliers <- cochrane.orcutt(FD_Muslims_Final2_no_outliers)
summary(Corrected_FD_Muslims_Final2_no_outliers)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change_imputed + Minaret_yes_percent + Muslim_share_10 +
##     NM_Foreigner_share_10 + Violent_offences + Municipality_type +
##     Income_Category, data = final_merged_no_outliers_main)
##
##                                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                    23.356698   1.116989  20.910 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase -0.783982   0.305862  -2.563  0.010447 *
## Change_Muslim_CatLow Increase  -0.598523   0.292373  -2.047  0.040782 *
## Change_Muslim_CatLow Decrease   0.098899   0.387059   0.256  0.798353
## Change_Muslim_CatHigh Decrease  1.309843   0.435129   3.010  0.002645 **
## Change_NM_foreigners            0.072325   0.038510   1.878  0.060521 .
## SVP_change_imputed              0.103707   0.015856   6.540 7.851e-11 ***
## Minaret_yes_percent            -0.512253   0.013357 -38.350 < 2.2e-16 ***
## Muslim_share_10                -1.066850   0.170815  -6.246 5.188e-10 ***
## NM_Foreigner_share_10           0.058018   0.019340   3.000  0.002736 **
## Violent_offences                1.139099   0.080926  14.076 < 2.2e-16 ***
## Municipality_typeSemi-Urban    -0.970330   0.277728  -3.494  0.000487 ***
## Municipality_typeUrban         -2.066258   0.364383  -5.671 1.640e-08 ***
## Income_Categoryrich            -1.114034   0.254828  -4.372 1.299e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.3956 on 1916 degrees of freedom
## Multiple R-squared:  0.5592 ,  Adjusted R-squared:  0.5572
## F-statistic: 186.6 on 9 and 1916 DF,  p-value: < 0e+00
##
## Durbin-Watson statistic
## (original):    1.42555 , p-value: 2.576e-37
## (transformed): 2.14918 , p-value: 9.995e-01
```

The significance of the primary variable of interest changed noticeably, as well as the coefficients. Yet, these results do not imply that outliers must be removed, as they are expected to be natural (see section 13).

While the scale of the coefficients is reduced, the results still support H1, whereby increases in the Muslim population decreases Islamophobia. Similarly, large decreases in Muslim shares are still associated with an increase in Islamophobia.

1st Simplified Model Without Outliers

```
# Calculating diagnostics
hat_values_simplified1 <- hatvalues(FD_Muslims_Final2_simplified1)
cooks_dist_simplified1 <- cooks.distance(FD_Muslims_Final2_simplified1)
std_res_simplified1 <- rstandard(FD_Muslims_Final2_simplified1)

# Identifying outliers, using standard thresholds
high_leverage_simplified1 <- which(hat_values_simplified1 > 2 * mean(hat_values_simplified1))
high_cooks_simplified1 <- which(cooks_dist_simplified1 > 4/
                             (nrow(final_merged)-
                                length(coef(FD_Muslims_Final2_simplified1))))
high_std_res_simplified1 <- which(abs(std_res_simplified1) > 3)
outliers_simplified1 <- unique(c(high_leverage_simplified1,
                            high_cooks_simplified1, high_std_res_simplified1))


# Refitting the model without outliers
final_merged_no_outliers_simplified1 <- final_merged[-outliers_simplified1, ]
FD_Muslims_Final2_simplified1_no_outliers <-
  lm(Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
                                      SVP_change_imputed + Violent_offences +
                                      Municipality_type + Income_Category,
                                    data = final_merged_no_outliers_simplified1)

dwtest(FD_Muslims_Final2_simplified1_no_outliers)
```

```
##
##  Durbin-Watson test
##
## data:  FD_Muslims_Final2_simplified1_no_outliers
## DW = 0.93535, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```
Corrected_FD_Muslims_Final2_simplified1_no_outliers <-
  cochrane.orcutt(FD_Muslims_Final2_simplified1_no_outliers)
summary(Corrected_FD_Muslims_Final2_simplified1_no_outliers)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change_imputed + Violent_offences + Municipality_type +
##     Income_Category, data = final_merged_no_outliers_simplified1)
##
##                                   Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                     -13.368997   0.682132 -19.599 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase  -0.063675   0.340839  -0.187  0.851822
## Change_Muslim_CatLow Increase   -0.384220   0.324460  -1.184  0.236486
## Change_Muslim_CatLow Decrease   -0.151844   0.431526  -0.352  0.724968
## Change_Muslim_CatHigh Decrease  -0.455526   0.441651  -1.031  0.302474
```

```
## Change_NM_foreigners            -0.044720   0.044841  -0.997  0.318742
## SVP_change_imputed               0.059550   0.019808   3.006  0.002678 **
## Violent_offences                 1.679290   0.124250  13.515 < 2.2e-16 ***
## Municipality_typeSemi-Urban     -0.362465   0.335802  -1.079  0.280544
## Municipality_typeUrban          -0.726869   0.386042  -1.883  0.059866 .
## Income_Categoryrich              1.532923   0.291001   5.268 1.534e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.4727 on 1949 degrees of freedom
## Multiple R-squared:  0.1051 ,  Adjusted R-squared:  0.1024
## F-statistic: 22.8 on 6 and 1949 DF,  p-value: < 5.897e-41
##
## Durbin-Watson statistic
## (original):    0.93535 , p-value: 8.686e-124
## (transformed): 2.38495 , p-value: 1e+00
```

No meaningful difference after the removal of outliers. Still, note the now negative coefficients for all levels of the change in Muslims, by opposition to small decreases having a positive coefficient initially.

2nd Simplified Model Without Outliers

```r
# Calculating diagnostics
hat_values_simplified2 <- hatvalues(FD_Muslims_Final2_simplified2)
cooks_dist_simplified2 <- cooks.distance(FD_Muslims_Final2_simplified2)
std_res_simplified2 <- rstandard(FD_Muslims_Final2_simplified2)

# Identifying outliers, using standard thresholds
high_leverage_simplified2 <- which(hat_values_simplified2 > 2 * mean(hat_values_simplified2))
high_cooks_simplified2 <- which(cooks_dist_simplified2 > 4/
                             (nrow(final_merged)-
                                length(coef(FD_Muslims_Final2_simplified2))))
high_std_res_simplified2 <- which(abs(std_res_simplified2) > 3)
outliers_simplified2 <- unique(c(high_leverage_simplified2,
                            high_cooks_simplified2, high_std_res_simplified2))


# Refitting the model without outliers
final_merged_no_outliers_simplified2 <- final_merged[-outliers_simplified2, ]
FD_Muslims_Final2_simplified2_no_outliers <-
  lm(Yes_percent_change ~ Change_Muslim_Cat + Violent_offences +
                                    Municipality_type + Income_Category,
                                  data = final_merged_no_outliers_simplified2)

dwtest(FD_Muslims_Final2_simplified2_no_outliers)
```

```
##
##  Durbin-Watson test
##
## data:  FD_Muslims_Final2_simplified2_no_outliers
## DW = 0.95765, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
Corrected_FD_Muslims_Final2_simplified2_no_outliers <-
  cochrane.orcutt(FD_Muslims_Final2_simplified2_no_outliers)
summary(Corrected_FD_Muslims_Final2_simplified2_no_outliers)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Violent_offences +
##     Municipality_type + Income_Category, data = final_merged_no_outliers_simplified2)
##
##                              Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                 -13.64226    0.65778 -20.740 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase  -0.22485    0.34546  -0.651   0.51520
## Change_Muslim_CatLow Increase   -0.37070    0.32899  -1.127   0.25997
## Change_Muslim_CatLow Decrease   -0.13619    0.43281  -0.315   0.75304
## Change_Muslim_CatHigh Decrease  -0.29828    0.45087  -0.662   0.50833
## Violent_offences              1.73198    0.12200  14.196 < 2.2e-16 ***
## Municipality_typeSemi-Urban    -0.38242    0.33552  -1.140   0.25452
## Municipality_typeUrban         -0.85191    0.38603  -2.207   0.02744 *
## Income_Categoryrich            1.38515    0.29419   4.708  2.67e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.5906 on 2006 degrees of freedom
## Multiple R-squared:  0.1077 ,  Adjusted R-squared:  0.1059
## F-statistic: 30.2 on 4 and 2006 DF,  p-value: < 6.512e-45
##
## Durbin-Watson statistic
## (original):    0.95765 , p-value: 7.929e-122
## (transformed): 2.38126 , p-value: 1e+00
```

Significance levels are unchanged but the direction of the two levels for decreasing shares has inversed, and is now negative.

Vanilla Model

```r
# Calculating diagnostics
hat_values_vanilla <- hatvalues(FD_Muslims_Vanilla)
cooks_dist_vanilla <- cooks.distance(FD_Muslims_Vanilla)
std_res_vanilla <- rstandard(FD_Muslims_Vanilla)

# Identifying outliers, using standard thresholds
high_leverage_vanilla <- which(hat_values_vanilla > 2 * mean(hat_values_vanilla))
high_cooks_vanilla <- which(cooks_dist_vanilla > 4/
                             (nrow(final_merged)-
                                length(coef(FD_Muslims_Vanilla))))
high_std_res_vanilla <- which(abs(std_res_vanilla) > 3)
outliers_vanilla <- unique(c(high_leverage_vanilla,
                            high_cooks_vanilla, high_std_res_vanilla))


# Refitting the model without outliers
final_merged_no_outliers_vanilla <- final_merged[-outliers_vanilla, ]
FD_Muslims_vanilla_no_outliers <- lm(Yes_percent_change ~ Change_Muslim_Cat,
                                     data = final_merged_no_outliers_vanilla)
```

```r
dwtest(FD_Muslims_vanilla_no_outliers)
```

```
##
##  Durbin-Watson test
##
## data:  FD_Muslims_vanilla_no_outliers
## DW = 0.68733, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
Corrected_FD_Muslims_vanilla_no_outliers <-
  cochrane.orcutt(FD_Muslims_vanilla_no_outliers)
summary(Corrected_FD_Muslims_vanilla_no_outliers)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat, data = final_merged_no_outliers_vanilla)
##
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -5.97656    0.45022 -13.275   <2e-16 ***
## Change_Muslim_CatHigh Increase -0.11172    0.33581  -0.333   0.7394
## Change_Muslim_CatLow Increase  -0.31441    0.32465  -0.968   0.3329
## Change_Muslim_CatLow Decrease  -0.26751    0.43535  -0.614   0.5390
## Change_Muslim_CatHigh Decrease -0.27772    0.45695  -0.608   0.5434
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.7985 on 2014 degrees of freedom
## Multiple R-squared:  6e-04 ,  Adjusted R-squared:  1e-04
## F-statistic: 0.3 on 1 and 2014 DF,  p-value: < 8.744e-01
##
## Durbin-Watson statistic
## (original):    0.68733 , p-value: 1.572e-191
## (transformed): 2.47222 , p-value: 1e+00
```

Significance levels are unchanged but the direction of the two levels for decreasing shares has inversed, and is now negative.

End of Part 10

Part 11: Assessing Hypothesis H2: Arabs vs non-Arab Muslim Effects

The Turkish represent 70% of the initial non-arab Muslim presence by themselves, and still a considerable share afterwards. Turkish immigration is not new in Switzerland, starting in the 1960s. Around 45% of all Turks are naturalized, compared to 11% for other Muslims. They represent the largest non-European foreigner group in Switzerland (Le Temps). There is a reasonable possibility that the change in Turkish population could distort the effects for the entire MNA/NAM group. Accordingly, we investigate this with a third model controlling for the Turkish population.

Part 11.1 - Set Up of All the Relevant Subsets

```
# 1) Set up for the Arab Model

# We keep the same boundary values for the no change level.

# Create the thresholds for the changes in shares
negative_changes_arabs <- final_merged$Change_arabs[final_merged$Change_arabs <
                                                      no_change_lower_bound]
positive_changes_arabs <- final_merged$Change_arabs[final_merged$Change_arabs >
                                                      no_change_upper_bound]

median_negative_change <- median(negative_changes_arabs, na.rm = TRUE)
median_positive_change <- median(positive_changes_arabs, na.rm = TRUE)

final_merged <- final_merged %>%
  mutate(Change_Arabs_Cat2 = case_when(
    Change_arabs < median_negative_change ~ "High Decrease",
    Change_arabs >= median_negative_change & Change_arabs < no_change_lower_bound ~
      "Low Decrease",
    Change_arabs >= no_change_lower_bound & Change_arabs <= no_change_upper_bound ~
      "No Change",
    Change_arabs > no_change_upper_bound & Change_arabs <= median_positive_change ~
      "Low Increase",
    Change_arabs > median_positive_change ~ "High Increase"
  ))

final_merged$Change_Arabs_Cat2 <- factor(final_merged$Change_Arabs_Cat2,
                                         levels = c("No Change", "High Increase",
                                                    "Low Increase", "Low Decrease",
                                                    "High Decrease"))



# 2) Set up for the MNA Model

# Repeating the same process
negative_changes_MNA <- final_merged$Change_MNA[final_merged$Change_MNA <
                                                  no_change_lower_bound]
positive_changes_MNA <- final_merged$Change_MNA[final_merged$Change_MNA >
                                                  no_change_upper_bound]

median_negative_change <- median(negative_changes_MNA, na.rm = TRUE)
median_positive_change <- median(positive_changes_MNA, na.rm = TRUE)

final_merged <- final_merged %>%
  mutate(Change_MNA_Cat2 = case_when(
    Change_MNA < median_negative_change ~ "High Decrease",
    Change_MNA >= median_negative_change & Change_MNA < no_change_lower_bound ~
      "Low Decrease",
    Change_MNA >= no_change_lower_bound & Change_MNA <= no_change_upper_bound ~
      "No Change",
    Change_MNA > no_change_upper_bound & Change_MNA <= median_positive_change ~
```

```r
      "Low Increase",
    Change_MNA > median_positive_change ~ "High Increase"
  ))

# Convert to factor,  set "No Change" as the reference category, reorder
final_merged$Change_MNA_Cat2 <- factor(final_merged$Change_MNA_Cat2,
                                       levels = c("No Change", "High Increase",
                                                  "Low Increase", "Low Decrease",
                                                  "High Decrease"))




# 3) Set up for the MNA without Turks Model

# Create the necessary variables
final_merged$Change_MNA_NO_TURKS <- final_merged$Change_MNA -
  final_merged$Change_turkish
final_merged$MNA_NO_TURKS_share_10 <- final_merged$MNA_share_10 -
  final_merged$Turkish_share_10

# Creating the level thresholds, as done above.
negative_changes_MNA_NO_TURKS <-
  final_merged$Change_MNA_NO_TURKS[final_merged$Change_MNA_NO_TURKS <
                                     no_change_lower_bound]
positive_changes_MNA_NO_TURKS <-
  final_merged$Change_MNA_NO_TURKS[final_merged$Change_MNA_NO_TURKS >
                                     no_change_upper_bound]

median_negative_change <- median(negative_changes_MNA_NO_TURKS)
median_positive_change <- median(positive_changes_MNA_NO_TURKS)


