Exercise 2

2024-01-23

Group members:

- Tashfeen Ahmed
- Adrian Alarcon
- Yvan Kammelu
- Zhicheng Zhong

```
#install.packages(c("arrow","gender", "wru", "lubridate", "gtsummary"))
# Load required libraries
library(broom)
library(gender)
library(wru)
library(lubridate)

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(gtsummary)
library(arrow)

##
## Attaching package: 'arrow'

## The following object is masked from 'package:lubridate':
##
##     duration
```

```
## The following object is masked from 'package:utils':
##
##     timestamp

library(tidyr)
library(zoo)

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

library(purrr)

data<- read_feather("app_data_starter.feather")

# Task 1: Create individual-level variables
examiner_names <- data %>% distinct(examiner_name_first)

examiner_names

## # A tibble: 2,595 × 1
##    examiner_name_first
##    <chr>
##  1 JACQUELINE
##  2 BEKIR
##  3 CYNTHIA
##  4 MARY
##  5 MICHAEL
##  6 LINDA
##  7 KARA
##  8 VANESSA
##  9 TERESA
## 10 SUN
## # ℹ 2,585 more rows
```

Obtaining gender of the examiner

Using the gender package, we identify the gender of the examiner based on the first name, according to the documentation.

```
# get a table of names and gender



examiner_names_gender <- examiner_names %>%
  do(results = gender(.$examiner_name_first, method = "ssa")) %>%
  unnest(cols = c(results), keep_empty = TRUE) %>%
  select(
    examiner_name_first = name,
```

```
    gender,
    proportion_female
  )


head(examiner_names_gender,10)
```

```
## # A tibble: 10 × 3
##    examiner_name_first gender proportion_female
##    <chr>               <chr>              <dbl>
##  1 AARON               male              0.0082
##  2 ABDEL               male              0
##  3 ABDOU               male              0
##  4 ABDUL               male              0
##  5 ABDULHAKIM          male              0
##  6 ABDULLAH            male              0
##  7 ABDULLAHI           male              0
##  8 ABIGAIL             female            0.998
##  9 ABIMBOLA            female            0.944
## 10 ABRAHAM             male              0.0031
```

In this part, we joined the gender data obtained in the previous step into the main dataset.

```
# remove extra colums from the gender table
examiner_names_gender <- examiner_names_gender %>%
  select(examiner_name_first, gender)

# joining gender back to the dataset
data <- data %>%
  left_join(examiner_names_gender, by = "examiner_name_first")

# cleaning up
rm(examiner_names)
rm(examiner_names_gender)
gc()
```

```
##            used  (Mb) gc trigger  (Mb) limit (Mb)  max used  (Mb)
## Ncells  4496757 240.2    8038890 429.4         NA   4517262 241.3
## Vcells 59559191 454.5  114748791 875.5      16384 104004775 793.5
```

Obtaining the race of the examiner

Based on the last name, and using the `wru` package, we identified the probability of the examiner to be of an specific race among Asian, Black, Hispanic and other.

```
library(wru)

examiner_surnames <- data %>%
  select(surname = examiner_name_last) %>%
  distinct()
```

```
examiner_surnames
```

```
## # A tibble: 3,806 × 1
##    surname
##    <chr>
##  1 HOWARD
##  2 YILDIRIM
##  3 HAMILTON
##  4 MOSHER
##  5 BARR
##  6 GRAY
##  7 MCMILLIAN
##  8 FORD
##  9 STRZELECKA
## 10 KIM
## # ℹ 3,796 more rows
```

```
examiner_race <- predict_race(voter.file = examiner_surnames, surname.only =
T) %>%
  as_tibble()
```

