exercise_4_tashfeen_ahmed

2024-04-02

```r
# Install and load the arrow package
#install.packages("arrow")
#install_genderdata_package()
#install.packages("gender")
#install.packages("devtools")
#devtools::install_github("ropensci/genderdata", type = "source")

#install.packages("path/to/wru_package_directory", repos = NULL, type = "source")
#install.packages("ethnicolr")
#install.packages("wru")

library(gender)
```

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

```r
library(arrow)
```

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

##
## Attaching package: 'arrow'

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

```r
library(dplyr)
```

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

##
## 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
```

```r
library(tidyr)
```

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

```r
library(wru)
```

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

```
##
## 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.
```

```r
library(lubridate)
```

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

```
##
## Attaching package: 'lubridate'
```

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

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

```r
library(igraph)
```

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

```
##
## Attaching package: 'igraph'
```

```
## The following objects are masked from 'package:lubridate':
##
##     %--%, union
```

```
## The following object is masked from 'package:tidyr':
##
##     crossing
```

```
## The following objects are masked from 'package:dplyr':
##
##     as_data_frame, groups, union

## The following objects are masked from 'package:stats':
##
##     decompose, spectrum

## The following object is masked from 'package:base':
##
##     union
```

```
# set option to view all columns
options(dplyr.width = Inf)
```

```
# Read Parquet file
parquet_file <- "D:\\Google Drive\\McGill\\Winter Semester\\W2\\Talent-Analytics-Assignments\\Part 2\\E
applications  <- read_parquet(parquet_file)

# Read CSV file
edge_link <- "D:\\Google Drive\\McGill\\Winter Semester\\W2\\Talent-Analytics-Assignments\\Part 2\\Exer
edges <- read.csv(edge_link)
```

```
str(applications)
```

```
## tibble [2,018,477 x 16] (S3: tbl_df/tbl/data.frame)
##  $ application_number  : chr [1:2018477] "08284457" "08413193" "08531853" "08637752" ...
##  $ filing_date         : Date[1:2018477], format: "2000-01-26" "2000-10-11" ...
##  $ examiner_name_last  : chr [1:2018477] "HOWARD" "YILDIRIM" "HAMILTON" "MOSHER" ...
##  $ examiner_name_first : chr [1:2018477] "JACQUELINE" "BEKIR" "CYNTHIA" "MARY" ...
##  $ examiner_name_middle: chr [1:2018477] "V" "L" NA NA ...
##  $ examiner_id         : num [1:2018477] 96082 87678 63213 73788 77294 ...
##  $ examiner_art_unit   : num [1:2018477] 1764 1764 1752 1648 1762 ...
##  $ uspc_class          : chr [1:2018477] "508" "208" "430" "530" ...
##  $ uspc_subclass       : chr [1:2018477] "273000" "179000" "271100" "388300" ...
##  $ patent_number       : chr [1:2018477] "6521570" "6440298" "5607816" "6927281" ...
##  $ patent_issue_date   : Date[1:2018477], format: "2003-02-18" "2002-08-27" ...
##  $ abandon_date        : Date[1:2018477], format: NA NA ...
##  $ disposal_type       : chr [1:2018477] "ISS" "ISS" "ISS" "ISS" ...
##  $ appl_status_code    : num [1:2018477] 150 250 250 250 161 150 135 161 161 250 ...
##  $ appl_status_date    : chr [1:2018477] "30jan2003 00:00:00" "27sep2010 00:00:00" "30mar2009 00:00:
##  $ tc                  : num [1:2018477] 1700 1700 1700 1600 1700 1700 1600 1600 1600 1700 ...
```

```
# Guess the Gender
# get a list of first names without repetitions
examiner_names <- applications %>%
  distinct(examiner_name_first)

# 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) %>%
```

```r
  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) max used  (Mb)
## Ncells  4466168 238.6    8160662 435.9  4487900 239.7
## Vcells 49430640 377.2   92905758 708.9 79746856 608.5
```