# Re-code the MNA no Turk changes into a categorical variable with 5 categories
final_merged <- final_merged %>%
  mutate(Change_MNA_NO_TURKS_Cat = case_when(
    Change_MNA_NO_TURKS < median_negative_change ~ "High Decrease",
    Change_MNA_NO_TURKS >= median_negative_change & Change_MNA_NO_TURKS <
      no_change_lower_bound ~ "Low Decrease",
    Change_MNA_NO_TURKS >= no_change_lower_bound & Change_MNA_NO_TURKS <=
      no_change_upper_bound ~ "No Change",
    Change_MNA_NO_TURKS > no_change_upper_bound & Change_MNA_NO_TURKS <=
      median_positive_change ~ "Low Increase",
    Change_MNA_NO_TURKS > median_positive_change ~ "High Increase"
  ))

# Convert to factor,  set "No Change" as the reference category, reorder
final_merged$Change_MNA_NO_TURKS_Cat <- factor(final_merged$Change_MNA_NO_TURKS_Cat,
                                     levels = c("No Change", "High Increase",
                                                "Low Increase", "Low Decrease",
                                                "High Decrease"))
```

Part 11.2: Modelling

The below models consist of the main specification applied to the 3 different Muslim sub-groups.

1. Model for Arabs

```r
FD_Arabs <- lm(Yes_percent_change ~ Change_Arabs_Cat2 + Change_MNA
                    + Change_NM_foreigners + SVP_change_imputed + Minaret_yes_percent
                    + MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10
                    + Violent_offences + Municipality_type + Income_Category,
                      data = final_merged)
# Original Output
summary(FD_Arabs)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_Arabs_Cat2 + Change_MNA +
##     Change_NM_foreigners + SVP_change_imputed + Minaret_yes_percent +
##     MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category, data = final_merged)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.4415  -3.3046   0.0044   3.4504  20.4542
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    27.13676    1.11249  24.393  < 2e-16 ***
## Change_Arabs_Cat2High Increase -1.19326    0.34617  -3.447 0.000578 ***
## Change_Arabs_Cat2Low Increase  -0.53434    0.32657  -1.636 0.101937
## Change_Arabs_Cat2Low Decrease  -0.42897    0.47445  -0.904 0.366020
## Change_Arabs_Cat2High Decrease  1.10820    0.51618   2.147 0.031913 *
## Change_MNA                     -0.55243    0.30538  -1.809 0.070594 .
## Change_NM_foreigners            0.10559    0.04065   2.598 0.009453 **
## SVP_change_imputed              0.06425    0.01233   5.212 2.05e-07 ***
## Minaret_yes_percent            -0.55996    0.01366 -40.995  < 2e-16 ***
## MNA_share_10                   -1.22219    0.19030  -6.422 1.65e-10 ***
## Arab_share_10                  -0.52997    0.44568  -1.189 0.234527
## NM_Foreigner_share_10           0.03897    0.01848   2.109 0.035059 *
## Violent_offences                1.06588    0.07151  14.905  < 2e-16 ***
## Municipality_typeSemi-Urban    -0.86901    0.30448  -2.854 0.004358 **
## Municipality_typeUrban         -2.07035    0.40581  -5.102 3.67e-07 ***
## Income_Categoryrich            -1.81037    0.28273  -6.403 1.87e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.488 on 2109 degrees of freedom
## Multiple R-squared:  0.6007, Adjusted R-squared:  0.5978
## F-statistic: 211.5 on 15 and 2109 DF,  p-value: < 2.2e-16
```

```r
dwtest(FD_Arabs)
```

```
##
```

```
## 	Durbin-Watson test
##
## data:  FD_Arabs
## DW = 1.3707, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
Corrected_FD_Arabs <- cochrane.orcutt(FD_Arabs)
# Corrected Output
summary(Corrected_FD_Arabs)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Arabs_Cat2 + Change_MNA +
##     Change_NM_foreigners + SVP_change_imputed + Minaret_yes_percent +
##     MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category, data = final_merged)
##
##                                Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                   22.735419   1.197150  18.991 < 2.2e-16 ***
## Change_Arabs_Cat2High Increase -1.052708   0.310905  -3.386 0.0007223 ***
## Change_Arabs_Cat2Low Increase  -0.728923   0.290295  -2.511 0.0121141 *
## Change_Arabs_Cat2Low Decrease  -0.701038   0.417322  -1.680 0.0931348 .
## Change_Arabs_Cat2High Decrease  0.901727   0.459080   1.964 0.0496374 *
## Change_MNA                    -0.380393   0.272887  -1.394 0.1634768
## Change_NM_foreigners           0.101074   0.037702   2.681 0.0074007 **
## SVP_change_imputed             0.074478   0.014445   5.156 2.756e-07 ***
## Minaret_yes_percent           -0.488594   0.014566 -33.544 < 2.2e-16 ***
## MNA_share_10                  -0.805809   0.185189  -4.351 1.418e-05 ***
## Arab_share_10                 -0.766301   0.405175  -1.891 0.0587234 .
## NM_Foreigner_share_10          0.012284   0.019193   0.640 0.5222277
## Violent_offences               0.963800   0.089447  10.775 < 2.2e-16 ***
## Municipality_typeSemi-Urban   -0.634418   0.306484  -2.070 0.0385750 *
## Municipality_typeUrban        -1.585331   0.394465  -4.019 6.052e-05 ***
## Income_Categoryrich           -1.072562   0.283667  -3.781 0.0001605 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.1594 on 2112 degrees of freedom
## Multiple R-squared:  0.4439 ,  Adjusted R-squared:  0.441
## F-statistic: 112.2 on 11 and 2112 DF,  p-value: < 3.24e-255
##
## Durbin-Watson statistic
## (original):    1.37068 , p-value: 1.218e-48
## (transformed): 2.20552 , p-value: 1e+00
```

While the same pattern as for the full Muslim main specification applies for the increases - increasing Muslim shares decrease Islamophobia - the coefficients for decreasing shares are now insignificant.

2. Model for MNA

```r
FD_MNA <- lm(Yes_percent_change ~ Change_MNA_Cat2 + Change_arabs
                 + Change_NM_foreigners + SVP_change_imputed
                 + Minaret_yes_percent + MNA_share_10 + Arab_share_10
                 + NM_Foreigner_share_10 + Violent_offences
```

```r
                 + Municipality_type + Income_Category, data = final_merged)
# Original Output
summary(FD_MNA)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_MNA_Cat2 + Change_arabs +
##     Change_NM_foreigners + SVP_change_imputed + Minaret_yes_percent +
##     MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category, data = final_merged)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.6062  -3.3190   0.0369   3.4879  20.3281
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  27.47057    1.12035  24.520  < 2e-16 ***
## Change_MNA_Cat2High Increase -0.57950    0.35516  -1.632 0.102902
## Change_MNA_Cat2Low Increase  -0.88511    0.34119  -2.594 0.009547 **
## Change_MNA_Cat2Low Decrease  -0.56178    0.44071  -1.275 0.202548
## Change_MNA_Cat2High Decrease  0.60227    0.51568   1.168 0.242975
## Change_arabs                 -1.21462    0.36344  -3.342 0.000846 ***
## Change_NM_foreigners          0.10605    0.04065   2.608 0.009158 **
## SVP_change_imputed            0.06347    0.01233   5.146 2.90e-07 ***
## Minaret_yes_percent          -0.56175    0.01365 -41.163  < 2e-16 ***
## MNA_share_10                 -1.34473    0.19873  -6.767 1.70e-11 ***
## Arab_share_10                -0.40498    0.42862  -0.945 0.344844
## NM_Foreigner_share_10         0.03930    0.01849   2.126 0.033638 *
## Violent_offences              1.05518    0.07175  14.707  < 2e-16 ***
## Municipality_typeSemi-Urban  -0.87791    0.30461  -2.882 0.003991 **
## Municipality_typeUrban       -2.09794    0.40566  -5.172 2.54e-07 ***
## Income_Categoryrich          -1.85408    0.28334  -6.544 7.52e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.49 on 2109 degrees of freedom
## Multiple R-squared:  0.6004, Adjusted R-squared:  0.5976
## F-statistic: 211.3 on 15 and 2109 DF,  p-value: < 2.2e-16
```

```r
dwtest(FD_MNA)
```

```
##
##  Durbin-Watson test
##
## data:  FD_MNA
## DW = 1.3785, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
Corrected_FD_MNA <- cochrane.orcutt(FD_MNA)
# Corrected Output
summary(Corrected_FD_MNA)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_MNA_Cat2 + Change_arabs +
##     Change_NM_foreigners + SVP_change_imputed + Minaret_yes_percent +
##     MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category, data = final_merged)
##
##                                Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                   22.938020   1.205350  19.030 < 2.2e-16 ***
## Change_MNA_Cat2High Increase -0.518150   0.321070  -1.614  0.106716
## Change_MNA_Cat2Low Increase  -0.864857   0.306558  -2.821  0.004829 **
## Change_MNA_Cat2Low Decrease  -0.327393   0.392506  -0.834  0.404314
## Change_MNA_Cat2High Decrease  0.213187   0.453127   0.470  0.638062
## Change_arabs                 -0.903969   0.329312  -2.745  0.006102 **
## Change_NM_foreigners          0.105369   0.037762   2.790  0.005312 **
## SVP_change_imputed            0.073727   0.014458   5.099 3.713e-07 ***
## Minaret_yes_percent          -0.490213   0.014576 -33.631 < 2.2e-16 ***
## MNA_share_10                 -0.903782   0.188879  -4.785 1.829e-06 ***
## Arab_share_10                -0.602506   0.391512  -1.539  0.123974
## NM_Foreigner_share_10         0.011013   0.019271   0.571  0.567747
## Violent_offences              0.959502   0.089512  10.719 < 2.2e-16 ***
## Municipality_typeSemi-Urban  -0.632756   0.307592  -2.057  0.039797 *
## Municipality_typeUrban       -1.643982   0.394488  -4.167 3.206e-05 ***
## Income_Categoryrich          -1.115035   0.283951  -3.927 8.883e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.1681 on 2112 degrees of freedom
## Multiple R-squared:  0.4437 ,  Adjusted R-squared:  0.4408
## F-statistic: 112.1 on 11 and 2112 DF,  p-value: < 4.416e-255
##
## Durbin-Watson statistic
## (original):    1.37846 , p-value: 1.654e-47
## (transformed): 2.20351 , p-value: 1e+00
```

The low decrease is s.s. at the 1% level.

3. Model for Non-Arab and non-Turkish Muslims

```r
FD_MNA_NO_TURKS <- lm(Yes_percent_change ~ Change_MNA_NO_TURKS_Cat + Change_arabs
                 + Change_turkish + Change_NM_foreigners + SVP_change_imputed
                 + Minaret_yes_percent + Turkish_share_10 + MNA_NO_TURKS_share_10
                 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences
                 + Municipality_type + Income_Category, data = final_merged)
# Original Output
summary(FD_MNA_NO_TURKS)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_MNA_NO_TURKS_Cat + Change_arabs +
##     Change_turkish + Change_NM_foreigners + SVP_change_imputed +
##     Minaret_yes_percent + Turkish_share_10 + MNA_NO_TURKS_share_10 +
##     Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category, data = final_merged)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.7014  -3.3342   0.0177   3.4659  20.2072
##
## Coefficients:
##                                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          27.25838    1.12825  24.160  < 2e-16 ***
## Change_MNA_NO_TURKS_CatHigh Increase -1.28975    0.34793  -3.707 0.000215 ***
## Change_MNA_NO_TURKS_CatLow Increase  -1.10361    0.32036  -3.445 0.000582 ***
## Change_MNA_NO_TURKS_CatLow Decrease  -0.69409    0.52350  -1.326 0.185033
## Change_MNA_NO_TURKS_CatHigh Decrease  0.77972    0.58966   1.322 0.186205
## Change_arabs                         -1.20728    0.36977  -3.265 0.001112 **
## Change_turkish                        0.53116    0.50441   1.053 0.292446
## Change_NM_foreigners                  0.11080    0.04051   2.735 0.006282 **
## SVP_change_imputed                    0.06388    0.01228   5.200 2.19e-07 ***
## Minaret_yes_percent                  -0.55701    0.01387 -40.156  < 2e-16 ***
## Turkish_share_10                     -1.00183    0.23689  -4.229 2.45e-05 ***
## MNA_NO_TURKS_share_10                -0.31494    0.59777  -0.527 0.598355
## Arab_share_10                        -0.62976    0.46353  -1.359 0.174415
## NM_Foreigner_share_10                 0.04159    0.01842   2.257 0.024083 *
## Violent_offences                      1.03436    0.07172  14.422  < 2e-16 ***
## Municipality_typeSemi-Urban          -0.90513    0.30306  -2.987 0.002853 **
## Municipality_typeUrban               -2.09617    0.40524  -5.173 2.53e-07 ***
## Income_Categoryrich                  -1.82974    0.28214  -6.485 1.10e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.465 on 2107 degrees of freedom
## Multiple R-squared:  0.6044, Adjusted R-squared:  0.6012
## F-statistic: 189.4 on 17 and 2107 DF,  p-value: < 2.2e-16
```

```
dwtest(FD_MNA_NO_TURKS)
```

```
##
##  Durbin-Watson test
##
## data:  FD_MNA_NO_TURKS
## DW = 1.3796, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```
Corrected_FD_MNA_NO_TURKS <- cochrane.orcutt(FD_MNA_NO_TURKS)
# Corrected output
summary(Corrected_FD_MNA_NO_TURKS)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_MNA_NO_TURKS_Cat + Change_arabs +
##     Change_turkish + Change_NM_foreigners + SVP_change_imputed +
##     Minaret_yes_percent + Turkish_share_10 + MNA_NO_TURKS_share_10 +
##     Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category, data = final_merged)
##
##                                       Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                          23.017857   1.203916  19.119 < 2.2e-16 ***
```

```
## Change_MNA_NO_TURKS_CatHigh Increase -1.253889   0.313730  -3.997 6.644e-05 ***
## Change_MNA_NO_TURKS_CatLow Increase  -1.022888   0.287979  -3.552 0.0003908 ***
## Change_MNA_NO_TURKS_CatLow Decrease  -0.360668   0.472637  -0.763 0.4454905
## Change_MNA_NO_TURKS_CatHigh Decrease  0.601151   0.527405   1.140 0.2544874
## Change_arabs                         -0.803458   0.333981  -2.406 0.0162272 *
## Change_turkish                        0.646948   0.451644   1.432 0.1521698
## Change_NM_foreigners                  0.103573   0.037604   2.754 0.0059324 **
## SVP_change_imputed                    0.073638   0.014402   5.113 3.459e-07 ***
## Minaret_yes_percent                  -0.490251   0.014651 -33.463 < 2.2e-16 ***
## Turkish_share_10                     -0.555482   0.228727  -2.429 0.0152413 *
## MNA_NO_TURKS_share_10                -0.494093   0.546277  -0.904 0.3658480
## Arab_share_10                        -0.591315   0.416246  -1.421 0.1555836
## NM_Foreigner_share_10                 0.013822   0.019200   0.720 0.4716712
## Violent_offences                      0.945473   0.089234  10.595 < 2.2e-16 ***
## Municipality_typeSemi-Urban          -0.636198   0.306095  -2.078 0.0377902 *
## Municipality_typeUrban               -1.622729   0.394488  -4.114 4.047e-05 ***
## Income_Categoryrich                  -1.130958   0.282949  -3.997 6.635e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.147 on 2110 degrees of freedom
## Multiple R-squared:  0.4496 ,  Adjusted R-squared:  0.4462
## F-statistic: 101.2 on 13 and 2110 DF,  p-value: < 7.123e-258
##
## Durbin-Watson statistic
## (original):    1.37961 , p-value: 2.252e-47
## (transformed): 2.19981 , p-value: 1e+00
```

Coefficients for increasing shares are the most significant and much larger than in any of the 2 alternative models.

Part 11.3 - Testing First-Difference Assumptions and Robustness Checks:

1) No autocorrelation: all models have previously been showed to suffer from autocorrelation and already corrected
2) Multicollinearity
3) Homoskedasticity
4) Models Without Imputation
5) Models Without Outliers
6) Tests for the linearity and normality assumptions are computed simultaneously for all models in section 12

2) Multicollinearity For conciseness, only VIF tests are computed, considering correlation matrices as redundant.