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

```
## Proceeding with last name predictions...
```

```
## ℹ All local files already up-to-date!
```

```
## 701 (18.4%) individuals' last names were not matched.
```

```
examiner_race
```

```
## # A tibble: 3,806 × 6
##    surname    pred.whi pred.bla pred.his pred.asi pred.oth
##    <chr>         <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
##  1 HOWARD       0.597   0.295    0.0275   0.00690   0.0741
##  2 YILDIRIM     0.807   0.0273   0.0694   0.0165    0.0798
##  3 HAMILTON     0.656   0.239    0.0286   0.00750   0.0692
##  4 MOSHER       0.915   0.00425  0.0291   0.00917   0.0427
##  5 BARR         0.784   0.120    0.0268   0.00830   0.0615
##  6 GRAY         0.640   0.252    0.0281   0.00748   0.0724
##  7 MCMILLIAN    0.322   0.554    0.0212   0.00340   0.0995
##  8 FORD         0.576   0.320    0.0275   0.00621   0.0697
##  9 STRZELECKA   0.472   0.171    0.220    0.0825    0.0543
## 10 KIM          0.0169  0.00282  0.00546  0.943     0.0319
## # ℹ 3,796 more rows
```

```
examiner_race <- examiner_race %>%
  mutate(max_race_p = pmax(pred.asi, pred.bla, pred.his, pred.oth, pred.whi))
%>%
  mutate(race = case_when(
    max_race_p == pred.asi ~ "Asian",
```

```
    max_race_p == pred.bla ~ "black",
    max_race_p == pred.his ~ "Hispanic",
    max_race_p == pred.oth ~ "other",
    max_race_p == pred.whi ~ "white",
    TRUE ~ NA_character_
  ))

examiner_race
```

```
## # A tibble: 3,806 × 8
##    surname    pred.whi pred.bla pred.his pred.asi pred.oth max_race_p race
##    <chr>         <dbl>    <dbl>    <dbl>    <dbl>    <dbl>      <dbl>
<chr>
##  1 HOWARD       0.597   0.295    0.0275   0.00690   0.0741      0.597
white
##  2 YILDIRIM     0.807   0.0273   0.0694   0.0165    0.0798      0.807
white
##  3 HAMILTON     0.656   0.239    0.0286   0.00750   0.0692      0.656
white
##  4 MOSHER       0.915   0.00425  0.0291   0.00917   0.0427      0.915
white
##  5 BARR         0.784   0.120    0.0268   0.00830   0.0615      0.784
white
##  6 GRAY         0.640   0.252    0.0281   0.00748   0.0724      0.640
white
##  7 MCMILLIAN    0.322   0.554    0.0212   0.00340   0.0995      0.554
black
##  8 FORD         0.576   0.320    0.0275   0.00621   0.0697      0.576
white
##  9 STRZELECKA   0.472   0.171    0.220    0.0825    0.0543      0.472
white
## 10 KIM          0.0169  0.00282  0.00546  0.943     0.0319      0.943
Asian
## # ℹ 3,796 more rows
```

On this step, we cleaned the dataset removing extra columns

```
# removing extra columns
examiner_race <- examiner_race %>%
  select(surname,race)

data <- data %>%
  left_join(examiner_race, by = c("examiner_name_last" = "surname"))

rm(examiner_race)
rm(examiner_surnames)
gc()
```

```
##            used  (Mb) gc trigger  (Mb) limit (Mb)  max used  (Mb)
## Ncells  4605650 246.0    8038890 429.4         NA   6588121 351.9
## Vcells 61778071 471.4  114748791 875.5      16384 113446620 865.6

library(lubridate) # to work with dates

examiner_dates <- data %>%
  select(examiner_id, filing_date, appl_status_date)

examiner_dates

## # A tibble: 2,018,477 × 3
##    examiner_id filing_date appl_status_date
##          <dbl> <date>      <chr>
##  1       96082 2000-01-26  30jan2003 00:00:00
##  2       87678 2000-10-11  27sep2010 00:00:00
##  3       63213 2000-05-17  30mar2009 00:00:00
##  4       73788 2001-07-20  07sep2009 00:00:00
##  5       77294 2000-04-10  19apr2001 00:00:00
##  6       68606 2000-04-28  16jul2001 00:00:00
##  7       89557 2004-01-26  15may2017 00:00:00
##  8       97543 2000-06-23  03apr2002 00:00:00
##  9       98714 2000-02-04  27nov2002 00:00:00
## 10       65530 2002-02-20  23mar2009 00:00:00
## # ℹ 2,018,467 more rows

examiner_dates <- examiner_dates %>%
  mutate(start_date = ymd(filing_date), end_date =
as_date(dmy_hms(appl_status_date)))
```

