R Notebook

```
library(tidyverse)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.1
```

```
## Warning: package 'lubridate' was built under R version 4.3.1
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.5.0     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.1
## -- 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(arrow)
```

```
## Warning: package 'arrow' was built under R version 4.3.1
```

```
##
## Attaching package: 'arrow'
##
## The following object is masked from 'package:lubridate':
##
##     duration
##
## The following object is masked from 'package:utils':
##
##     timestamp
```

```
library(tidyverse)
library(lubridate)
library(gender)
library(igraph)
```

```
## Warning: package 'igraph' was built under R version 4.3.1
```

```
##
## Attaching package: 'igraph'
##
## The following objects are masked from 'package:lubridate':
```

```
##
##     %--%, union
##
## The following objects are masked from 'package:dplyr':
##
##     as_data_frame, groups, union
##
## The following objects are masked from 'package:purrr':
##
##     compose, simplify
##
## The following object is masked from 'package:tidyr':
##
##     crossing
##
## The following object is masked from 'package:tibble':
##
##     as_data_frame
##
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
##
## The following object is masked from 'package:base':
##
##     union
```

```
library(dplyr)
```

```
applications <- read_parquet("/Users/kaz/DataspellProjects/Org-Analytics/E3/app_data_sample.parquet")
edges <- read_csv("/Users/kaz/DataspellProjects/Org-Analytics/E3/edges_sample.csv")
```

```
## Rows: 32906 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): application_number
## dbl  (2): ego_examiner_id, alter_examiner_id
## date (1): advice_date
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

get the gender var

```
library(gender)
examiner_names <- applications %>%
  distinct(examiner_name_first)

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)

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

# joining gender back to the dataset
applications <- applications %>%
  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  4530441  242    8044888 429.7         NA  4549933 243.0
## Vcells 49663651  379   93185342 711.0      16384 79979476 610.2
```

Get the race var

```
library(wru)
```

```
## Warning: package 'wru' was built under R version 4.3.1
```

```
##
## Please cite as:
##
## Khanna K, Bertelsen B, Olivella S, Rosenman E, Rossell Hayes A, Imai K
## (2024). _wru: Who are You? Bayesian Prediction of Racial Category Using
## Surname, First Name, Middle Name, and Geolocation_. R package version
## 3.0.1, <https://CRAN.R-project.org/package=wru>.
##
## Note that wru 2.0.0 uses 2020 census data by default.
## Use the argument `year = "2010"`, to replicate analyses produced with earlier package versions.
```

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

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

```
## Predicting race for 2020
```

```
## 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 <- 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_
  ))



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

applications <- applications %>%
  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  4738658 253.1    8044888 429.7         NA  6962786 371.9
## Vcells 52052832 397.2   93185342 711.0      16384 92294058 704.2
```

Get Tenure

```
library(lubridate) # to work with dates

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

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

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)

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

rm(examiner_dates)
gc()
```

```
##            used  (Mb) gc trigger  (Mb) limit (Mb)  max used  (Mb)
## Ncells  4747606 253.6    8044888 429.7         NA   8044888 429.7
## Vcells 58129065 443.5  111902410 853.8      16384 111622153 851.7
```

Create Var for application processing time

```
# diff between filing date and patent_issue_date or abandon_date
applications <- applications %>%
  mutate(
    patent_issue_date = ymd(patent_issue_date),
    abandon_date = ymd(abandon_date),
    app_proc_time = case_when(
      !is.na(patent_issue_date) ~ interval(filing_date, patent_issue_date) %/% days(1),
      !is.na(abandon_date) ~ interval(filing_date, abandon_date) %/% days(1),
      TRUE ~ NA_real_
    )
  )
```

Check the summary stat of the new var

```
summary(applications$app_proc_time)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##  -13636     765    1079    1190    1481   17898  329761
```

Let's delete the erroneous values. For NA values, we will remove them as well for NOW.