```
VIF(FD_Arabs)
```

```
##                          GVIF Df GVIF^(1/(2*Df))
## Change_Arabs_Cat2     1.545891  4        1.055960
## Change_MNA            1.363417  1        1.167654
## Change_NM_foreigners  1.055921  1        1.027580
## SVP_change_imputed    1.110755  1        1.053923
```

```
## Minaret_yes_percent   1.530064  1        1.236958
## MNA_share_10          2.187887  1        1.479151
## Arab_share_10         1.794732  1        1.339676
## NM_Foreigner_share_10 1.822278  1        1.349918
## Violent_offences      1.324647  1        1.150933
## Municipality_type     1.864485  2        1.168530
## Income_Category       1.409761  1        1.187334
```

```
VIF(FD_MNA)
```

```
##                           GVIF Df GVIF^(1/(2*Df))
## Change_MNA_Cat2       1.729839  4        1.070904
## Change_arabs          1.286363  1        1.134179
## Change_NM_foreigners  1.055567  1        1.027408
## SVP_change_imputed    1.111087  1        1.054081
## Minaret_yes_percent   1.526339  1        1.235451
## MNA_share_10          2.384430  1        1.544160
## Arab_share_10         1.658868  1        1.287970
## NM_Foreigner_share_10 1.823659  1        1.350429
## Violent_offences      1.332474  1        1.154328
## Municipality_type     1.876381  2        1.170389
## Income_Category       1.414960  1        1.189521
```

```
VIF(FD_MNA_NO_TURKS)
```

```
##                             GVIF Df GVIF^(1/(2*Df))
## Change_MNA_NO_TURKS_Cat 1.749908  4        1.072450
## Change_arabs            1.343747  1        1.159201
## Change_turkish          1.984764  1        1.408817
## Change_NM_foreigners    1.057433  1        1.028316
## SVP_change_imputed      1.112484  1        1.054744
## Minaret_yes_percent     1.591359  1        1.261491
## Turkish_share_10        2.726035  1        1.651071
## MNA_NO_TURKS_share_10   2.113936  1        1.453938
## Arab_share_10           1.957829  1        1.399224
## NM_Foreigner_share_10   1.827552  1        1.351870
## Violent_offences        1.343625  1        1.159148
## Municipality_type       1.875862  2        1.170308
## Income_Category         1.415810  1        1.189878
```

Multicollinearity is not problematic in any of the models

```
bptest(FD_Arabs)
```

3) Homoskedasticity - Breusch-Pagan Tests

```
##
##  studentized Breusch-Pagan test
##
## data:  FD_Arabs
## BP = 97.347, df = 15, p-value = 4.144e-14
```

```
bptest(FD_MNA)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  FD_MNA
## BP = 108.22, df = 15, p-value = 3.537e-16
```

```
bptest(FD_MNA_NO_TURKS)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  FD_MNA_NO_TURKS
## BP = 117.49, df = 17, p-value < 2.2e-16
```

There is some degree of Heteroskedasticity in all models, considering all have very significant p-values (reject H0 of no heteroskedasticity)

```
# Arabs
FD_Arabs_no_impute <- lm(Yes_percent_change ~ Change_Arabs_Cat2 + Change_MNA
                    + Change_NM_foreigners + SVP_change + Minaret_yes_percent
                    + MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10
                    + Violent_offences + Municipality_type + Income_Category_no_impute,
                    data = final_merged)
dwtest(FD_Arabs_no_impute)
```

4) Models without Imputation

```
##
##  Durbin-Watson test
##
## data:  FD_Arabs_no_impute
## DW = 1.3529, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```
Corrected_Arabs_no_impute <- cochrane.orcutt(FD_Arabs_no_impute)
summary(Corrected_Arabs_no_impute)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Arabs_Cat2 + Change_MNA +
##     Change_NM_foreigners + SVP_change + Minaret_yes_percent +
##     MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category_no_impute, data = final_merged)
##
##                                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                   22.3645953  1.2027056  18.595 < 2.2e-16 ***
## Change_Arabs_Cat2High Increase -1.0001553  0.3038261  -3.292 0.0010122 **
```

```
## Change_Arabs_Cat2Low Increase  -0.6900221  0.2842277  -2.428 0.0152806 *
## Change_Arabs_Cat2Low Decrease  -0.7085139  0.4070257  -1.741 0.0818852 .
## Change_Arabs_Cat2High Decrease  0.7969258  0.4463667   1.785 0.0743514 .
## Change_MNA                     -0.3710583  0.2656455  -1.397 0.1626206
## Change_NM_foreigners            0.0835048  0.0373764   2.234 0.0255807 *
## SVP_change                      0.0795283  0.0153202   5.191 2.299e-07 ***
## Minaret_yes_percent            -0.4804716  0.0147288 -32.621 < 2.2e-16 ***
## MNA_share_10                   -0.7687066  0.1797961  -4.275 1.996e-05 ***
## Arab_share_10                  -0.5630388  0.3941765  -1.428 0.1533321
## NM_Foreigner_share_10           0.0029123  0.0189034   0.154 0.8775740
## Violent_offences                0.9346526  0.0899281  10.393 < 2.2e-16 ***
## Municipality_typeSemi-Urban    -0.6095596  0.3037107  -2.007 0.0448773 *
## Municipality_typeUrban         -1.5159547  0.3871256  -3.916 9.302e-05 ***
## Income_Category_no_imputerich  -0.9676514  0.2796089  -3.461 0.0005498 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.009 on 2042 degrees of freedom
## Multiple R-squared:  0.4364 ,  Adjusted R-squared:  0.4334
## F-statistic: 105.2 on 11 and 2042 DF,  p-value: < 1.493e-240
##
## Durbin-Watson statistic
## (original):    1.35288 , p-value: 1.083e-49
## (transformed): 2.23717 , p-value: 1e+00
```

```
# MNA
FD_MNA_no_impute <- lm(Yes_percent_change ~ Change_MNA_Cat2 + Change_arabs
                 + Change_NM_foreigners + SVP_change
                 + Minaret_yes_percent + MNA_share_10 + Arab_share_10
                 + NM_Foreigner_share_10 + Violent_offences
                 + Municipality_type + Income_Category_no_impute, data = final_merged)
dwtest(FD_MNA_no_impute)
```

```
##
##  Durbin-Watson test
##
## data:  FD_MNA_no_impute
## DW = 1.3596, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```
Corrected_FD_MNA_no_impute <- cochrane.orcutt(FD_MNA_no_impute)
summary(Corrected_FD_MNA_no_impute)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_MNA_Cat2 + Change_arabs +
##     Change_NM_foreigners + SVP_change + Minaret_yes_percent +
##     MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category_no_impute, data = final_merged)
##
##                                 Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                   22.5979182  1.2097406  18.680 < 2.2e-16 ***
## Change_MNA_Cat2High Increase  -0.4574348  0.3177448  -1.440 0.1501258
## Change_MNA_Cat2Low Increase   -0.8283787  0.3023812  -2.740 0.0062065 **
```

```
## Change_MNA_Cat2Low Decrease   -0.3322023  0.3862108  -0.860 0.3898032
## Change_MNA_Cat2High Decrease   0.1168776  0.4420190   0.264 0.7914848
## Change_arabs                  -0.9186189  0.3197179  -2.873 0.0041052 **
## Change_NM_foreigners           0.0869260  0.0374607   2.320 0.0204143 *
## SVP_change                     0.0787988  0.0153237   5.142 2.974e-07 ***
## Minaret_yes_percent           -0.4826426  0.0147292 -32.768 < 2.2e-16 ***
## MNA_share_10                  -0.8333403  0.1834873  -4.542 5.907e-06 ***
## Arab_share_10                 -0.4570679  0.3826752  -1.194 0.2324599
## NM_Foreigner_share_10          0.0020156  0.0189823   0.106 0.9154493
## Violent_offences               0.9311626  0.0899003  10.358 < 2.2e-16 ***
## Municipality_typeSemi-Urban   -0.6040670  0.3048769  -1.981 0.0476865 *
## Municipality_typeUrban        -1.5795338  0.3870966  -4.080 4.668e-05 ***
## Income_Category_no_imputerich -1.0185332  0.2799010  -3.639 0.0002806 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.0163 on 2042 degrees of freedom
## Multiple R-squared:  0.4369 ,  Adjusted R-squared:  0.4338
## F-statistic: 105.4 on 11 and 2042 DF,  p-value: < 6.966e-241
##
## Durbin-Watson statistic
## (original):    1.35960 , p-value: 1.017e-48
## (transformed): 2.23433 , p-value: 1e+00
```

```r
# MNA No Turks
FD_MNA_NO_TURKS_no_impute <- lm(Yes_percent_change ~ Change_MNA_NO_TURKS_Cat + Change_arabs
                 + Change_turkish + Change_NM_foreigners + SVP_change
                 + Minaret_yes_percent + Turkish_share_10 + MNA_NO_TURKS_share_10
                 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences
                 + Municipality_type + Income_Category_no_impute, data = final_merged)
dwtest(FD_MNA_NO_TURKS_no_impute)
```

```
##
##  Durbin-Watson test
##
## data:  FD_MNA_NO_TURKS_no_impute
## DW = 1.3602, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
Corrected_FD_MNA_NO_TURKS_no_impute <- cochrane.orcutt(FD_MNA_NO_TURKS_no_impute)
summary(Corrected_FD_MNA_NO_TURKS_no_impute)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_MNA_NO_TURKS_Cat + Change_arabs +
##     Change_turkish + Change_NM_foreigners + SVP_change + Minaret_yes_percent +
##     Turkish_share_10 + MNA_NO_TURKS_share_10 + Arab_share_10 +
##     NM_Foreigner_share_10 + Violent_offences + Municipality_type +
##     Income_Category_no_impute, data = final_merged)
##
##                                       Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                         22.6545753  1.2078917  18.755 < 2.2e-16
## Change_MNA_NO_TURKS_CatHigh Increase -1.1944599  0.3070609  -3.890 0.0001035
## Change_MNA_NO_TURKS_CatLow Increase  -0.9580061  0.2824902  -3.391 0.0007089
```

```
## Change_MNA_NO_TURKS_CatLow Decrease  -0.3338339  0.4580383  -0.729 0.4661870
## Change_MNA_NO_TURKS_CatHigh Decrease  0.5238215  0.5158726   1.015 0.3100318
## Change_arabs                        -0.8191444  0.3246007  -2.524 0.0116934
## Change_turkish                       0.6323837  0.4365086   1.449 0.1475666
## Change_NM_foreigners                 0.0845363  0.0372853   2.267 0.0234773
## SVP_change                           0.0786774  0.0152598   5.156 2.769e-07
## Minaret_yes_percent                 -0.4824459  0.0147958 -32.607 < 2.2e-16
## Turkish_share_10                    -0.5241918  0.2218471  -2.363 0.0182284
## MNA_NO_TURKS_share_10               -0.3742359  0.5541105  -0.675 0.4995102
## Arab_share_10                       -0.4590782  0.4102492  -1.119 0.2632625
## NM_Foreigner_share_10                0.0048039  0.0189471   0.254 0.7998754
## Violent_offences                     0.9180140  0.0895556  10.251 < 2.2e-16
## Municipality_typeSemi-Urban         -0.6090279  0.3033728  -2.008 0.0448259
## Municipality_typeUrban              -1.5551252  0.3870297  -4.018 6.080e-05
## Income_Category_no_imputerich       -1.0355862  0.2790724  -3.711 0.0002121
##
## (Intercept)                         ***
## Change_MNA_NO_TURKS_CatHigh Increase ***
## Change_MNA_NO_TURKS_CatLow Increase  ***
## Change_MNA_NO_TURKS_CatLow Decrease
## Change_MNA_NO_TURKS_CatHigh Decrease
## Change_arabs                        *
## Change_turkish
## Change_NM_foreigners                *
## SVP_change                          ***
## Minaret_yes_percent                 ***
## Turkish_share_10                    *
## MNA_NO_TURKS_share_10
## Arab_share_10
## NM_Foreigner_share_10
## Violent_offences                    ***
## Municipality_typeSemi-Urban         *
## Municipality_typeUrban              ***
## Income_Category_no_imputerich       ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.9956 on 2040 degrees of freedom
## Multiple R-squared:  0.4431 ,  Adjusted R-squared:  0.4396
## F-statistic: 95.3 on 13 and 2040 DF,  p-value: < 9.422e-244
##
## Durbin-Watson statistic
## (original):    1.36023 , p-value: 1.168e-48
## (transformed): 2.22944 , p-value: 1e+00
```

- slightly reduced significance for the Arab model, but broadly similar.
- only very minor changes in the (significant) estimates for the MNA/NAM model
- only very minor changes in the (significant) estimates for the model excluding Turks

5) Models without Outliers   Arabs

```r
# Calculating diagnostics
hat_values_Arabs <- hatvalues(FD_Arabs)
```

```r
cooks_dist_Arabs <- cooks.distance(FD_Arabs)
std_res_Arabs <- rstandard(FD_Arabs)

# Identifying outliers, using standard thresholds
high_leverage_Arabs <- which(hat_values_Arabs > 2 * mean(hat_values_Arabs))
high_cooks_Arabs <- which(cooks_dist_Arabs > 4/(nrow(final_merged)-length(coef(FD_Arabs))))
high_std_res_Arabs <- which(abs(std_res_Arabs) > 3)
outliers_Arabs <- unique(c(high_leverage_Arabs, high_cooks_Arabs, high_std_res_Arabs))


# Refitting the model without outliers
Arabs_no_outliers <- final_merged[-outliers_Arabs, ]
FD_Arabs_no_outliers <- lm(Yes_percent_change ~ Change_Arabs_Cat2 + Change_MNA
                    + Change_NM_foreigners + SVP_change + Minaret_yes_percent
                    + MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10
                    + Violent_offences + Municipality_type + Income_Category,
                    data = Arabs_no_outliers)
dwtest(FD_Arabs_no_outliers)
```