After the cleaning and preprocessing steps, we grouped the data at a examiner level. This would allow us to perform a regression models

```
examiner_dates <- examiner_dates %>%
  group_by(examiner_id) %>%
  summarise(
    earliest_date = min(start_date, na.rm = TRUE),
    latest_date = max(end_date, na.rm = TRUE),
    tenure_days = interval(earliest_date, latest_date) %/% days(1)
    ) %>%
  filter(year(latest_date)<2018)

examiner_dates

## # A tibble: 5,625 × 4
##    examiner_id earliest_date latest_date tenure_days
##          <dbl> <date>        <date>            <dbl>
##  1       59012 2004-07-28    2015-07-24         4013
##  2       59025 2009-10-26    2017-05-18         2761
##  3       59030 2005-12-12    2017-05-22         4179
##  4       59040 2007-09-11    2017-05-23         3542
```

```
##  5       59052 2001-08-21    2007-02-28         2017
##  6       59054 2000-11-10    2016-12-23         5887
##  7       59055 2004-11-02    2007-12-26         1149
##  8       59056 2000-03-24    2017-05-22         6268
##  9       59074 2000-01-31    2017-03-17         6255
## 10       59081 2011-04-21    2017-05-19         2220
## # ℹ 5,615 more rows
```

```
data <- data %>%
  left_join(examiner_dates, by = "examiner_id")

rm(examiner_dates)
gc()
```

```
##            used  (Mb) gc trigger   (Mb) limit (Mb)  max used  (Mb)
## Ncells  4614161 246.5    8038890  429.4         NA   8038890 429.4
## Vcells 67853113 517.7  137778549 1051.2      16384 116340619 887.7
```

```
data
```

```
## # A tibble: 2,018,477 × 26
##    application_number filing_date examiner_name_last examiner_name_first
##    <chr>              <date>      <chr>              <chr>
##  1 08284457           2000-01-26  HOWARD             JACQUELINE
##  2 08413193           2000-10-11  YILDIRIM           BEKIR
##  3 08531853           2000-05-17  HAMILTON           CYNTHIA
##  4 08637752           2001-07-20  MOSHER             MARY
##  5 08682726           2000-04-10  BARR               MICHAEL
##  6 08687412           2000-04-28  GRAY               LINDA
##  7 08716371           2004-01-26  MCMILLIAN          KARA
##  8 08765941           2000-06-23  FORD               VANESSA
##  9 08776818           2000-02-04  STRZELECKA         TERESA
## 10 08809677           2002-02-20  KIM                SUN
## # ℹ 2,018,467 more rows
## # ℹ 22 more variables: examiner_name_middle <chr>, examiner_id <dbl>,
## #   examiner_art_unit <dbl>, uspc_class <chr>, uspc_subclass <chr>,
## #   patent_number <chr>, patent_issue_date <date>, abandon_date <date>,
## #   disposal_type <chr>, appl_status_code <dbl>, appl_status_date <chr>,
## #   tc <dbl>, gender.x <chr>, race.x <chr>, earliest_date.x <date>,
## #   latest_date.x <date>, tenure_days.x <dbl>, gender.y <chr>, race.y
<chr>, …
```

```
data <- data %>%
  select(
    application_number,
    filing_date,
    examiner_name_last,
    examiner_name_first,
    examiner_name_middle,
    examiner_id,
    examiner_art_unit,
```

```
    uspc_class,
    uspc_subclass,
    patent_number,
    patent_issue_date,
    abandon_date,
    disposal_type,
    appl_status_code,
    appl_status_date,
    tc,
    gender = gender.y,  # Renaming the column to remove the suffix
    race = race.y,      # Renaming the column to remove the suffix
    earliest_date = earliest_date.y,  # Renaming the column to remove the
suffix
    latest_date = latest_date.y,  # Renaming the column to remove the suffix
    tenure_days = tenure_days.y   # Renaming the column to remove the suffix
  )
data
```

```
## # A tibble: 2,018,477 × 21
##    application_number filing_date examiner_name_last examiner_name_first
##    <chr>              <date>      <chr>              <chr>
##  1 08284457           2000-01-26  HOWARD             JACQUELINE
##  2 08413193           2000-10-11  YILDIRIM           BEKIR
##  3 08531853           2000-05-17  HAMILTON           CYNTHIA
##  4 08637752           2001-07-20  MOSHER             MARY
##  5 08682726           2000-04-10  BARR               MICHAEL
##  6 08687412           2000-04-28  GRAY               LINDA
##  7 08716371           2004-01-26  MCMILLIAN          KARA
##  8 08765941           2000-06-23  FORD               VANESSA
##  9 08776818           2000-02-04  STRZELECKA         TERESA
## 10 08809677           2002-02-20  KIM                SUN
## # ℹ 2,018,467 more rows
## # ℹ 17 more variables: examiner_name_middle <chr>, examiner_id <dbl>,
## #   examiner_art_unit <dbl>, uspc_class <chr>, uspc_subclass <chr>,
## #   patent_number <chr>, patent_issue_date <date>, abandon_date <date>,
## #   disposal_type <chr>, appl_status_code <dbl>, appl_status_date <chr>,
## #   tc <dbl>, gender <chr>, race <chr>, earliest_date <date>,
## #   latest_date <date>, tenure_days <dbl>
```

