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.
```

```
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)
```

```r
# 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  4530355  242    8044735 429.7         NA  4549841 243.0
## Vcells 49663632  379   93185307 711.0      16384 79979447 610.2
```

```r
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.
```

```r
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  4738566 253.1    8044735 429.7         NA  6962527 371.9
## Vcells 52052726 397.2   93185307 711.0      16384 92293283 704.2
```

```
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  4747514 253.6    8044735 429.7         NA   8044735 429.7
## Vcells 58128959 443.5  111902368 853.8      16384 111623313 851.7
```

Pick two art_units

176 and 218

```r
workgroup_176 <- applications %>%
        filter(substr(examiner_art_unit, 1, 3) == '176')

workgroup_218 <- applications %>%
        filter(substr(examiner_art_unit, 1, 3) == '218')

# Summary Statistics for Workgroup 367
cat("Summary for Workgroup 176:\n")
```

```
## Summary for Workgroup 176:
```

```r
workgroup_176 %>% select(c(race, gender)) %>% table() %>% print()
```

```
##           gender
## race       female  male
##   Asian      8094  9954
##   Hispanic    658  1331
##   black      3230     0
##   white     16093 42276
```

```r
# Summary Statistics for Workgroup 765
cat("\nSummary for Workgroup 218:\n")
```

```
##
## Summary for Workgroup 218:
```

```r
workgroup_218 %>% select(c(race, gender)) %>% table() %>% print()
```

```
##           gender
## race       female  male
##   Asian      2626 14255
##   Hispanic    361  1539
##   black         0  1487
##   white      3185 24978
```

Other Summary Stats

```r
summary(workgroup_176 %>% select(tenure_days)) %>% print()
```

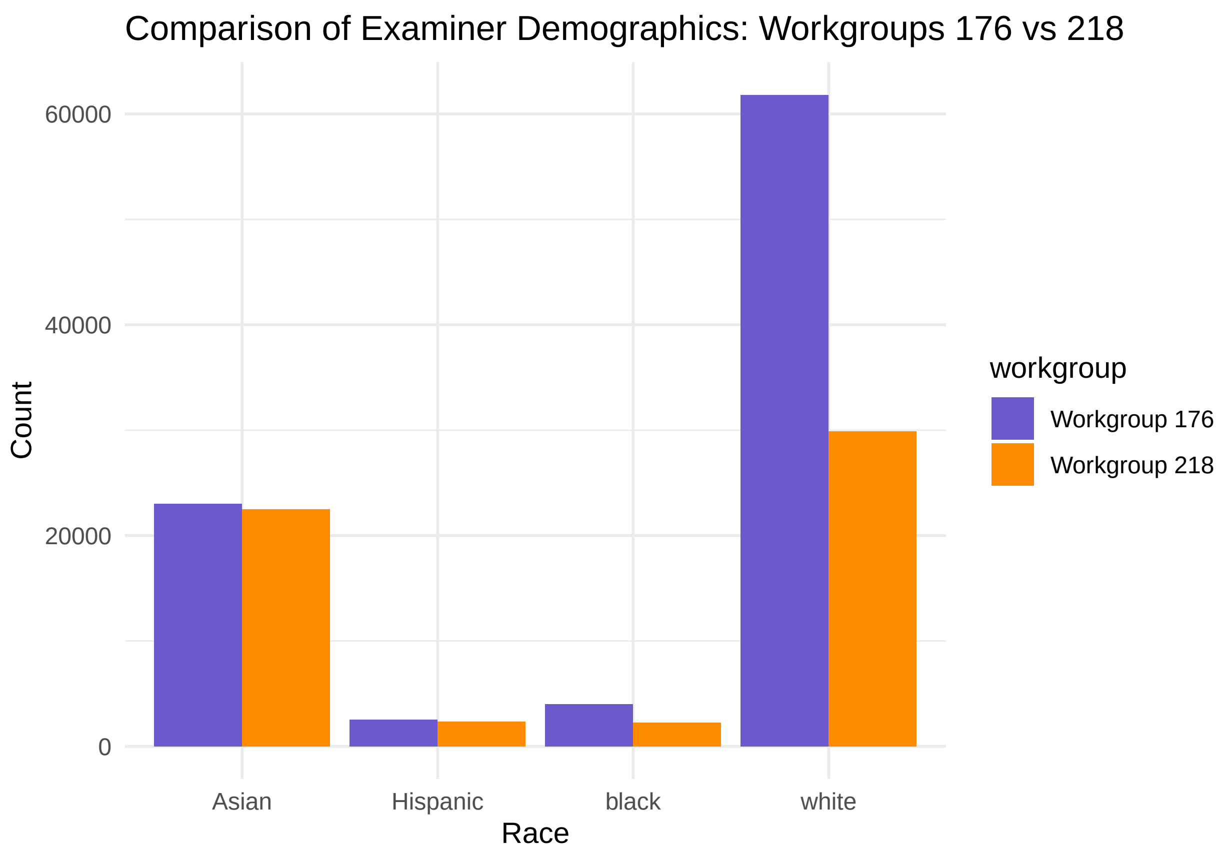
```
##   tenure_days
##  Min.   : 339
##  1st Qu.:4524
##  Median :6294
##  Mean   :5501
##  3rd Qu.:6342
##  Max.   :6350
##  NA's   :1017
```