```
applications <- applications %>%
  filter(app_proc_time > 0)

# remove na
applications <- applications %>%
  filter(!is.na(app_proc_time))
```

Graph Network

```
advice_network <- graph_from_data_frame(d = edges[, c("ego_examiner_id", "alter_examiner_id")], directe
```

```
## Warning in graph_from_data_frame(d = edges[, c("ego_examiner_id",
## "alter_examiner_id")], : In 'd' 'NA' elements were replaced with string "NA"
```

```
degree_centrality <- degree(advice_network, mode = "all")

# Calculate betweenness centrality for each node (examiner)
betweenness_centrality <- betweenness(advice_network, directed = TRUE)

# Create a dataframe of centrality scores
centrality_scores <- data.frame(
  examiner_id = V(advice_network)$name,
  degree = degree_centrality,
  betweenness = betweenness_centrality
)
```

```
applications$examiner_id <- as.character(applications$examiner_id)
centrality_scores$examiner_id <- as.character(centrality_scores$examiner_id)


# erge the centrality scores with the applications data
applications <- applications %>%
  left_join(centrality_scores, by = "examiner_id")
```

Linear Regression

- Need to chekc the datatypes of cols before running the regression

```
str(applications)
```

```
## tibble [1,688,673 x 24] (S3: tbl_df/tbl/data.frame)
##  $ application_number  : chr [1:1688673] "08284457" "08413193" "08637752" "08682726" ...
##  $ filing_date         : Date[1:1688673], format: "2000-01-26" "2000-10-11" ...
##  $ examiner_name_last  : chr [1:1688673] "HOWARD" "YILDIRIM" "MOSHER" "BARR" ...
##  $ examiner_name_first : chr [1:1688673] "JACQUELINE" "BEKIR" "MARY" "MICHAEL" ...
##  $ examiner_name_middle: chr [1:1688673] "V" "L" NA "E" ...
##  $ examiner_id         : chr [1:1688673] "96082" "87678" "73788" "77294" ...
##  $ examiner_art_unit   : num [1:1688673] 1764 1764 1648 1762 1734 ...
##  $ uspc_class          : chr [1:1688673] "508" "208" "530" "427" ...
##  $ uspc_subclass       : chr [1:1688673] "273000" "179000" "388300" "430100" ...
##  $ patent_number       : chr [1:1688673] "6521570" "6440298" "6927281" NA ...
##  $ patent_issue_date   : Date[1:1688673], format: "2003-02-18" "2002-08-27" ...
##  $ abandon_date        : Date[1:1688673], format: NA NA ...
##  $ disposal_type       : chr [1:1688673] "ISS" "ISS" "ISS" "ABN" ...
##  $ appl_status_code    : num [1:1688673] 150 250 250 161 150 161 161 250 250 250 ...
##  $ appl_status_date    : chr [1:1688673] "30jan2003 00:00:00" "27sep2010 00:00:00" "07sep2009 00:00:0
##  $ tc                  : num [1:1688673] 1700 1700 1600 1700 1700 1600 1600 1700 1700 1600 ...
##  $ gender              : chr [1:1688673] "female" NA "female" "male" ...
##  $ race                : chr [1:1688673] "white" "white" "white" "white" ...
##  $ earliest_date       : Date[1:1688673], format: "2000-01-10" "2000-01-04" ...
##  $ latest_date         : Date[1:1688673], format: "2016-04-01" "2016-09-09" ...
##  $ tenure_days         : num [1:1688673] 5926 6093 6331 6332 6345 ...
##  $ app_proc_time       : num [1:1688673] 1119 685 1481 261 459 ...
##  $ degree              : num [1:1688673] NA NA 3 42 NA 13 NA NA 26 1 ...
##  $ betweenness         : num [1:1688673] NA NA 0 0 NA ...
```

creating more year variables to control for time and possibly age and other time variant factors. Start year may act as a proxy for age

also create a workgroup variable

```
# create new variables - start year and filling year
applications <- applications %>%
  mutate(
    start_year = year(earliest_date),
    filing_year = year(filing_date)
  )


# create a workgroup variable (first 3 digits of art unit)

applications <- applications %>%
  mutate(
    workgroup = substr(examiner_art_unit, 1, 3)
  )
```

Convert the start_year to more generic values

```
summary(applications$start_year)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    2000    2000    2000    2002    2003    2015   18239
```

```
# Convert start year to more generic values - subtract 2000, which is the min value
applications$start_year <- applications$start_year - 2000
```

```
summary(applications$start_year)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   0.000   0.000   0.000   1.604   3.000  15.000   18239
```

```
# count the number of unique examiner art unit
length(unique(applications$examiner_art_unit))
```

```
## [1] 291
```

```
# count the number of unique uspc class
length(unique(applications$uspc_class))
```