Task 2: Create a panel dataset

———————————————

```
library(dplyr)
library(lubridate)
library(zoo)

# Convert dates to quarters
```

```r
data <- data %>%
  mutate(
    filing_year_quarter = as.yearqtr(filing_date),
    abandon_year_quarter = as.yearqtr(abandon_date),
    issue_year_quarter = as.yearqtr(patent_issue_date)
  )

# Aggregate applications data by quarter
panel_data <- data %>%
  group_by(examiner_id, filing_year_quarter) %>%
  summarise(
    num_new_applications = n_distinct(application_number),
    num_abandoned_applications = sum(disposal_type == "ABN", na.rm = TRUE),
    num_issued_patents = sum(disposal_type == "ISS", na.rm = TRUE),
    num_in_process_applications = sum(disposal_type == "PEND", na.rm = TRUE),
    current_art_unit = first(examiner_art_unit),
    .groups = 'drop'
  )

# Add the count of people and women in each art unit per quarter
art_unit_info <- data %>%
  group_by(filing_year_quarter, examiner_art_unit) %>%
  summarise(
    num_people_in_art_unit = n_distinct(examiner_id),
    num_women_in_art_unit = sum(gender == "female", na.rm = TRUE),
    .groups = 'drop'
  )

# Join the art unit info with the main panel data
panel_data <- panel_data %>%
  left_join(art_unit_info, by = c("filing_year_quarter", "current_art_unit" =
"examiner_art_unit"))

# Mark the last five quarters for each examiner
panel_data <- panel_data %>%
  group_by(examiner_id) %>%
  mutate(
    # Get a list of the last five quarters of activity for each examiner
    last_five_quarters = list(tail(sort(unique(filing_year_quarter)), 5))
  ) %>%
  ungroup() %>%
  mutate(
    # Check if the current quarter is in the last five quarters of activity
    separation_indicator = if_else(map_lgl(filing_year_quarter, ~ .x %in%
last_five_quarters[[1]]), 1, 0)
  )


# Detect changes in current_art_unit
```

```r
panel_data <- panel_data %>%
  group_by(examiner_id) %>%
  mutate(
    # If the current art unit is different from the previous one, it's a move
(1), otherwise, it's not (0).
    # For the first row of each examiner (where there is no "previous" art
unit), use NA as the default value.
    AU_move_indicator = if_else(current_art_unit != lag(current_art_unit,
default = NA), 1, 0)
  ) %>%
  mutate(
    # Replace NA with 0 - assumes that the first observation is not a move.
    AU_move_indicator = replace_na(AU_move_indicator, 0)
  ) %>%
  ungroup()

table(panel_data$separation_indicator)

##
##      0      1
## 175481  15400

table(panel_data$AU_move_indicator)

##
##      0      1
## 168875  22006
```