```r
summary(workgroup_218 %>% select(tenure_days)) %>% print()
```

```
##   tenure_days
##  Min.   : 633
##  1st Qu.:5307
##  Median :6015
##  Mean   :5705
##  3rd Qu.:6322
##  Max.   :6349
##  NA's   :160
```

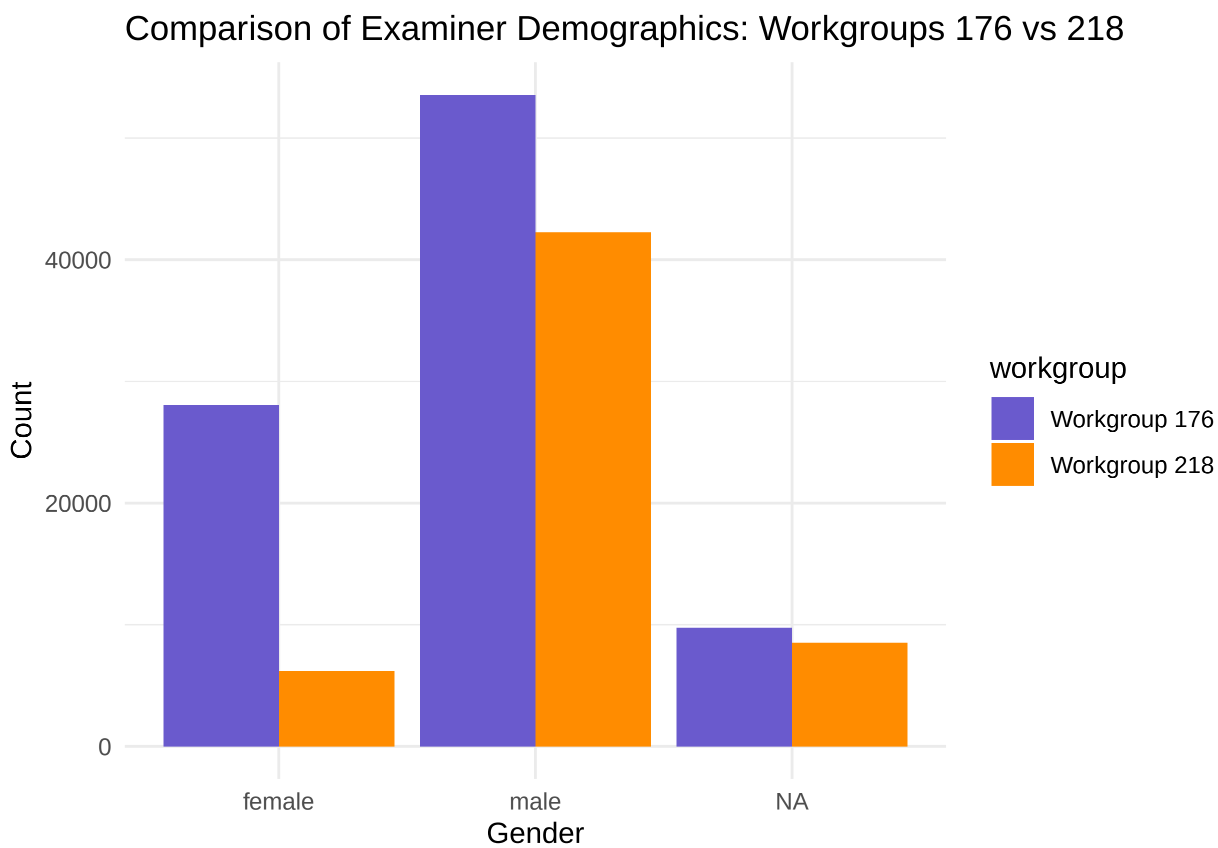
Plotting - distribution of race