```r
# Guess the examiner's race

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.
```

```r
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_
```

```r
  ))

# 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) max used  (Mb)
## Ncells  4570006 244.1    8160662 435.9  6093609 325.5
## Vcells 51639394 394.0   92905758 708.9 92889890 708.7
```

```r
applications <- applications %>%
  mutate(
    filing_date = as.Date(filing_date),
    patent_issue_date = as.Date(patent_issue_date),
    abandon_date = as.Date(abandon_date),
    final_decision_date = coalesce(patent_issue_date, abandon_date),
    app_proc_time = as.numeric(final_decision_date - filing_date),
    # Replace negative app_proc_time with NA
    app_proc_time = ifelse(app_proc_time < 0, NA, app_proc_time)
  )
```

```r
library(dplyr)
library(tidygraph)
```

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

##
## Attaching package: 'tidygraph'

## The following object is masked from 'package:igraph':
##
##     groups

## The following object is masked from 'package:stats':
##
##     filter
```

```r
library(ggraph)
```

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

## Loading required package: ggplot2

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

```r
edges <- edges %>%
  mutate(
    from = as.character(ego_examiner_id),
    to = as.character(alter_examiner_id)
  ) %>%
  mutate(
    from = ifelse(is.nan(as.numeric(from)), NA, from),
    to = ifelse(is.nan(as.numeric(to)), NA, to)
  ) %>%
  drop_na()

applications <- applications %>%
  relocate(examiner_id, .before = application_number) %>%
  mutate(examiner_id = as.character(examiner_id)) %>%
  drop_na(examiner_id) %>%
  rename(name = examiner_id)


graph <- tbl_graph(
  edges = (edges %>% relocate(from, to)),
  directed = TRUE
)

applications <- applications %>%
  mutate(name = as.character(name)) %>%
  distinct(name, .keep_all = TRUE)

graph <- graph %>%
  activate(nodes) %>%
  inner_join(
    (applications %>% distinct(name, .keep_all = TRUE)),
    by = "name"
  )
```

```r
graph %>%
  activate(nodes) %>%
  mutate(
    degree = centrality_degree(),
    betweenness = centrality_betweenness(),
    closeness = centrality_closeness()
  ) %>%
  select(name, degree, betweenness, closeness) %>%
  arrange(-degree)
```

```
## # A tbl_graph: 2504 nodes and 17809 edges
## #
## # A directed multigraph with 130 components
## #
## # Node Data: 2,504 x 4 (active)
##    name  degree betweenness closeness
##    <chr>  <dbl>       <dbl>     <dbl>
##  1 83670    198         0    0.000403
##  2 97910    176       132.   0.00787
```

```
##   3 73920    174        0     0.00971
##   4 67226    122      876.    0.00746
##   5 80730    120        0     0.000286
##   6 75615    117        0     0.000457
##   7 62152    115        0     0.000324
##   8 69098    115        3.00  0.333
##   9 67690    114        0     0.0333
##  10 74061    114     2689.    0.000454
## # i 2,494 more rows
## #
## # Edge Data: 17,809 x 6
##    from    to application_number advice_date ego_examiner_id alter_examiner_id
##   <int> <int>              <int> <chr>                 <int>             <int>
## 1   158  1462            9402488 2008-11-17            84356             66266
## 2   158  1463            9402488 2008-11-17            84356             63519
## 3   158  1464            9402488 2008-11-17            84356             98531
## # i 17,806 more rows
```

```
node_data <- graph %>%
  activate(nodes) %>%
  mutate(
    degree = centrality_degree(),
    betweenness = centrality_betweenness(),
    closeness = centrality_closeness()
  ) %>%
  select(name, degree, betweenness, closeness) %>%
  as_tibble() # Convert to a tibble/data frame for joining

# Joining the centrality measures back to the applications dataframe
applications <- applications %>%
  left_join(node_data, by = c("name" = "name"))

# rename name to examiner_id
applications <- applications %>%
  rename(examiner_id = name)

head(applications,5)
```

```