```
##
##  Durbin-Watson test
##
## data:  FD_Arabs_no_outliers
## DW = 1.4131, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
Corrected_Arabs_no_outliers <- cochrane.orcutt(FD_Arabs_no_outliers)
summary(Corrected_Arabs_no_outliers)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_Arabs_Cat2 + Change_MNA +
##     Change_NM_foreigners + SVP_change + Minaret_yes_percent +
##     MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category, data = Arabs_no_outliers)
##
##                                  Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                     23.116629   1.118786  20.662 < 2.2e-16 ***
## Change_Arabs_Cat2High Increase -1.070680   0.286900  -3.732 0.0001956 ***
## Change_Arabs_Cat2Low Increase  -0.675310   0.264427  -2.554 0.0107306 *
## Change_Arabs_Cat2Low Decrease  -0.639693   0.394333  -1.622 0.1049224
## Change_Arabs_Cat2High Decrease  0.679262   0.460252   1.476 0.1401500
## Change_MNA                      0.018548   0.293432   0.063 0.9496058
## Change_NM_foreigners            0.056608   0.038334   1.477 0.1399174
## SVP_change                      0.113068   0.015915   7.105 1.696e-12 ***
## Minaret_yes_percent            -0.502860   0.013598 -36.980 < 2.2e-16 ***
## MNA_share_10                   -0.909384   0.210107  -4.328 1.581e-05 ***
## Arab_share_10                  -0.312556   0.564777  -0.553 0.5800445
## NM_Foreigner_share_10           0.040323   0.019404   2.078 0.0378362 *
## Violent_offences                1.079054   0.082620  13.060 < 2.2e-16 ***
## Municipality_typeSemi-Urban    -0.956050   0.281723  -3.394 0.0007039 ***
## Municipality_typeUrban         -2.027806   0.369390  -5.490 4.566e-08 ***
## Income_Categoryrich            -0.957310   0.260439  -3.676 0.0002437 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.464 on 1914 degrees of freedom
## Multiple R-squared:  0.5388 ,  Adjusted R-squared:  0.5361
## F-statistic: 148.8 on 11 and 1914 DF,  p-value: < 2.443e-307
##
## Durbin-Watson statistic
## (original):    1.41311 , p-value: 7.354e-39
## (transformed): 2.15811 , p-value: 9.998e-01
```

Remarkably, the estimates for decreasing shares are no longer significant. Minor changes in those for increasing shares.

MNA

```r
# Calculating diagnostics
hat_values_MNA <- hatvalues(FD_MNA)
cooks_dist_MNA <- cooks.distance(FD_MNA)
std_res_MNA <- rstandard(FD_MNA)

# Identifying outliers, using standard thresholds
high_leverage_MNA <- which(hat_values_MNA > 2 * mean(hat_values_MNA))
high_cooks_MNA <- which(cooks_dist_MNA > 4/
                             (nrow(final_merged)-
                                length(coef(FD_MNA))))
high_std_res_MNA <- which(abs(std_res_MNA) > 3)
outliers_MNA <- unique(c(high_leverage_MNA,
                            high_cooks_MNA, high_std_res_MNA))


# Refitting the model without outliers
MNA_no_outliers <- final_merged[-outliers_MNA, ]
FD_MNA_no_outliers <- lm(Yes_percent_change ~ Change_MNA_Cat2 + Change_arabs
                 + Change_NM_foreigners + SVP_change
                 + Minaret_yes_percent + MNA_share_10 + Arab_share_10
                 + NM_Foreigner_share_10 + Violent_offences
                 + Municipality_type + Income_Category, data = MNA_no_outliers)
dwtest(FD_MNA_no_outliers)
```

```
##
##  Durbin-Watson test
##
## data:  FD_MNA_no_outliers
## DW = 1.4031, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
Corrected_FD_MNA_no_outliers <- cochrane.orcutt(FD_MNA_no_outliers)
summary(Corrected_FD_MNA_no_outliers)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_MNA_Cat2 + Change_arabs +
##     Change_NM_foreigners + SVP_change + Minaret_yes_percent +
```

```
##     MNA_share_10 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences +
##     Municipality_type + Income_Category, data = MNA_no_outliers)
##
##                               Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                  23.636212   1.124268  21.024 < 2.2e-16 ***
## Change_MNA_Cat2High Increase -0.432167   0.292701  -1.476  0.139979
## Change_MNA_Cat2Low Increase  -0.841090   0.275398  -3.054  0.002289 **
## Change_MNA_Cat2Low Decrease  -0.485379   0.357335  -1.358  0.174519
## Change_MNA_Cat2High Decrease  0.242711   0.432590   0.561  0.574820
## Change_arabs                 -1.035470   0.348361  -2.972  0.002992 **
## Change_NM_foreigners          0.041761   0.038119   1.096  0.273418
## SVP_change                    0.115397   0.016022   7.202 8.475e-13 ***
## Minaret_yes_percent          -0.508631   0.013587 -37.436 < 2.2e-16 ***
## MNA_share_10                 -0.969893   0.220988  -4.389 1.202e-05 ***
## Arab_share_10                 0.097012   0.518239   0.187  0.851527
## NM_Foreigner_share_10         0.029633   0.018866   1.571  0.116421
## Violent_offences              1.098092   0.082110  13.373 < 2.2e-16 ***
## Municipality_typeSemi-Urban  -0.870613   0.278361  -3.128  0.001789 **
## Municipality_typeUrban       -1.936473   0.365365  -5.300 1.291e-07 ***
## Income_Categoryrich          -1.055962   0.257142  -4.107 4.186e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.4308 on 1913 degrees of freedom
## Multiple R-squared:  0.5435 ,  Adjusted R-squared:  0.5409
## F-statistic: 151.5 on 11 and 1913 DF,  p-value: < 0e+00
##
## Durbin-Watson statistic
## (original):    1.40306 , p-value: 4.085e-40
## (transformed): 2.17261 , p-value: 9.999e-01
```

No meaningful difference after the removal of outliers (very small reduction in estimate size for the single significant level). Still, note the now negative coefficients for all levels of the change in Muslims.

MNA No Turks

```
# Calculating diagnostics
hat_values_No_Turks <- hatvalues(FD_MNA_NO_TURKS)
cooks_dist_No_Turks <- cooks.distance(FD_MNA_NO_TURKS)
std_res_No_Turks <- rstandard(FD_MNA_NO_TURKS)

# Identifying outliers, using standard thresholds
high_leverage_No_Turks <- which(hat_values_No_Turks > 2 * mean(hat_values_No_Turks))
high_cooks_No_Turks <- which(cooks_dist_No_Turks > 4/
                             (nrow(final_merged)-
                                length(coef(FD_MNA_NO_TURKS))))
high_std_res_No_Turks <- which(abs(std_res_No_Turks) > 3)
outliers_No_Turks <- unique(c(high_leverage_No_Turks,
                            high_cooks_No_Turks, high_std_res_No_Turks))


# Refitting the model without outliers
No_Turks_no_outliers <- final_merged[-outliers_No_Turks, ]
FD_MNA_NO_TURKS_no_outliers <- lm(Yes_percent_change ~ Change_MNA_NO_TURKS_Cat + Change_arabs
```

```
                 + Change_turkish + Change_NM_foreigners + SVP_change
                 + Minaret_yes_percent + Turkish_share_10 + MNA_NO_TURKS_share_10
                 + Arab_share_10 + NM_Foreigner_share_10 + Violent_offences
                 + Municipality_type + Income_Category, data = No_Turks_no_outliers)
dwtest(FD_MNA_NO_TURKS_no_outliers)
```

```
##
##  Durbin-Watson test
##
## data:  FD_MNA_NO_TURKS_no_outliers
## DW = 1.4011, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```
Corrected_FD_MNA_NO_TURKS_no_outliers <- cochrane.orcutt(FD_MNA_NO_TURKS_no_outliers)
summary(Corrected_FD_MNA_NO_TURKS_no_outliers)
```

```
## Call:
## lm(formula = Yes_percent_change ~ Change_MNA_NO_TURKS_Cat + Change_arabs +
##     Change_turkish + Change_NM_foreigners + SVP_change + Minaret_yes_percent +
##     Turkish_share_10 + MNA_NO_TURKS_share_10 + Arab_share_10 +
##     NM_Foreigner_share_10 + Violent_offences + Municipality_type +
##     Income_Category, data = No_Turks_no_outliers)
##
##                                       Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                          23.037537   1.131265  20.364 < 2.2e-16 ***
## Change_MNA_NO_TURKS_CatHigh Increase -1.289310   0.291068  -4.430 9.980e-06 ***
## Change_MNA_NO_TURKS_CatLow Increase  -1.028723   0.262084  -3.925 8.977e-05 ***
## Change_MNA_NO_TURKS_CatLow Decrease  -0.299923   0.452201  -0.663  0.507250
## Change_MNA_NO_TURKS_CatHigh Decrease  0.414648   0.543270   0.763  0.445413
## Change_arabs                         -0.982111   0.363956  -2.698  0.007029 **
## Change_turkish                        1.078186   0.487931   2.210  0.027245 *
## Change_NM_foreigners                  0.045568   0.038224   1.192  0.233358
## SVP_change                            0.103853   0.015710   6.611 4.960e-11 ***
## Minaret_yes_percent                  -0.499513   0.013744 -36.343 < 2.2e-16 ***
## Turkish_share_10                     -0.762903   0.263683  -2.893  0.003856 **
## MNA_NO_TURKS_share_10                -0.616400   0.787627  -0.783  0.433958
## Arab_share_10                         0.083047   0.567381   0.146  0.883645
## NM_Foreigner_share_10                 0.045460   0.019159   2.373  0.017755 *
## Violent_offences                      1.090963   0.084605  12.895 < 2.2e-16 ***
## Municipality_typeSemi-Urban          -0.901721   0.283892  -3.176  0.001516 **
## Municipality_typeUrban               -2.060315   0.371599  -5.544 3.362e-08 ***
## Income_Categoryrich                  -1.057746   0.260701  -4.057 5.166e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.4778 on 1899 degrees of freedom
## Multiple R-squared:  0.539 ,  Adjusted R-squared:  0.5358
## F-statistic: 130.3 on 13 and 1899 DF,  p-value: < 3.627e-303
##
## Durbin-Watson statistic
## (original):    1.40112 , p-value: 3.699e-40
## (transformed): 2.15713 , p-value: 9.998e-01
```

The results are broadly unchanged.

End of Part 11

Part 12: Remaining Appendix Content, Model Diagnostics, & Extra Content

12.1 Descriptive Statistics

Computed for all variables in our models and other relevant variables used to build these models. Missing values were already calculated at the start of the script.

```
# Numerical variables - Referendum Results
summary(final_merged$Minaret_yes_percent)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   28.40   55.83   64.08   63.05   71.16   96.00
```

```
sd(final_merged$Minaret_yes_percent, na.rm=TRUE)
```

```
## [1] 10.78459
```

```
summary(final_merged$FaceBan_yes_percent)
```

```
## Warning: Unknown or uninitialised column: `FaceBan_yes_percent`.
```

```
## Length  Class   Mode
##      0   NULL   NULL
```

```
sd(final_merged$FaceBan_yes_percent, na.rm=TRUE)
```

```
## Warning: Unknown or uninitialised column: `FaceBan_yes_percent`.
```

```
## [1] NA
```

```
summary(final_merged$Yes_percent_change)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -41.538 -11.950  -7.655  -5.677   0.357  29.689
```

```
sd(final_merged$Yes_percent_change, na.rm=TRUE)
```

```
## [1] 8.65449
```

```
# Numerical variables - Changes in Demographics
summary(final_merged$Change_arabs)
```

```
##     Min.   1st Qu.   Median     Mean   3rd Qu.     Max.
## -4.17500  0.00000  0.06545  0.15160  0.28726  2.77769
```

```r
sd(final_merged$Change_arabs, na.rm=TRUE)
```

```
## [1] 0.3717614
```

```r
summary(final_merged$Change_MNA)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.66617 -0.01001  0.07449  0.13057  0.32148  3.44430
```

```r
sd(final_merged$Change_MNA, na.rm=TRUE)
```

```
## [1] 0.4553492
```

```r
summary(final_merged$Change_MNA_NO_TURKS)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.68852  0.00000  0.09268  0.15448  0.28734  2.83454
```

```r
sd(final_merged$Change_MNA_NO_TURKS, na.rm=TRUE)
```

```
## [1] 0.3270956
```

```r
summary(final_merged$Change_NM_foreigners)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -15.400   1.011   2.614   2.860   4.470  23.472
```

```r
sd(final_merged$Change_NM_foreigners, na.rm=TRUE)
```

```
## [1] 3.010568
```

```r
# Numerical variables - Initial Demographics
summary(final_merged$Muslim_share_10)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.08905 0.36265 0.77271 0.98285 9.46669
```

```r
sd(final_merged$Muslim_share_10, na.rm=TRUE)
```

```
## [1] 1.114674
```

```r
summary(final_merged$Arab_share_10)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.1006  0.2101  0.2736  5.3958
```

```r
sd(final_merged$Arab_share_10, na.rm=TRUE)
```

```
## [1] 0.3579691
```

```r
summary(final_merged$MNA_share_10)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  0.2101  0.5626  0.6699  8.8456
```

```r
sd(final_merged$MNA_share_10, na.rm=TRUE)
```

```
## [1] 0.9256411
```

```r
summary(final_merged$MNA_NO_TURKS_share_10)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.07776 0.17060 0.22797 3.68852
```

```r
sd(final_merged$MNA_NO_TURKS_share_10, na.rm=TRUE)
```

```
## [1] 0.2884326
```

```r
summary(final_merged$NM_Foreigner_share_10)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   7.074  11.361  13.560  18.693  55.535
```

```r
sd(final_merged$NM_Foreigner_share_10, na.rm=TRUE)
```

```
## [1] 8.700986
```

```r
# Categorical variables - Changes in Demographics
summary(final_merged$Change_Muslim_Cat)
```

```
##     No Change High Increase  Low Increase  Low Decrease High Decrease
##           391           654           654           213           213
```

```r
summary(final_merged$Change_Arabs_Cat2)
```

```
##     No Change High Increase  Low Increase  Low Decrease High Decrease
##           675           552           553           173           172
```

```r
summary(final_merged$Change_MNA_Cat2)
```

```
##     No Change High Increase  Low Increase  Low Decrease High Decrease
##           523           567           567           234           234
```

```r
summary(final_merged$Change_MNA_NO_TURKS_Cat)
```

```
##     No Change High Increase  Low Increase  Low Decrease High Decrease
##           642           605           605           137           136
```

```r
# Covariates
summary(final_merged$SVP_change)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
## -49.020  -5.905  -2.455  -1.002   2.060  76.534      12
```

```r
sd(final_merged$SVP_change, na.rm=TRUE)
```

```
## [1] 9.907491
```

```r
summary(final_merged$Violent_offences)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   3.000   4.200   4.472   5.600  14.500
```

```r
sd(final_merged$Violent_offences, na.rm=TRUE)
```

```
## [1] 1.916616
```

```r
table(final_merged$Municipality_type)
```

```
##
##      Rural Semi-Urban      Urban
##        688        926        511
```