```
## [1] 414
```

```
# count the number of unique degree
length(unique(applications$degree))
```

```
## [1] 157
```

```
# count the workgroup
length(unique(applications$workgroup))
```

```
## [1] 38
```

Changing the data types of the relevant columns

I will use workgroup instead of examiner_art_unit

```
# Convert relevant columns to factors
applications$gender <- as.factor(applications$gender)
applications$race <- as.factor(applications$race)
# applications$examiner_art_unit <- as.factor(applications$examiner_art_unit)
applications$workgroup <- as.factor(applications$workgroup)
# applications$start_year <- as.factor(applications$start_year) # decided to treat it as numeric
applications$filling_year <- as.factor(applications$filing_year)
```

```
# Model with interaction term, controlling for other variables
model <- lm(app_proc_time ~ betweenness * gender + degree + tenure_days   + race + start_year + filing_
            data = applications)
```

```
library(stargazer)
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```
# Using stargazer to generate an HTML table of the model summary
stargazer(model, type = "text", title = "Regression Results")
```

```
##
## Regression Results
## ================================================
##                            Dependent variable:
##                        ---------------------------
##                               app_proc_time
## ------------------------------------------------
## betweenness                     0.001***
##                                 (0.0002)
##
## gendermale                     -11.848***
```

```
##                                (1.360)
##
## degree                         -0.034*
##                                (0.018)
##
## tenure_days                   -0.085***
##                                (0.004)
##
## raceHispanic                    2.528
##                                (4.157)
##
## raceblack                     -33.525***
##                                (3.332)
##
## raceother                      -14.358
##                                (17.643)
##
## racewhite                     -5.785***
##                                (1.400)
##
## start_year                    13.175***
##                                (1.595)
##
## filing_year                   -56.820***
##                                (0.152)
##
## workgroup161                    75.810
##                                (54.196)
##
## workgroup162                   -85.263
##                                (54.177)
##
## workgroup163                  131.270**
##                                (54.188)
##
## workgroup164                  126.429**
##                                (54.167)
##
## workgroup165                    41.930
##                                (54.198)
##
## workgroup166                  211.566***
##                                (55.377)
##
## workgroup167                  172.617***
##                                (54.444)
##
## workgroup170                   -206.814
##                               (154.939)
##
## workgroup171                    12.864
##                                (54.180)
##
## workgroup172                   -80.974
```

```
##                                (54.183)
##
## workgroup173                    -13.354
##                                (54.187)
##
## workgroup174                    -53.436
##                                (54.182)
##
## workgroup175                  -290.347***
##                                (54.209)
##
## workgroup176                    -64.308
##                                (54.169)
##
## workgroup177                     6.173
##                                (54.179)
##
## workgroup178                   334.736***
##                                (54.238)
##
## workgroup179                   108.513**
##                                (54.140)
##
## workgroup210                   351.120***
##                                (103.143)
##
## workgroup211                    97.435*
##                                (54.188)
##
## workgroup212                   144.708***
##                                (54.231)
##
## workgroup213                   155.745***
##                                (54.351)
##
## workgroup214                   416.988***
##                                (54.507)
##
## workgroup215                   266.867***
##                                (54.261)
##
## workgroup216                   284.254***
##                                (54.211)
##
## workgroup217                   398.875***
##                                (54.282)
##
## workgroup218                     81.912
##                                (54.199)
##
## workgroup219                   413.667***
##                                (54.255)
##
## workgroup240                     54.214
```

```
##                                  (90.056)
##
## workgroup241                    292.140***
##                                  (54.459)
##
## workgroup242                    473.313***
##                                  (54.303)
##
## workgroup243                    433.967***
##                                  (54.282)
##
## workgroup244                    498.066***
##                                  (54.258)
##
## workgroup245                    465.029***
##                                  (54.240)
##
## workgroup246                    299.008***
##                                  (54.255)
##
## workgroup247                    250.452***
##                                  (54.263)
##
## workgroup248                    339.235***
##                                  (54.455)
##
## workgroup249                    434.434***
##                                  (54.551)
##
## betweenness:gendermale           0.001***
##                                  (0.0002)
##
## Constant                      115,636.400***
##                                  (308.819)
##
## ------------------------------------------------
## Observations                      916,857
## R2                                 0.196
## Adjusted R2                        0.196
## Residual Std. Error        562.242 (df = 916808)
## F Statistic           4,659.125*** (df = 48; 916808)
## ================================================
## Note:                    *p<0.1; **p<0.05; ***p<0.01
```