## # A tibble: 5 x 23
##   examiner_id application_number filing_date examiner_name_last
##   <chr>       <chr>              <date>      <chr>
## 1 96082       08284457           2000-01-26  HOWARD
## 2 87678       08413193           2000-10-11  YILDIRIM
## 3 63213       08531853           2000-05-17  HAMILTON
## 4 73788       08637752           2001-07-20  MOSHER
## 5 77294       08682726           2000-04-10  BARR
##   examiner_name_first examiner_name_middle examiner_art_unit uspc_class
##   <chr>               <chr>                            <dbl> <chr>
## 1 JACQUELINE          V                                 1764 508
## 2 BEKIR               L                                 1764 208
## 3 CYNTHIA             <NA>                              1752 430
## 4 MARY                <NA>                              1648 530
## 5 MICHAEL             E                                 1762 427
##   uspc_subclass patent_number patent_issue_date abandon_date disposal_type
```

```
##   <chr>          <chr>          <date>            <date>       <chr>
## 1 273000         6521570        2003-02-18        NA           ISS
## 2 179000         6440298        2002-08-27        NA           ISS
## 3 271100         5607816        1997-03-04        NA           ISS
## 4 388300         6927281        2005-08-09        NA           ISS
## 5 430100         <NA>           NA                2000-12-27   ABN
##   appl_status_code appl_status_date      tc gender race  final_decision_date
##              <dbl> <chr>              <dbl> <chr>  <chr> <date>
## 1              150 30jan2003 00:00:00  1700 female white 2003-02-18
## 2              250 27sep2010 00:00:00  1700 <NA>   white 2002-08-27
## 3              250 30mar2009 00:00:00  1700 female white 1997-03-04
## 4              250 07sep2009 00:00:00  1600 female white 2005-08-09
## 5              161 19apr2001 00:00:00  1700 male   white 2000-12-27
##   app_proc_time degree betweenness closeness
##           <dbl>  <dbl>       <dbl>     <dbl>
## 1          1119     NA          NA        NA
## 2           685     NA          NA        NA
## 3            NA      0           0       NaN
## 4          1481      2           0       0.5
## 5           261      0           0       NaN
```

```
#null values in applications data each column
sapply(applications, function(x) sum(is.na(x)))
```

```
##          examiner_id   application_number          filing_date
##                    0                    0                    0
##   examiner_name_last  examiner_name_first examiner_name_middle
##                    0                    0                 1370
##    examiner_art_unit           uspc_class        uspc_subclass
##                    0                    0                    0
##        patent_number    patent_issue_date         abandon_date
##                 2606                 2605                 3399
##        disposal_type     appl_status_code     appl_status_date
##                    0                    1                    1
##                   tc               gender                 race
##                    0                  799                    0
##  final_decision_date        app_proc_time               degree
##                  356                  357                 3144
##          betweenness            closeness
##                 3144                 4211
```

```
# total rows in applications data
nrow(applications)
```

```
## [1] 5648
```

```
# Dropping rows with NA in regression columns
applications <- applications %>%
  drop_na(app_proc_time, degree, gender, examiner_art_unit, uspc_class,disposal_type,race)
```

Build linear regression model

```r
applications <- applications %>%
  mutate(
    examiner_art_unit = as.factor(examiner_art_unit),
    uspc_class = as.factor(uspc_class),
    gender = as.factor(gender),
    race = as.factor(race),
    disposal_type = as.factor(disposal_type)
  )
```

I wanted to use examiner_art_unit, uspc_class as categorical variable but considering ther are too many they are not added as features

disposal_type categorical varaible is used because it tellls about the status of appplication"ISS" (issued), "ABN" (abandoned), "PEND' (PENDING). There must be a differnce in processing times for each of the category