```r
table(final_merged$Income_Category)
```

```
##
## poor rich
## 1060 1065
```

```r
summary(final_merged$SVP_percent_07)
```

```
## Warning: Unknown or uninitialised column: `SVP_percent_07`.
```

```
## Length  Class   Mode
##      0   NULL   NULL
```

```r
sd(final_merged$SVP_percent_07, na.rm=TRUE)
```

```
## Warning: Unknown or uninitialised column: `SVP_percent_07`.
```

```
## [1] NA
```

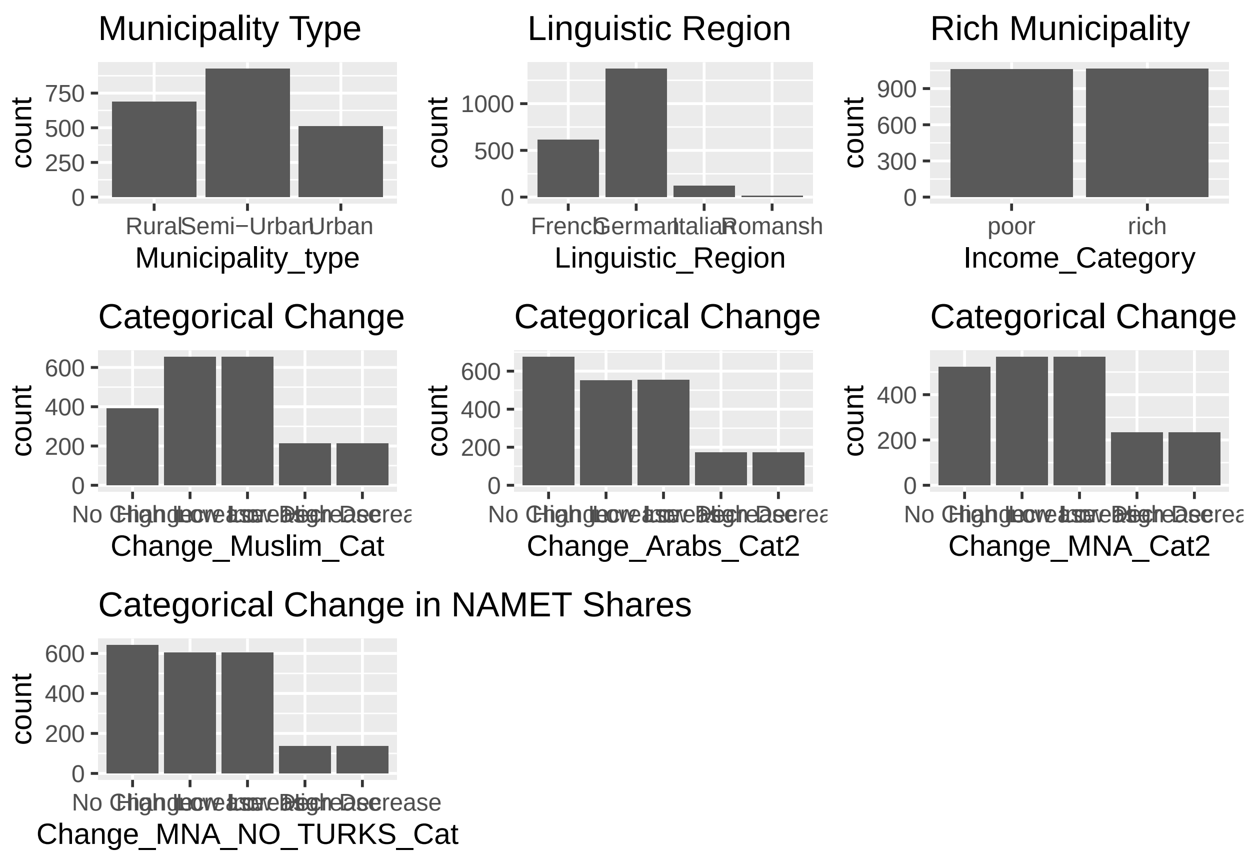
```r
summary(final_merged$Pop_density_change_in_hundreds)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -2.51441  0.03413  0.18980  0.49049  0.54572 13.62235
```

```r
sd(final_merged$Pop_density_change_in_hundreds, na.rm=TRUE)
```

```
## [1] 1.006808
```

```r
# Bar charts for the classes of categorical variables
p1 <- ggplot(final_merged, aes(x = Municipality_type)) +
  geom_bar() +
  ggtitle("Municipality Type")

p2 <- ggplot(final_merged, aes(x = Linguistic_Region)) +
  geom_bar() +
  ggtitle("Linguistic Region")

p3 <- ggplot(final_merged, aes(x = Income_Category)) +
  geom_bar() +
  ggtitle("Rich Municipality")

p4 <- ggplot(final_merged, aes(x = Change_Muslim_Cat)) +
  geom_bar() +
  ggtitle("Categorical Change in Muslim Shares")

p5 <- ggplot(final_merged, aes(x = Change_Arabs_Cat2)) +
  geom_bar() +
  ggtitle("Categorical Change in Arab Shares")

p6 <- ggplot(final_merged, aes(x = Change_MNA_Cat2)) +
  geom_bar() +
  ggtitle("Categorical Change in NAM Shares")

p7 <- ggplot(final_merged, aes(x = Change_MNA_NO_TURKS_Cat)) +
  geom_bar() +
  ggtitle("Categorical Change in NAMET Shares")

grid_figure <- gridExtra::grid.arrange(p1, p2, p3, p4, p5, p6, p7, nrow = 3, ncol = 3)
```



12.2 WLS and HC1 Main Specification:

WLS Main Specification

```r
# Starting by running the WLS model of our main specification, pre-autocorrelation correction

# Define weights to use
wt <- 1 / lm(abs(FD_Muslims_Final2$residuals) ~ FD_Muslims_Final2$fitted.values)$fitted.values^2

# Perform weighted least squares
WLS_main_specification <- lm(Yes_percent_change ~ Change_Muslim_Cat +
                               Change_NM_foreigners +  SVP_change_imputed +
                               Minaret_yes_percent + Muslim_share_10 +
                               NM_Foreigner_share_10 + Violent_offences +
                               Municipality_type + Income_Category,
                             data = final_merged, weights = wt)
summary(WLS_main_specification)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change_imputed + Minaret_yes_percent + Muslim_share_10 +
##     NM_Foreigner_share_10 + Violent_offences + Municipality_type +
##     Income_Category, data = final_merged, weights = wt)
```

```
##
## Weighted Residuals:
##     Min      1Q  Median      3Q     Max
## -6.9697 -0.7859 -0.0046  0.8461  5.5594
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   26.57074    1.14287  23.249  < 2e-16 ***
## Change_Muslim_CatHigh Increase -1.00272    0.36580  -2.741  0.00617 **
## Change_Muslim_CatLow Increase  -0.79671    0.35586  -2.239  0.02527 *
## Change_Muslim_CatLow Decrease  -0.04237    0.47158  -0.090  0.92842
## Change_Muslim_CatHigh Decrease  1.65459    0.51432   3.217  0.00131 **
## Change_NM_foreigners            0.09171    0.04047   2.266  0.02354 *
## SVP_change_imputed              0.05947    0.01219   4.879 1.14e-06 ***
## Minaret_yes_percent            -0.55027    0.01364 -40.349  < 2e-16 ***
## Muslim_share_10                -1.28441    0.14991  -8.568  < 2e-16 ***
## NM_Foreigner_share_10           0.04358    0.01837   2.373  0.01773 *
## Violent_offences                1.06152    0.07121  14.906  < 2e-16 ***
## Municipality_typeSemi-Urban    -0.88553    0.30080  -2.944  0.00328 **
## Municipality_typeUrban         -1.97118    0.40150  -4.910 9.83e-07 ***
## Income_Categoryrich            -1.71981    0.27860  -6.173 8.01e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.296 on 2111 degrees of freedom
## Multiple R-squared:  0.5796, Adjusted R-squared:  0.577
## F-statistic: 223.9 on 13 and 2111 DF,  p-value: < 2.2e-16
```

```r
# Note that the model fit has improved, with the RSE going from over 5 to 1.3.

# Compare heteroskedasticity to original level.
bptest(WLS_main_specification)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  WLS_main_specification
## BP = 32.151, df = 13, p-value = 0.002282
```

```r
# However, heteroskedasticity is still problematic, as per the Breusch-Pagan test,
# which p-value is well below 0.05.
```

HC1 Main Specification

```r
# Calculate HC1 standard errors
robust_se <- vcovHC(FD_Muslims_Final2, type = "HC1")


# Summarise the model with robust standard errors
coeftest(FD_Muslims_Final2, vcov = robust_se)
```

```
##
```

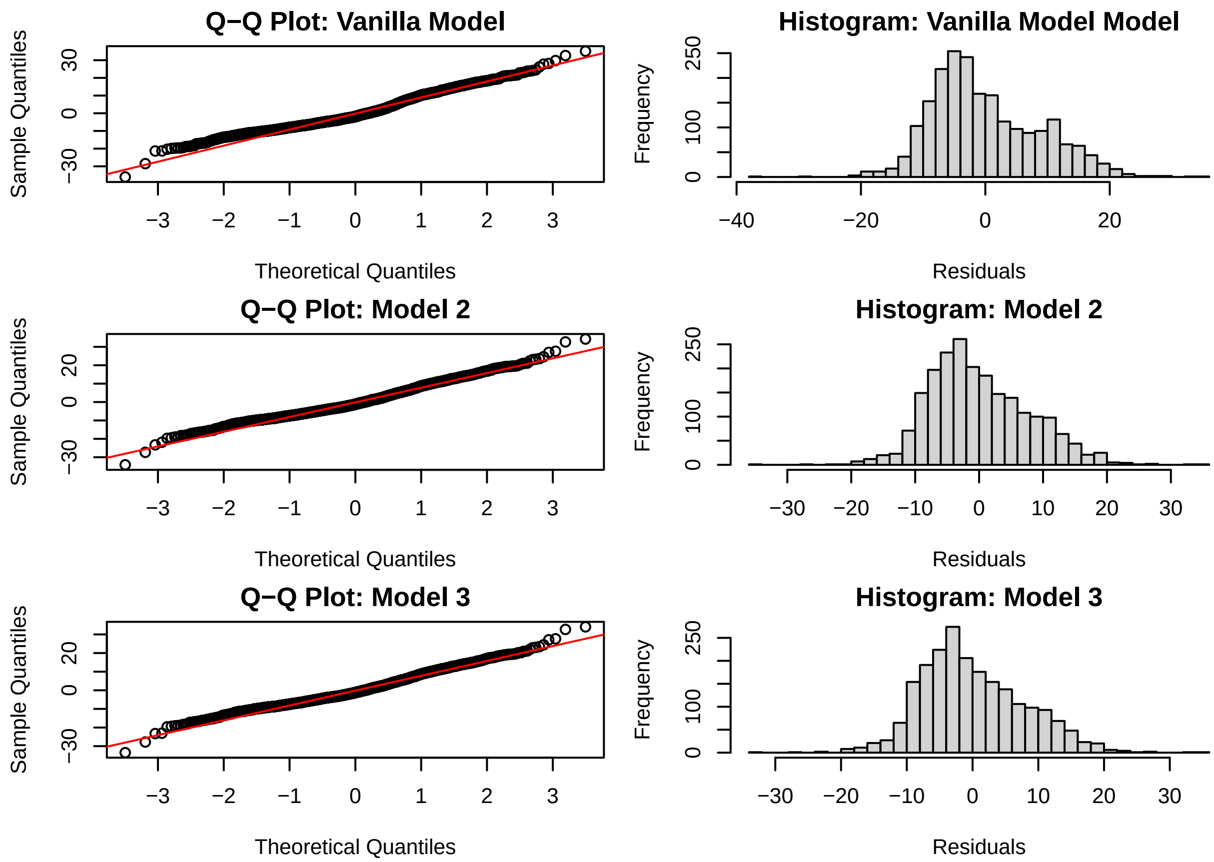
```
## t test of coefficients:
##
##                                Estimate Std. Error  t value  Pr(>|t|)
## (Intercept)                   27.396796   1.278775  21.4243 < 2.2e-16 ***
## Change_Muslim_CatHigh Increase -0.986446   0.397835  -2.4795  0.013233 *
## Change_Muslim_CatLow Increase  -0.700647   0.390084  -1.7961  0.072614 .
## Change_Muslim_CatLow Decrease   0.039968   0.478356   0.0836  0.933420
## Change_Muslim_CatHigh Decrease  1.843707   0.576028   3.2007  0.001391 **
## Change_NM_foreigners            0.099804   0.047761   2.0897  0.036769 *
## SVP_change_imputed              0.063162   0.015640   4.0384 5.573e-05 ***
## Minaret_yes_percent            -0.564813   0.015183 -37.2006 < 2.2e-16 ***
## Muslim_share_10                -1.360126   0.152386  -8.9255 < 2.2e-16 ***
## NM_Foreigner_share_10           0.044848   0.020609   2.1762  0.029654 *
## Violent_offences                1.082518   0.080750  13.4058 < 2.2e-16 ***
## Municipality_typeSemi-Urban    -0.838877   0.324215  -2.5874  0.009736 **
## Municipality_typeUrban         -1.946176   0.397649  -4.8942 1.062e-06 ***
## Income_Categoryrich            -1.848705   0.285063  -6.4852 1.100e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Barely any difference with the original output. This suggests that heteroskedasticity
# may not be an important issue, particularly when combined with the low impact of WLS.
```

12.3 Normality of Residuals

12.3.1 Initial 3 Muslim Models

```
# Set up the plotting area to handle 6 plots
par(mfrow = c(3, 2), mar = c(4, 4, 2, 1))

# Vanilla Model
mna_residuals <- resid(Corrected_FD_Muslims_Vanilla)
qqnorm(mna_residuals, main = "Q-Q Plot: Vanilla Model")
qqline(mna_residuals, col = "red")
hist(mna_residuals, main = "Histogram: Vanilla Model Model", xlab = "Residuals", breaks = 30)

# Main Specification Model with Time-Constant Controls but NO Time-Varying Ones
arab_residuals <- resid(Corrected_FD_Muslims_Final2_simplified2)
qqnorm(arab_residuals, main = "Q-Q Plot: Model 2")
qqline(arab_residuals, col = "red")
hist(arab_residuals, main = "Histogram: Model 2", xlab = "Residuals", breaks = 30)