```r
# Combine the two groups for plotting, adding a workgroup identifier
applications_combined <- applications %>%
        filter(substr(examiner_art_unit, 1, 3) %in% c('176', '218')) %>%
        mutate(workgroup = case_when(
                substr(examiner_art_unit, 1, 3) == '176' ~ 'Workgroup 176',
                substr(examiner_art_unit, 1, 3) == '218' ~ 'Workgroup 218'
        ))

# Plotting the demographics comparison
ggplot(applications_combined, aes(x = race, fill = workgroup)) +
        geom_bar(position = "dodge") +
        labs(title = "Comparison of Examiner Demographics: Workgroups 176 vs 218",
             x = "Race",
             y = "Count") +
        theme_minimal() +
        scale_fill_manual(values = c("Workgroup 176" = "slateblue", "Workgroup 218" = "darkorange"))
```



Plotting - distribution of gender

```
applications_combined <- applications %>%
        filter(substr(examiner_art_unit, 1, 3) %in% c('176', '218')) %>%
        mutate(workgroup = case_when(
                substr(examiner_art_unit, 1, 3) == '176' ~ 'Workgroup 176',
                substr(examiner_art_unit, 1, 3) == '218' ~ 'Workgroup 218'
        ))

# Plotting the demographics comparison
ggplot(applications_combined, aes(x = gender, fill = workgroup)) +
        geom_bar(position = "dodge") +
        labs(title = "Comparison of Examiner Demographics: Workgroups 176 vs 218",
             x = "Gender",
             y = "Count") +
        theme_minimal() +
        scale_fill_manual(values = c("Workgroup 176" = "slateblue", "Workgroup 218" = "darkorange"))
```



Creating advice edge 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"
```

Calculate the centrality measures

```
# Calculate degree centrality for each node (examiner)
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
)
```

```r
workgroup_176$examiner_id <- as.character(workgroup_176$examiner_id)
centrality_scores$examiner_id <- as.character(centrality_scores$examiner_id)

# Merge the centrality scores with the applications data for workgroup 176
applications_176_with_scores <- workgroup_176 %>%
  left_join(centrality_scores, by = "examiner_id")

# Repeat for workgroup 218, ensuring type consistency
workgroup_218$examiner_id <- as.character(workgroup_218$examiner_id)
applications_218_with_scores <- workgroup_218 %>%
  left_join(centrality_scores, by = "examiner_id")
```

How does between centrality "affect" or correlate with employee characteristic?

```r
# Correlation analysis between centrality and tenure_days
cor.test(applications_176_with_scores$degree, applications_176_with_scores$tenure_days, use = "complete
```

```
##
##  Pearson's product-moment correlation
##
## data:  applications_176_with_scores$degree and applications_176_with_scores$tenure_days
## t = 11.928, df = 60414, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.04051526 0.05642570
## sample estimates:
##        cor
## 0.04847356
```