Race is used as well to understand affect of race in processing times

```r
#Model 1: Degree Centrality with Categorical Variables

model_degree <- lm(app_proc_time ~ degree +race +disposal_type , data = applications)
summary(model_degree)
```

```
##
## Call:
## lm(formula = app_proc_time ~ degree + race + disposal_type, data = applications)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2464.0  -808.2  -240.2   678.9  4275.1
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       1869.99      50.59  36.963  < 2e-16 ***
## degree              10.29       1.49   6.908 6.51e-12 ***
## raceblack           96.09     135.31   0.710   0.4777
## raceHispanic        26.14     146.64   0.178   0.8586
## raceother         -542.76     751.52  -0.722   0.4702
## racewhite         -126.15      50.89  -2.479   0.0133 *
## disposal_typeISS    92.74      47.03   1.972   0.0488 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1061 on 2088 degrees of freedom
## Multiple R-squared:  0.02804,    Adjusted R-squared:  0.02525
## F-statistic: 10.04 on 6 and 2088 DF,  p-value: 5.821e-11
```

Model 1: Degree Centrality with Categorical Variables Model Formula: `app_proc_time ~ degree + race + disposal_type`

- **Degree Centrality**: The coefficient for **`degree`** is positive (Estimate = 10.29), indicating that as an examiner's network centrality increases, the application processing time also increases slightly. This could suggest that examiners central to the network may be involved in more complex or a higher volume of cases, potentially leading to longer processing times

- **Race**: The model considered race as a categorical variable. Notably, **`racewhite`** has a negative coefficient (Estimate = -126.15), suggesting that applications handled by white examiners are associated with slightly shorter processing times compared to the baseline race category.

- **Disposal Type**: **`disposal_typeISS`** (indicating a patent was issued) is positively associated with processing time (Estimate = 92.74), which might reflect the additional scrutiny and time required for applications that eventually get approved

```
#Model 2: Betweenness Centrality with Categorical Variables

model_betweenness <- lm(app_proc_time ~ betweenness +race +disposal_type, data = applications)
summary(model_betweenness)
```

```
##
## Call:
## lm(formula = app_proc_time ~ betweenness + race + disposal_type,
##     data = applications)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1761.5  -797.8  -232.1   687.1  4219.5
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       1.943e+03  5.008e+01  38.798   <2e-16 ***
## betweenness       8.116e-03  8.704e-03   0.932   0.3512
## raceblack         8.642e+01  1.368e+02   0.632   0.5278
## raceHispanic      4.138e+01  1.483e+02   0.279   0.7803
## raceother        -5.457e+02  7.599e+02  -0.718   0.4728
## racewhite        -1.268e+02  5.152e+01  -2.461   0.0139 *
## disposal_typeISS  8.613e+01  4.755e+01   1.811   0.0702 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1073 on 2088 degrees of freedom
## Multiple R-squared:  0.006245,   Adjusted R-squared:  0.003389
## F-statistic: 2.187 on 6 and 2088 DF,  p-value: 0.04158
```

Model 2: Betweenness Centrality with Categorical Variables **Model Formula**: **`app_proc_time ~ betweenness + race + disposal_type`**

- **Betweenness Centrality**: The coefficient for **`betweenness`** is not statistically significant (Estimate = 8.166e-03, p-value = 0.3512), indicating that betweenness centrality might not have a clear impact on processing time in this model setup. This suggests that an examiner's role as a connector in the network does not significantly affect application processing times.

```r
#Model 3: Degree Centrality with Gender Interaction and Categorical Variables

model_degree_gender <- lm(app_proc_time ~ degree * gender + +race +disposal_type, data = applications)
summary(model_degree_gender)
```

```
##
## Call:
## lm(formula = app_proc_time ~ degree * gender + +race + disposal_type,
##     data = applications)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2336.0  -796.3  -236.9   667.6  4388.8
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        1762.076     63.567  27.720  < 2e-16 ***
## degree               12.533      2.736   4.581  4.9e-06 ***
## gendermale          157.424     56.247   2.799  0.00518 **
## raceblack           107.099    135.196   0.792  0.42835
## raceHispanic         25.608    146.580   0.175  0.86133
## raceother          -584.969    750.637  -0.779  0.43589
## racewhite          -132.025     50.887  -2.594  0.00954 **
## disposal_typeISS     90.661     46.977   1.930  0.05375 .
## degree:gendermale    -3.204      3.262  -0.982  0.32606
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1060 on 2086 degrees of freedom
## Multiple R-squared:  0.03169,    Adjusted R-squared:  0.02798
## F-statistic: 8.534 on 8 and 2086 DF,  p-value: 1.712e-11
```