# Main Specification Model with Time-Constant Controls AND Time-Varying Ones
main_spec_residuals <- resid(Corrected_FD_Muslims_Final2_simplified1)
qqnorm(main_spec_residuals, main = "Q-Q Plot: Model 3")
qqline(main_spec_residuals, col = "red")
hist(main_spec_residuals, main = "Histogram: Model 3", xlab = "Residuals", breaks = 30)
```

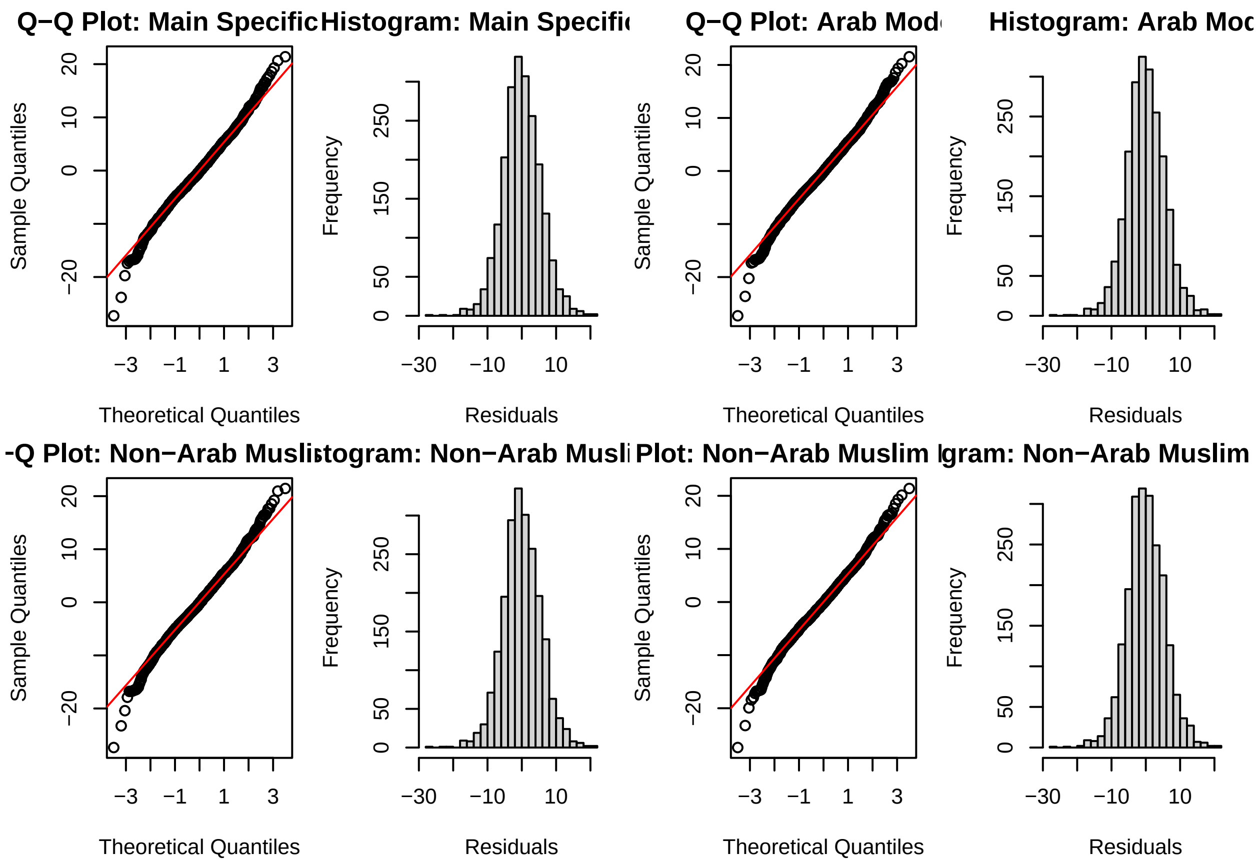



```r
par(mfrow = c(1, 1))
```

- Normality increases with the number of controls ### 12.3.2 Figure for only the Main Specification and Muslim Sub-Groups

```r
# Qq-plot to assess normality of residuals.
# Set up the plotting area to handle 8 plots
par(mfrow = c(2, 4), mar = c(4, 4, 2, 1))

# Main Specification Model
main_spec_residuals <- resid(Corrected_FD_Muslims_Final2)
qqnorm(main_spec_residuals, main = "Q-Q Plot: Main Specification")
qqline(main_spec_residuals, col = "red")
hist(main_spec_residuals, main = "Histogram: Main Specification", xlab = "Residuals", breaks = 30)

# Arab Model
arab_residuals <- resid(Corrected_FD_Arabs)
qqnorm(arab_residuals, main = "Q-Q Plot: Arab Model")
qqline(arab_residuals, col = "red")
hist(arab_residuals, main = "Histogram: Arab Model", xlab = "Residuals", breaks = 30)

# Non-Arab Muslim Model
mna_residuals <- resid(Corrected_FD_MNA)
qqnorm(mna_residuals, main = "Q-Q Plot: Non-Arab Muslim Model")
qqline(mna_residuals, col = "red")
```

```r
hist(mna_residuals, main = "Histogram: Non-Arab Muslim Model", xlab = "Residuals", breaks = 30)

# Non-Arab Muslim Excluding Turks Model
mna_no_turks_residuals <- resid(Corrected_FD_MNA_NO_TURKS)
qqnorm(mna_no_turks_residuals, main = "Q-Q Plot: Non-Arab Muslim Excl. Turks")
qqline(mna_no_turks_residuals, col = "red")
hist(mna_no_turks_residuals, main = "Histogram: Non-Arab Muslim Excl. Turks",
     xlab = "Residuals", breaks = 30)
```




```r
par(mfrow = c(1, 1))
```

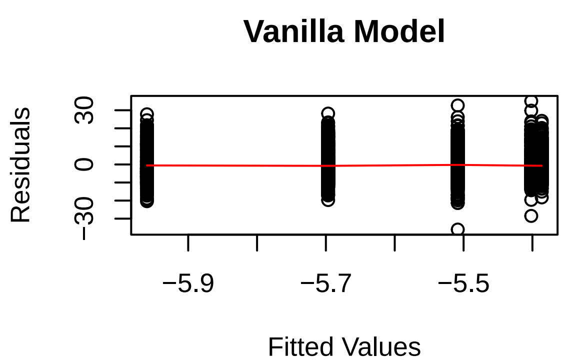
- We see that the addition of the lagged/baseline values in the main specification has a clear positive effect on the normality of the residuals.

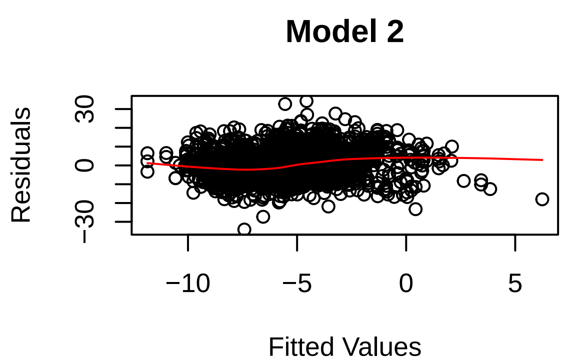
12.4 Linearity Assumption

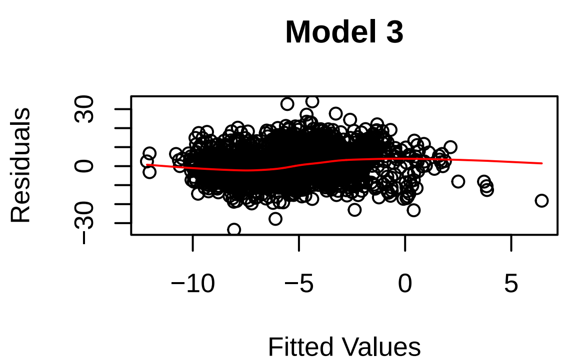
12.4.1 Linearity Assumption for the Initial 3 Muslim Models

```r
# Set up the plotting area
par(mfrow = c(2, 2))

# Vanilla Model plot
```

```r
plot(Corrected_FD_Muslims_Vanilla$fitted.values,
     Corrected_FD_Muslims_Vanilla$residuals,
     main = "Vanilla Model",
     xlab = "Fitted Values", ylab = "Residuals")
lines(lowess(Corrected_FD_Muslims_Vanilla$fitted.values,
             Corrected_FD_Muslims_Vanilla$residuals), col = "red")

# Main Specification Model with Time-Constant Controls but NO Time-Varying Ones
plot(Corrected_FD_Muslims_Final2_simplified2$fitted.values,
     Corrected_FD_Muslims_Final2_simplified2$residuals,
     main = "Model 2",
     xlab = "Fitted Values", ylab = "Residuals")
lines(lowess(Corrected_FD_Muslims_Final2_simplified2$fitted.values,
             Corrected_FD_Muslims_Final2_simplified2$residuals), col = "red")

# Main Specification Model with Time-Constant Controls AND Time-Varying Ones
plot(Corrected_FD_Muslims_Final2_simplified1$fitted.values,
     Corrected_FD_Muslims_Final2_simplified1$residuals,
     main = "Model 3",
     xlab = "Fitted Values", ylab = "Residuals")
lines(lowess(Corrected_FD_Muslims_Final2_simplified1$fitted.values,
             Corrected_FD_Muslims_Final2_simplified1$residuals), col = "red")
```

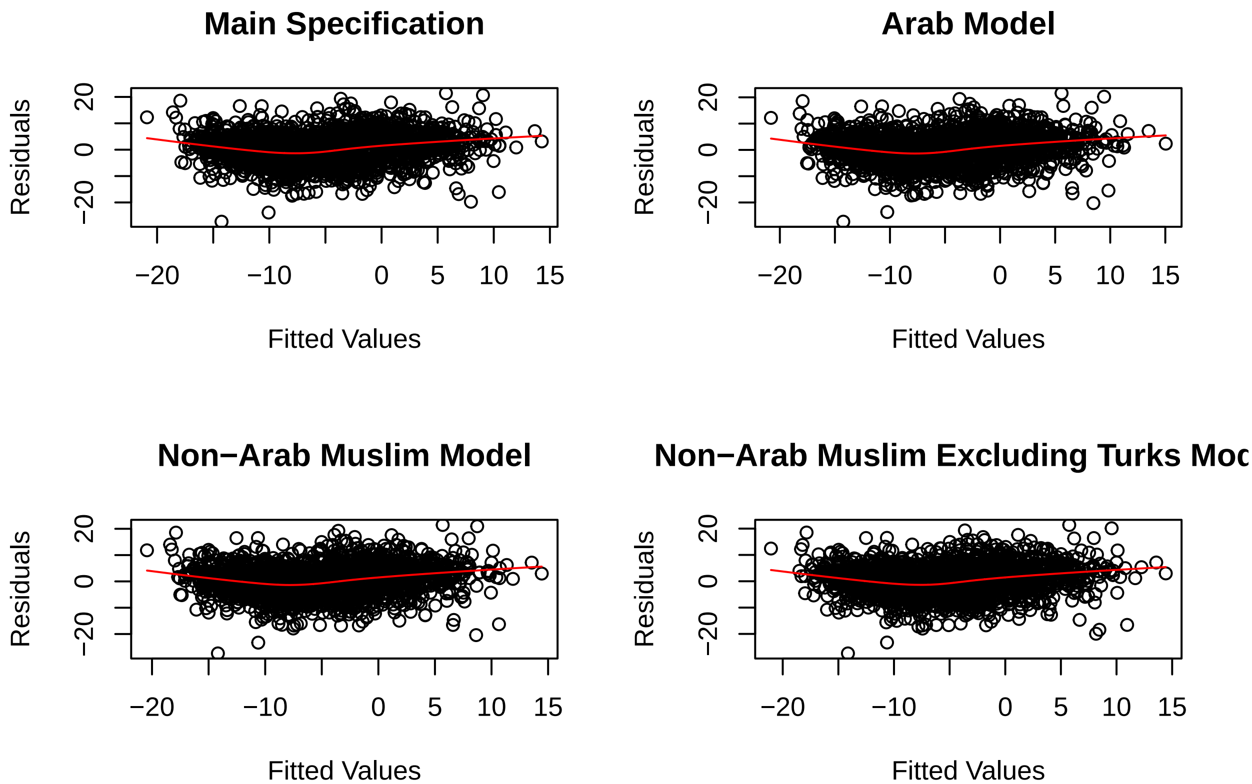






12.4.2 Linearity Assumption for Main Specificiation and Sub-Group Models

```
# Set up the plotting area
par(mfrow = c(2, 2))

# Main specification plot
plot(Corrected_FD_Muslims_Final2$fitted.values, Corrected_FD_Muslims_Final2$residuals,
     main = "Main Specification",
     xlab = "Fitted Values", ylab = "Residuals")
lines(lowess(Corrected_FD_Muslims_Final2$fitted.values,
             Corrected_FD_Muslims_Final2$residuals), col = "red")

# Arab Model plot
plot(Corrected_FD_Arabs$fitted.values, Corrected_FD_Arabs$residuals,
     main = "Arab Model",
     xlab = "Fitted Values", ylab = "Residuals")
lines(lowess(Corrected_FD_Arabs$fitted.values,
             Corrected_FD_Arabs$residuals), col = "red")

# Non-Arab Muslim Model plot
plot(Corrected_FD_MNA$fitted.values, Corrected_FD_MNA$residuals,
     main = "Non-Arab Muslim Model",
     xlab = "Fitted Values", ylab = "Residuals")
lines(lowess(Corrected_FD_MNA$fitted.values,
             Corrected_FD_MNA$residuals), col = "red")

# Non-Arab Muslim Excluding Turks Model plot
plot(Corrected_FD_MNA_NO_TURKS$fitted.values, Corrected_FD_MNA_NO_TURKS$residuals,
     main = "Non-Arab Muslim Excluding Turks Model",
     xlab = "Fitted Values", ylab = "Residuals")
lines(lowess(Corrected_FD_MNA_NO_TURKS$fitted.values,
             Corrected_FD_MNA_NO_TURKS$residuals), col = "red")
```




```
# Reset the plotting layout
par(mfrow = c(1, 1))
```

12.5 Main specification with Population Density:

We add a control for changes in the population density, which could artificially affect exposure to out-groups without their population shares changing. This is particularly likely for changes in population density due to the rural exodus. Alternatively, it could happen due to the urbanization of previously semi-urban/rural neighbourhoods.

```
FD_Muslims_Cat_9 <- lm(Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
                          SVP_change_imputed + Minaret_yes_percent
                        + Muslim_share_10 + NM_Foreigner_share_10 + Violent_offences
                        + Municipality_type + Income_Category + Pop_density_change_in_hundreds,
                      data = final_merged)
summary(FD_Muslims_Cat_9)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change_imputed + Minaret_yes_percent + Muslim_share_10 +
##     NM_Foreigner_share_10 + Violent_offences + Municipality_type +
##     Income_Category + Pop_density_change_in_hundreds, data = final_merged)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.3357  -3.2652  -0.0413   3.5583  20.5987
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    27.37777    1.12408  24.356  < 2e-16 ***
## Change_Muslim_CatHigh Increase -0.98271    0.36857  -2.666  0.00773 **
## Change_Muslim_CatLow Increase  -0.70105    0.35851  -1.955  0.05066 .
## Change_Muslim_CatLow Decrease   0.01976    0.47361   0.042  0.96672
## Change_Muslim_CatHigh Decrease  1.78915    0.51556   3.470  0.00053 ***
## Change_NM_foreigners            0.10412    0.04061   2.564  0.01041 *
## SVP_change_imputed              0.06249    0.01232   5.072 4.29e-07 ***
## Minaret_yes_percent            -0.56543    0.01352 -41.822  < 2e-16 ***
## Muslim_share_10                -1.31797    0.15678  -8.406  < 2e-16 ***
## NM_Foreigner_share_10           0.04888    0.01868   2.617  0.00895 **
## Violent_offences                1.08575    0.07027  15.451  < 2e-16 ***
## Municipality_typeSemi-Urban    -0.83637    0.30441  -2.748  0.00606 **
## Municipality_typeUrban         -1.85983    0.41211  -4.513 6.74e-06 ***
## Income_Categoryrich            -1.81776    0.28285  -6.427 1.61e-10 ***
## Pop_density_change_in_hundreds -0.17805    0.14686  -1.212  0.22551
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.48 on 2110 degrees of freedom
## Multiple R-squared:  0.6017, Adjusted R-squared:  0.5991
## F-statistic: 227.7 on 14 and 2110 DF,  p-value: < 2.2e-16
```

Including population density makes the coefficient for decreasing Muslim shares less significant. This may be due to population density falling as Muslim shares decrease.

We test whether this hypothesis holds:

```r
anova_model <- aov(Pop_density_change_in_hundreds ~ Change_Muslim_Cat, data = final_merged)
summary(anova_model)
```

```
##                     Df Sum Sq Mean Sq F value   Pr(>F)
## Change_Muslim_Cat    4   68.4  17.110    17.4 4.66e-14 ***
## Residuals         2120 2084.6   0.983
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# The change in Muslim population significantly affects population density
# (p < 0.001), indicating large differences in population density changes across
# Muslim share change categories.