Task 3: Estimate predictors for turnover and mobility

———————————————————

```r
# Prepare the data for regression
regression_data <- panel_data %>%
  filter(num_new_applications > 0)

# Regression model for Turnover
turnover_model <- glm(separation_indicator ~  num_new_applications +
num_abandoned_applications +
                        num_issued_patents +
                        num_people_in_art_unit + num_women_in_art_unit,
                      family = binomial(), data = regression_data)

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

# Regression model for Mobility (AU Move)
mobility_model <- glm(AU_move_indicator ~ num_new_applications +
num_abandoned_applications +
                        num_issued_patents + num_in_process_applications +
```

```
                      num_people_in_art_unit + num_women_in_art_unit,
                    family = binomial(), data = regression_data)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
# Create descriptive tables for both models
turnover_table_ <- tbl_regression(turnover_model)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
turnover_table_
```

```
## Table printed with `knitr::kable()`, not {gt}. Learn why at
## https://www.danieldsjoberg.com/gtsummary/articles/rmarkdown.html
```

```
## To suppress this message, include `message = FALSE` in the code chunk
header.
```

Characteristic	log(OR)	95% CI	p-value
num_new_applications	-0.58	-0.61, -0.56	<0.001
num_abandoned_applications	0.63	0.60, 0.65	<0.001
num_issued_patents	0.57	0.55, 0.60	<0.001
num_people_in_art_unit	0.01	0.01, 0.01	<0.001
num_women_in_art_unit	-0.01	-0.01, -0.01	<0.001

Showing the models

```
turnover_model

##
## Call:  glm(formula = separation_indicator ~ num_new_applications +
num_abandoned_applications +
##     num_issued_patents + num_people_in_art_unit + num_women_in_art_unit,
##     family = binomial(), data = regression_data)
##
## Coefficients:
##                 (Intercept)        num_new_applications
##                   -2.113591                   -0.581599
## num_abandoned_applications          num_issued_patents
##                    0.625752                    0.573120
##     num_people_in_art_unit       num_women_in_art_unit
##                    0.007489                   -0.005844
##
## Degrees of Freedom: 190880 Total (i.e. Null);  190875 Residual
## Null Deviance:       107100
## Residual Deviance: 100400    AIC: 100400

tidy_results <- tidy(mobility_model)
tidy_results

## # A tibble: 7 × 5
##   term                        estimate std.error statistic    p.value
##   <chr>                          <dbl>     <dbl>     <dbl>      <dbl>
## 1 (Intercept)                  -2.50    0.0139      -180.   0
## 2 num_new_applications         -0.142   0.00460      -30.9  2.47e-209
## 3 num_abandoned_applications    0.221   0.00481       45.9  0
## 4 num_issued_patents            0.148   0.00461       32.1  5.93e-226
## 5 num_in_process_applications  NA      NA             NA   NA
## 6 num_people_in_art_unit        0.0410  0.000589      69.7  0
## 7 num_women_in_art_unit        -0.0123  0.000224     -55.0  0

mobility_model
```

```
##
## Call:  glm(formula = AU_move_indicator ~ num_new_applications +
num_abandoned_applications +
##     num_issued_patents + num_in_process_applications +
num_people_in_art_unit +
##     num_women_in_art_unit, family = binomial(), data = regression_data)
##
## Coefficients:
##                 (Intercept)         num_new_applications
##                    -2.49914                     -0.14193
##  num_abandoned_applications           num_issued_patents
##                     0.22083                      0.14784
## num_in_process_applications       num_people_in_art_unit
##                          NA                      0.04101
##       num_women_in_art_unit
##                    -0.01232
##
## Degrees of Freedom: 190880 Total (i.e. Null);  190875 Residual
## Null Deviance:       136500
## Residual Deviance: 125400    AIC: 125400
```