Model 3: Degree Centrality with Gender Interaction **Model Formula**: **`app_proc_time ~ degree * gender + race + disposal_type`**

- **Degree and Gender Interaction**: The interaction term **`degree:gendermale`** is not significant (Estimate = 157.424, p-value = 0.00518), indicating that the effect of degree centrality on processing time does differ significantly between male and female examiners in this model.

```r
#Model 4: Betweenness Centrality with Gender Interaction and Categorical Variables
model_betweenness_gender <- lm(app_proc_time ~ betweenness * gender +race +disposal_type, data = applica
summary(model_betweenness_gender)
```

```
##
## Call:
## lm(formula = app_proc_time ~ betweenness * gender + race + disposal_type,
##     data = applications)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1740.8  -807.7  -242.5   685.1  4312.9
##
```

```
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)            1854.34166   61.40286  30.200   <2e-16 ***
## betweenness              -0.02409    0.02883  -0.835   0.4035
## gendermale              128.72866   52.47789   2.453   0.0142 *
## raceblack                96.87994  136.71236   0.709   0.4786
## raceHispanic             34.89838  148.13060   0.236   0.8138
## raceother              -585.87196  759.00019  -0.772   0.4403
## racewhite              -131.82748   51.52873  -2.558   0.0106 *
## disposal_typeISS         86.60328   47.50330   1.823   0.0684 .
## betweenness:gendermale    0.03452    0.03025   1.141   0.2539
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1071 on 2086 degrees of freedom
## Multiple R-squared:  0.01004,    Adjusted R-squared:  0.006245
## F-statistic: 2.645 on 8 and 2086 DF,  p-value: 0.006933
```

Model 4: Betweenness Centrality with Gender Interaction **Model Formula**: **`app_proc_time ~ betweenness * gender + race + disposal_type`**

- **Betweenness and Gender Interaction**: Similar to degree centrality, the interaction between **`betweenness`** and **`gendermale`** is statistically significant (Estimate = 128, p-value = 0.0142), suggesting significant difference in the effect of betweenness centrality on processing times across genders.

Explaining Regression Results and Implications for the USPTO

Conclusion and Implications for the USPTO

The analysis using linear regression models aimed to explore the influence of examiner centrality within the USPTO's network on patent application processing times, while considering other examiner characteristics and examining potential differences by gender. The findings suggest a slight increase in processing times with higher degree centrality but no clear impact from betweenness centrality. This increase might be attributed to the potential complexity and volume of cases handled by more central examiners

Implications of Centrality on Operational Efficiency

1. Enhanced Resource Allocation: The positive association between degree centrality and increased processing times implies that examiners who are more central to the network—those with more connections—might be dealing with a higher workload or more complex cases, potentially leading to delays. Recognizing this pattern, the USPTO could consider strategies for resource allocation that support central examiners, such as redistributing workload or providing additional administrative support, to streamline the processing timeline without sacrificing the quality of patent examination

Examination of Gender Interaction

Our analysis revealed that the interaction terms between gender and centrality measures (degree and betweenness) were statistically significant. This indicates that the impact of an examiner's centrality within the USPTO's network on application processing times does significantly differ between male and female examiners.

Implications for the USPTO

Operational Insights: The significant interaction effect between gender and centrality suggests that gender does modify how centrality influences processing times. This finding could imply that the USPTO's internal processes and examiner networks function in a manner that is different for both genders and they should deploy strategies to make sure both genders have same processing times if other conditions are the same. Moreover Race should be accounted as well and it should be noted that race have differnt processing times understanding why this is so could lead to the development of solutions