# To investigate whether this holds for both directions of share changes,
# we conduct a Tukey test:
TukeyHSD(anova_model)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
```

```
## Fit: aov(formula = Pop_density_change_in_hundreds ~ Change_Muslim_Cat, data = final_merged)
##
## $Change_Muslim_Cat
##                                  diff         lwr         upr     p adj
## High Increase-No Change      0.5127842  0.33972105  0.68584731 0.0000000
## Low Increase-No Change       0.2832355  0.11017236  0.45629862 0.0000812
## Low Decrease-No Change       0.1766414 -0.05390793  0.40719082 0.2239226
## High Decrease-No Change      0.3340224  0.10347302  0.56457177 0.0007514
## Low Increase-High Increase  -0.2295487 -0.37925833 -0.07983904 0.0002854
## Low Decrease-High Increase  -0.3361427 -0.54971976 -0.12256571 0.0001759
## High Decrease-High Increase -0.1787618 -0.39233881  0.03481524 0.1500422
## Low Decrease-Low Increase   -0.1065940 -0.32017107  0.10698298 0.6517877
## High Decrease-Low Increase   0.0507869 -0.16279012  0.26436393 0.9668206
## High Decrease-Low Decrease   0.1573810 -0.10494959  0.41971149 0.4731992
```

Increases in the Muslim population share are associated with significant increases in population density, and similarly, *high* decreases in the share also affect population density, but to a lesser extent.

To reduce the risk of controlling for the effect of the change in Muslims through the control for the change in population density, we remove this latter variable.

12.6 Main specification with Baseline SVP Support:

This baseline SVP support is not expected to be influenced by baseline support, with only present partisanship affecting votes, not past attitudes. For robustness, we nonetheless verify this hypothesis:

```
FD_Muslims_Cat_10 <- lm(Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
                          SVP_change_imputed + Minaret_yes_percent
                        + Muslim_share_10 + NM_Foreigner_share_10 + Violent_offences
                        + Municipality_type + Income_Category + Imputed_SVP_percent_07,
                       data = final_merged)
summary(FD_Muslims_Cat_10)
```

```
##
## Call:
## lm(formula = Yes_percent_change ~ Change_Muslim_Cat + Change_NM_foreigners +
##     SVP_change_imputed + Minaret_yes_percent + Muslim_share_10 +
##     NM_Foreigner_share_10 + Violent_offences + Municipality_type +
##     Income_Category + Imputed_SVP_percent_07, data = final_merged)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.2697  -3.2733  -0.0417   3.5221  20.7597
##
## Coefficients:
##                                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    27.318643   1.133198  24.108  < 2e-16 ***
## Change_Muslim_CatHigh Increase -0.985235   0.368663  -2.672 0.007588 **
## Change_Muslim_CatLow Increase  -0.706032   0.358746  -1.968 0.049192 *
## Change_Muslim_CatLow Decrease   0.031894   0.473679   0.067 0.946323
## Change_Muslim_CatHigh Decrease  1.837586   0.513851   3.576 0.000357 ***
## Change_NM_foreigners            0.100107   0.040466   2.474 0.013444 *
## SVP_change_imputed              0.057612   0.015910   3.621 0.000300 ***
```

```
## Minaret_yes_percent            -0.559893   0.016200 -34.561  < 2e-16 ***
## Muslim_share_10                -1.352944   0.153475  -8.815  < 2e-16 ***
## NM_Foreigner_share_10           0.043138   0.018648   2.313 0.020801 *
## Violent_offences                1.083219   0.070249  15.420  < 2e-16 ***
## Municipality_typeSemi-Urban    -0.850896   0.305270  -2.787 0.005362 **
## Municipality_typeUrban         -1.977454   0.409972  -4.823 1.51e-06 ***
## Income_Categoryrich            -1.828617   0.284129  -6.436 1.52e-10 ***
## Imputed_SVP_percent_07         -0.006773   0.012298  -0.551 0.581881
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.481 on 2110 degrees of freedom
## Multiple R-squared:  0.6015, Adjusted R-squared:  0.5989
## F-statistic: 227.5 on 14 and 2110 DF,  p-value: < 2.2e-16
```

```
anova_SVP_baseline <- aov(Imputed_SVP_percent_07 ~ NM_Foreigner_share_10,
                          data = final_merged)
summary(anova_SVP_baseline)
```

```
##                         Df Sum Sq Mean Sq F value Pr(>F)
## NM_Foreigner_share_10    1  23337   23337   117.7 <2e-16 ***
## Residuals             2123 420991     198
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The initial share of NM foreigners significantly affects the initial share of SVP support. Similarly to above, we decide to only keep one variable. Given the higher relevance (foreigner presence affecting SVP support rather than the reverse) and significance of baseline NM foreigner shares, we retain this variable and discard the initial SVP support.

12.7 Geographic Distribution of Municipalities and Migrations

By Linguistic Region

```
# Calculate the total number of municipalities in each region
total_muni_per_region <- table(final_merged$Linguistic_Region)

# Calculate the number of municipalities with no change in Arabs and MNA
no_change_Arab_shares <- final_merged$Change_Arabs_Cat2 %in% c("No Change")
no_change_MNA_shares <- final_merged$Change_MNA_Cat2 %in% c("No Change")
no_change_counts_arabs <- table(final_merged$Linguistic_Region[no_change_Arab_shares])
no_change_counts_MNA <- table(final_merged$Linguistic_Region[no_change_MNA_shares])

# Calculate the number of municipalities with increases in Arabs and MNA
increase_Arab_shares <- final_merged$Change_Arabs_Cat2 %in% c("High Increase", "Low Increase")
increase_MNA_shares <- final_merged$Change_MNA_Cat2 %in% c("High Increase", "Low Increase")
increase_counts_arabs <- table(final_merged$Linguistic_Region[increase_Arab_shares])
increase_counts_MNA <- table(final_merged$Linguistic_Region[increase_MNA_shares])

# Calculate the number of municipalities with decreases in Arabs and MNA
decrease_Arab_shares <- final_merged$Change_Arabs_Cat2 %in% c("Low Decrease", "High Decrease")
decrease_MNA_shares <- final_merged$Change_MNA_Cat2 %in% c("Low Decrease", "High Decrease")
```

```r
decrease_counts_arabs <- table(final_merged$Linguistic_Region[decrease_Arab_shares])
decrease_counts_MNA <- table(final_merged$Linguistic_Region[decrease_MNA_shares])

# Calculate proportions of all municipalities in a region which correspond to
# constant shares of one of the sub-groups
prop_no_change_arabs <- no_change_counts_arabs / total_muni_per_region
prop_no_change_MNA <- no_change_counts_MNA / total_muni_per_region

# Calculate proportions of all municipalities in a region which correspond to an
# increase in the shares of one of the sub-groups
prop_increase_arabs <- increase_counts_arabs / total_muni_per_region
prop_increase_MNA <- increase_counts_MNA / total_muni_per_region

# Calculate proportions of all municipalities in a region which correspond to a
# decrease in the shares of one of the sub-groups
prop_decrease_arabs <- decrease_counts_arabs / total_muni_per_region
prop_decrease_MNA <- decrease_counts_MNA / total_muni_per_region

# Combine into a data frame for better visualisation

# For constant shares
data.frame(Region = names(total_muni_per_region),
           Total_Municipalities = as.vector(total_muni_per_region),
           No_Change_Arabs = as.vector(no_change_counts_arabs),
           No_Change_MNA = as.vector(no_change_counts_MNA),
           Prop_No_Change_Arabs = as.vector(prop_no_change_arabs),
           Prop_No_Change_MNA = as.vector(prop_no_change_MNA))
```

```
##    Region Total_Municipalities No_Change_Arabs No_Change_MNA
## 1  French                  615             196           167
## 2  German                 1374             412           302
## 3 Italian                  121              57            46
## 4 Romansh                   15              10             8
##   Prop_No_Change_Arabs Prop_No_Change_MNA
## 1            0.3186992          0.2715447
## 2            0.2998544          0.2197962
## 3            0.4710744          0.3801653
## 4            0.6666667          0.5333333
```

```r
# For increasing shares
data.frame(Region = names(total_muni_per_region),
           Total_Municipalities = as.vector(total_muni_per_region),
           Increase_Arabs = as.vector(increase_counts_arabs),
           Increase_MNA = as.vector(increase_counts_MNA),
           Prop_Increase_Arabs = as.vector(prop_increase_arabs),
           Prop_Increase_MNA = as.vector(prop_increase_MNA))
```

```
##    Region Total_Municipalities Increase_Arabs Increase_MNA Prop_Increase_Arabs
## 1  French                  615            291          325           0.4731707
## 2  German                 1374            770          746           0.5604076
## 3 Italian                  121             41           58           0.3388430
## 4 Romansh                   15              3            5           0.2000000
```

```
##   Prop_Increase_MNA
## 1         0.5284553
## 2         0.5429403
## 3         0.4793388
## 4         0.3333333
```

```r
# For decreasing shares
data.frame(Region = names(total_muni_per_region),
  Total_Municipalities = as.vector(total_muni_per_region),
  Decrease_Arabs = as.vector(decrease_counts_arabs),
  Decrease_MNA = as.vector(decrease_counts_MNA),
  Prop_Decrease_Arabs = as.vector(prop_decrease_arabs),
  Prop_Decrease_MNA = as.vector(prop_decrease_MNA))
```

```
##    Region Total_Municipalities Decrease_Arabs Decrease_MNA Prop_Decrease_Arabs
## 1  French                  615            128          123           0.2081301
## 2  German                 1374            192          326           0.1397380
## 3 Italian                  121             23           17           0.1900826
## 4 Romansh                   15              2            2           0.1333333
##   Prop_Decrease_MNA
## 1         0.2000000
## 2         0.2372635
## 3         0.1404959
## 4         0.1333333
```

```r
# If migration preferences or patterns were identical, we would expect a uniform
# distribution across regions. The scale of migrations might be different,
# but should remain constant relative to each other.
```

By Municipality Type

```r
municipality_counts <- table(final_merged$Municipality_type)
print(municipality_counts)
```

```
##
##      Rural Semi-Urban      Urban
##        688        926        511
```

```r
prop.table(municipality_counts) * 100
```

```
##
##      Rural Semi-Urban      Urban
##   32.37647   43.57647   24.04706
```

Percentage of Constant Muslim Shares by Municipality Type

```r
constant_muslim_shares <- final_merged$Change_Muslim_Cat %in% c("No Change")
table(final_merged$Municipality_type[constant_muslim_shares])
```

```
##
##      Rural Semi-Urban      Urban
##        197        158         36
```

```r
constant_counts <- table(final_merged$Municipality_type[constant_muslim_shares])
prop.table(constant_counts)*100
```

```
##
##      Rural Semi-Urban      Urban
##  50.383632  40.409207   9.207161
```

12.8 Migration flows by Muslim countries

```r
# Explaining the difference in effects between Arabs and MNA Muslims:

# Deliminating both groups in the 2021 data
data_arab21 <- data_21[, arab_countries]
data_muslim21 <- data_21[, muslim_countries]

# Find variables in data_muslim but not in data_arab
MNA_vars <- setdiff(names(data_muslim21), names(data_arab21))

# Create data set corresponding to those variables
data_MNA21 <- data_muslim21[, MNA_vars]


# Repeat for the 2010 data
data_arab10 <- data_10[, arab_countries]
data_muslim10 <- data_10[, muslim_countries]

# Create data set corresponding to those variables
data_MNA10 <- data_muslim10[, MNA_vars]


# We now clean the data sets

# For data_arab
data_arab21 <- data_arab21[1, , drop = FALSE]  # Keep only the first row
data_arab21 <- t(data_arab21)  # Transpose the row into a column
colnames(data_arab21) <- "arab_immigrants_21"  # Rename the column
data_arab10 <- data_arab10[1, , drop = FALSE]  # idem
data_arab10 <- t(data_arab10) # idem
colnames(data_arab10) <- "arab_immigrants_10"


# For data_MNA
data_MNA21<- data_MNA21[1, , drop = FALSE]  # Keep only the first row
data_MNA21 <- t(data_MNA21)  # Transpose the row into a column
colnames(data_MNA21) <- "MNA_immigrants_21"  # Rename the column
data_MNA10 <- data_MNA10[1, , drop = FALSE]  # idem
data_MNA10 <- t(data_MNA10)  # idem
colnames(data_MNA10) <- "MNA_immigrants_10"  # idem


# Calculate the change in MNA and Arab immigrants between 2021 and 2010
```

```r
change_arab_immigrants <- data_arab21 - data_arab10
change_MNA_immigrants <- data_MNA21 - data_MNA10

# Merge the data into a single dataset for Arabs and MNA Muslims
Investigation_data_arab <- data.frame(arab_immigrants_10 = data_arab10,
                                      arab_immigrants_21 = data_arab21,
                                      change_arab_immigrants)

Investigation_data_MNA <- data.frame(MNA_immigrants_10 = data_MNA10,
                                     MNA_immigrants_21 = data_MNA21,
                                     change_MNA_immigrants)

# Rename the change column for clarity :
colnames(Investigation_data_arab)[colnames(Investigation_data_arab)
                                  == "arab_immigrants_21.1"] <- "Change_arab_immigrants"
colnames(Investigation_data_MNA)[colnames(Investigation_data_MNA)
                                 == "MNA_immigrants_21.1"] <- "Change_MNA_immigrants"

# Calculate the share of total change attributed to each country - Arabs
total_absolute_change_arab <- sum(abs(Investigation_data_arab$Change_arab_immigrants))
percentage_arab <- (abs(Investigation_data_arab$Change_arab_immigrants) /
                      total_absolute_change_arab) * 100
Investigation_data_arab$Percentage <- percentage_arab
Investigation_data_arab$Percentage <- round(percentage_arab, 1) # round percentage

# Calculate the share of total change attributed to each country - MNA
total_absolute_change_MNA <- sum(abs(Investigation_data_MNA$Change_MNA_immigrants))
percentage_MNA <- (abs(Investigation_data_MNA$Change_MNA_immigrants) /
                     total_absolute_change_MNA) * 100
Investigation_data_MNA$Percentage <- percentage_MNA
Investigation_data_MNA$Percentage <- round(percentage_MNA, 1) # round percentage


# Compute the total of all observations in each variable/column:
Total_arab <- colSums(Investigation_data_arab, na.rm = TRUE)
Investigation_data_arab <- rbind(Investigation_data_arab, Total_arab = Total_arab)
Total_MNA <- colSums(Investigation_data_MNA, na.rm = TRUE)
Investigation_data_MNA <- rbind(Investigation_data_MNA, Total_MNA = Total_MNA)


# Identify rows with negative values in the 'Change_arab_immigrants' column
Investigation_data_arab[Investigation_data_arab$Change_arab_immigrants < 0, ]
```

```
##            arab_immigrants_10 arab_immigrants_21 Change_arab_immigrants
## Mauritania                 79                 68                    -11
##            Percentage
## Mauritania          0
```

```r
Investigation_data_MNA[Investigation_data_MNA$Change_MNA_immigrants < 0, ]
```

```
##            MNA_immigrants_10 MNA_immigrants_21 Change_MNA_immigrants Percentage
## Chad                     118               117                    -1        0.0
```

```
## Mozambique                  133             100                     -33         0.1
## Niger                       117              94                     -23         0.1
## Suriname                     10               7                      -3         0.0
## Türkiye                   71835           68764                   -3071        10.5
```

```r
# Total of all negative changes = -3142 immigrants
(3071/3142)*100 # Turkey represents 97.74% of them by itself
```

```
## [1] 97.74029
```

Part 13: Maps to Visualise Data Across Switzerland

Note: this section is located at the bottom of the Markdown because it requires reinitialising the final_merged data set to avoid NAs in the 6 municipalities which were previously removed. This ensures that all municipalities can be mapped.

13.1: Set Up for the Maps

Reloading the Data including all Municipalities and creating the necessary objects

```r
# List of data sets to merge
datasets_list <- list(merged_data_final24, Referendum_data, income_merged,
                      social_help_merged, Pop_density_merged, SVP_merged)

# Use Reduce to merge all datasets on 'bfs_nr_new'
final_merged <- Reduce(function(x, y) merge(x, y, by = "bfs_nr_new",
                                            all = TRUE), datasets_list)






# Define thresholds for "No Change" as less than 0.05pp. variation in the shares
no_change_lower_bound <- -0.05
no_change_upper_bound <- 0.05

# Calculate medians for the changes outside the "No Change" category
negative_changes <- final_merged$Change_Muslims[final_merged$Change_Muslims <
                                                  no_change_lower_bound]
positive_changes <- final_merged$Change_Muslims[final_merged$Change_Muslims >
                                                  no_change_upper_bound]

median_negative_change <- median(negative_changes, na.rm = TRUE)
median_positive_change <- median(positive_changes, na.rm = TRUE)


# Recode the changes into a categorical variable
final_merged <- final_merged %>%
  mutate(Change_Muslim_Cat = case_when(
    Change_Muslims < median_negative_change ~ "High Decrease",
```

```r
    Change_Muslims >= median_negative_change & Change_Muslims <
      no_change_lower_bound ~ "Low Decrease",
    Change_Muslims >= no_change_lower_bound & Change_Muslims <=
      no_change_upper_bound ~ "No Change",
    Change_Muslims > no_change_upper_bound & Change_Muslims <=
      median_positive_change ~ "Low Increase",
    Change_Muslims > median_positive_change ~ "High Increase"
  ))

# Convert to factor, set "No Change" as reference category, and reorder other levels
final_merged$Change_Muslim_Cat <- factor(final_merged$Change_Muslim_Cat,
                                         levels = c("No Change", "High Increase",
                                                    "Low Increase", "Low Decrease",
                                                    "High Decrease"))




# 1) Set up for the Arab Model

# Find the level thresholds
negative_changes_arabs <- final_merged$Change_arabs[final_merged$Change_arabs <
                                                      no_change_lower_bound]
positive_changes_arabs <- final_merged$Change_arabs[final_merged$Change_arabs >
                                                      no_change_upper_bound]

median_negative_change <- median(negative_changes_arabs, na.rm = TRUE)
median_positive_change <- median(positive_changes_arabs, na.rm = TRUE)

final_merged <- final_merged %>%
  mutate(Change_Arabs_Cat2 = case_when(
    Change_arabs < median_negative_change ~ "High Decrease",
    Change_arabs >= median_negative_change & Change_arabs < no_change_lower_bound ~
      "Low Decrease",
    Change_arabs >= no_change_lower_bound & Change_arabs <= no_change_upper_bound ~
      "No Change",
    Change_arabs > no_change_upper_bound & Change_arabs <= median_positive_change ~
      "Low Increase",
    Change_arabs > median_positive_change ~ "High Increase"
  ))

final_merged$Change_Arabs_Cat2 <- factor(final_merged$Change_Arabs_Cat2,
                                         levels = c("No Change", "High Increase",
                                                    "Low Increase", "Low Decrease",
                                                    "High Decrease"))




# 2) Set up for the MNA Model

# Find the level thresholds
negative_changes_MNA <- final_merged$Change_MNA[final_merged$Change_MNA <
                                                  no_change_lower_bound]
```

```r
positive_changes_MNA <- final_merged$Change_MNA[final_merged$Change_MNA >
                                                  no_change_upper_bound]

median_negative_change <- median(negative_changes_MNA, na.rm = TRUE)
median_positive_change <- median(positive_changes_MNA, na.rm = TRUE)

final_merged <- final_merged %>%
  mutate(Change_MNA_Cat2 = case_when(
    Change_MNA < median_negative_change ~ "High Decrease",
    Change_MNA >= median_negative_change & Change_MNA < no_change_lower_bound ~
      "Low Decrease",
    Change_MNA >= no_change_lower_bound & Change_MNA <= no_change_upper_bound ~
      "No Change",
    Change_MNA > no_change_upper_bound & Change_MNA <= median_positive_change ~
      "Low Increase",
    Change_MNA > median_positive_change ~ "High Increase"
  ))

# Convert to factor,  set "No Change" as the reference category, reorder
final_merged$Change_MNA_Cat2 <- factor(final_merged$Change_MNA_Cat2,
                                       levels = c("No Change", "High Increase",
                                                  "Low Increase", "Low Decrease",
                                                  "High Decrease"))




# 3) Set up for the MNA without Turks Model

# Create the necessary variables
final_merged$Change_MNA_NO_TURKS <- final_merged$Change_MNA -
  final_merged$Change_turkish
final_merged$MNA_NO_TURKS_share_10 <- final_merged$MNA_share_10 -
  final_merged$Turkish_share_10

# Find the level thresholds
negative_changes_MNA_NO_TURKS <-
  final_merged$Change_MNA_NO_TURKS[final_merged$Change_MNA_NO_TURKS <
                                     no_change_lower_bound]
positive_changes_MNA_NO_TURKS <-
  final_merged$Change_MNA_NO_TURKS[final_merged$Change_MNA_NO_TURKS >
                                     no_change_upper_bound]

median_negative_change <- median(negative_changes_MNA_NO_TURKS)
median_positive_change <- median(positive_changes_MNA_NO_TURKS)


# Re-code the MNA no Turk changes into a categorical variable with 5 categories
final_merged <- final_merged %>%
  mutate(Change_MNA_NO_TURKS_Cat = case_when(
    Change_MNA_NO_TURKS < median_negative_change ~ "High Decrease",
    Change_MNA_NO_TURKS >= median_negative_change & Change_MNA_NO_TURKS <
      no_change_lower_bound ~ "Low Decrease",
```

```r
    Change_MNA_NO_TURKS >= no_change_lower_bound & Change_MNA_NO_TURKS <=
      no_change_upper_bound ~ "No Change",
    Change_MNA_NO_TURKS > no_change_upper_bound & Change_MNA_NO_TURKS <=
      median_positive_change ~ "Low Increase",
    Change_MNA_NO_TURKS > median_positive_change ~ "High Increase"
  ))

# Convert to factor,  set "No Change" as the reference category, reorder
final_merged$Change_MNA_NO_TURKS_Cat <- factor(final_merged$Change_MNA_NO_TURKS_Cat,
                                       levels = c("No Change", "High Increase",
                                                  "Low Increase", "Low Decrease",
                                                  "High Decrease"))
```

Setting up the GIS Data

```r
test <- read_sf(paste0("/Users/alexandredore/Desktop/Dissertation/ALL data/01_INST/",
                       "GesamtflÑche_gf/K4_polg20240101_gf"))
```

```
## Warning in CPL_read_ogr(dsn, layer, query, as.character(options), quiet, :
## automatically selected the first layer in a data source containing more than
## one.
```

```r
# Merge attributes from merged_sf into test based on matching municipality IDs
full_data <- merge(test, final_merged, by.x = "id", by.y = "bfs_nr_new", all.x = TRUE)
```

13.2:Mapping

13.2.1: Localising Outliers in the Main Specification

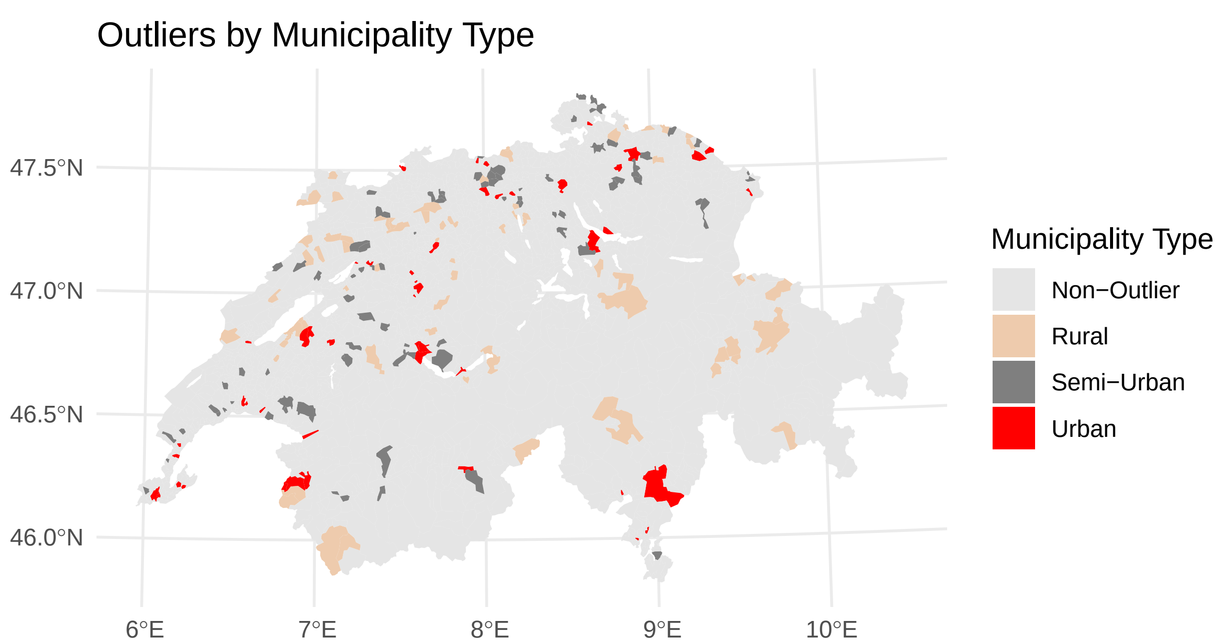
```r
# Subset the relevant columns for outliers
outlier_main_details <- full_data[outliers_Final2, c("Municipality_type", "Linguistic_Region")]

# Calculate frequency of each type and region among outliers
table(outlier_main_details$Municipality_type)
```

```
##
##      Rural Semi-Urban      Urban
##         69         82         47
```

```r
table(outlier_main_details$Linguistic_Region)
```

```
##
##  French  German Italian
##      65     125       8
```

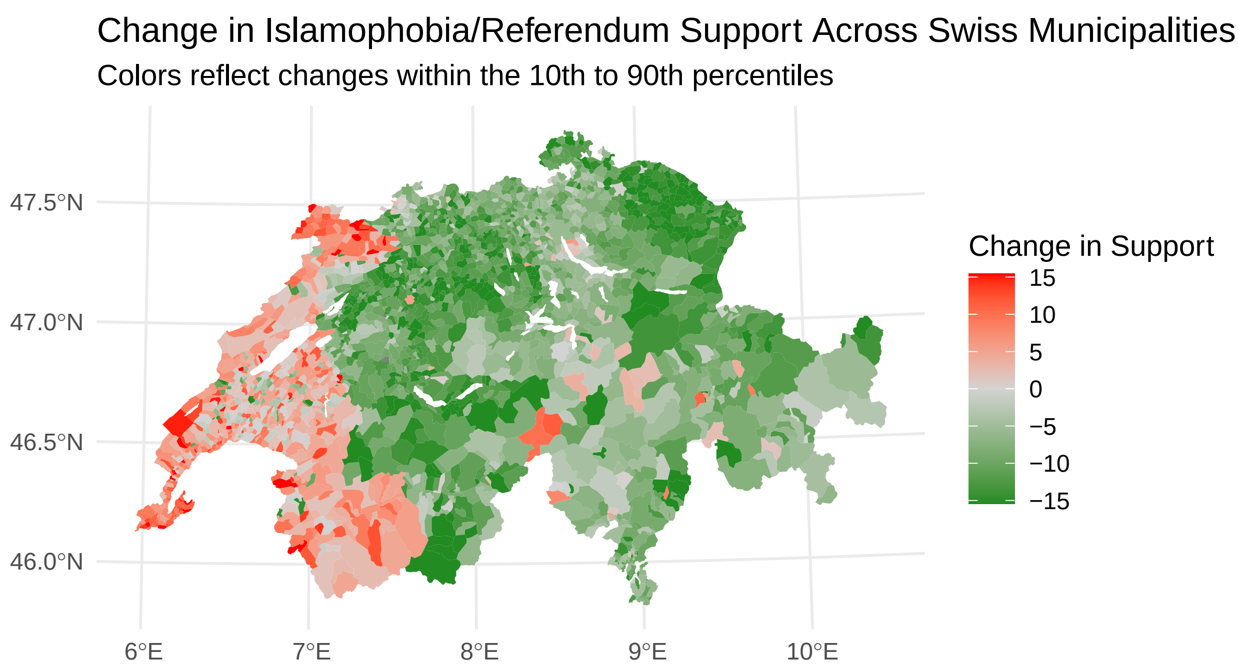
```r
# Add a column for coloring purposes
full_data$plot_color <- ifelse(row.names(full_data) %in% outliers_Final2,
                               as.character(full_data$Municipality_type),
                               "Non-Outlier")

# Plotting the entire dataset with different colors for municipality types among outliers
ggplot(data = full_data) +
  geom_sf(aes(fill = plot_color), size = 0.1, color = NA) +
  scale_fill_manual(values = c("Urban" = "red", "Semi-Urban" = "grey50",
                               "Rural" = "peachpuff2", "Non-Outlier" = "grey90"),
                    name = "Municipality Type") +
  labs(title = "Outliers by Municipality Type") +
  theme_minimal() +
  theme(legend.position = "right")
```



13.2.2: Mapping the Change in Islamophobic Referendum Support

```r
# Calculate the 10th and 90th percentiles for Yes_percent_change
percentile_10 <- quantile(full_data$Yes_percent_change, 0.1, na.rm = TRUE)
percentile_90 <- quantile(full_data$Yes_percent_change, 0.9, na.rm = TRUE)

# Adjust color intensity based on the 10th and 90th percentiles
max_abs_change <- max(abs(c(percentile_10, percentile_90)))
```

```r
# Create the plot
ggplot(data = full_data) +
geom_sf(aes(fill = Yes_percent_change), size = 0.1, color = NA) +
  scale_fill_gradient2(name = "Change in Support",
low = "forestgreen", mid = "lightgrey", high = "red", midpoint = 0, # No change point
limits = c(-max_abs_change, max_abs_change), oob = scales::oob_squish) +
labs(title = "Change in Islamophobia/Referendum Support Across Swiss Municipalities",
subtitle = "Colors reflect changes within the 10th to 90th percentiles") +
theme_minimal() + theme(legend.position = "right")
```



End of the Code