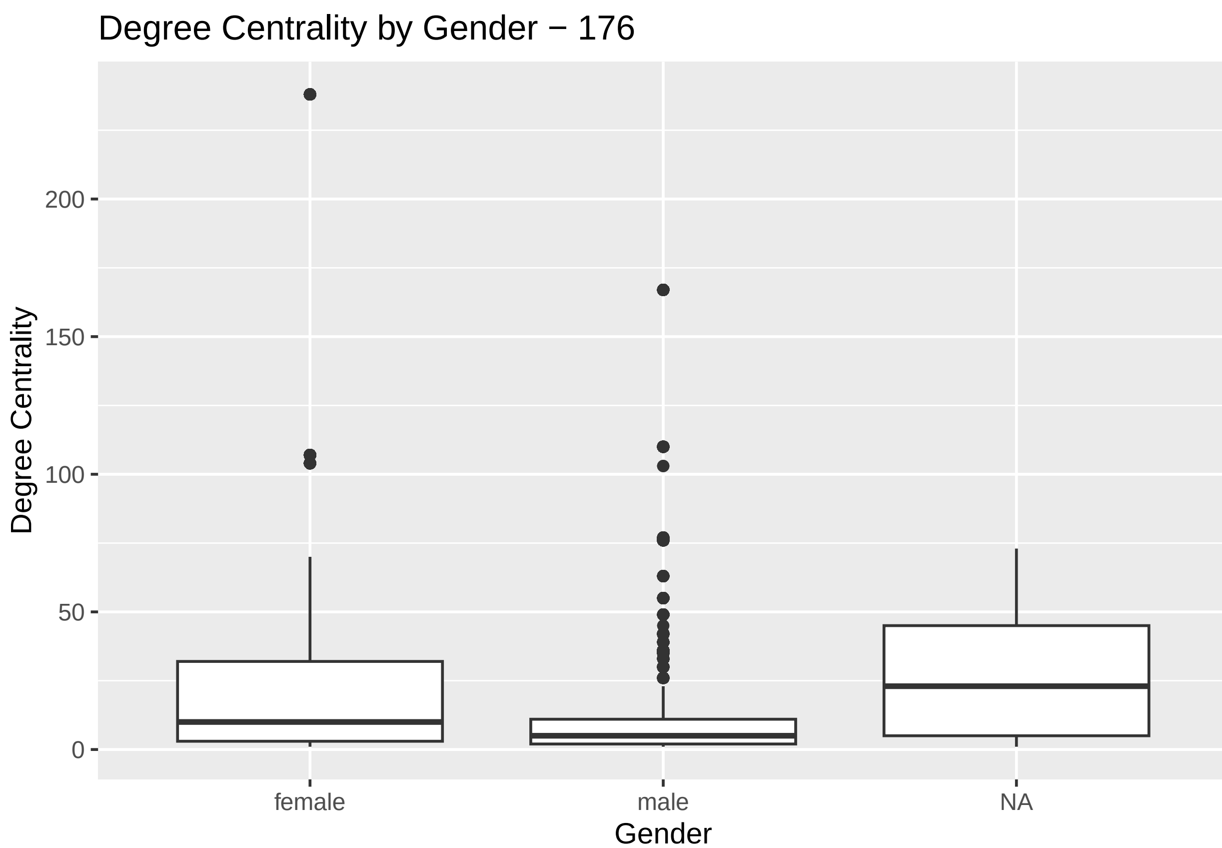
```r
cor.test(applications_176_with_scores$betweenness, applications_176_with_scores$tenure_days, use = "comp
```

```
##
##  Pearson's product-moment correlation
##
## data:  applications_176_with_scores$betweenness and applications_176_with_scores$tenure_days
## t = 8.0036, df = 60414, p-value = 1.23e-15
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.02457762 0.04050864
## sample estimates:
##       cor
## 0.0325452
```

- slight pos corr

```r
# Create a boxplot of betweenness centrality by gender
ggplot(applications_176_with_scores, aes(x = gender, y = degree)) +
  geom_boxplot() +
  labs(title = "Degree Centrality by Gender - 176",
       x = "Gender", y = "Degree Centrality")
```

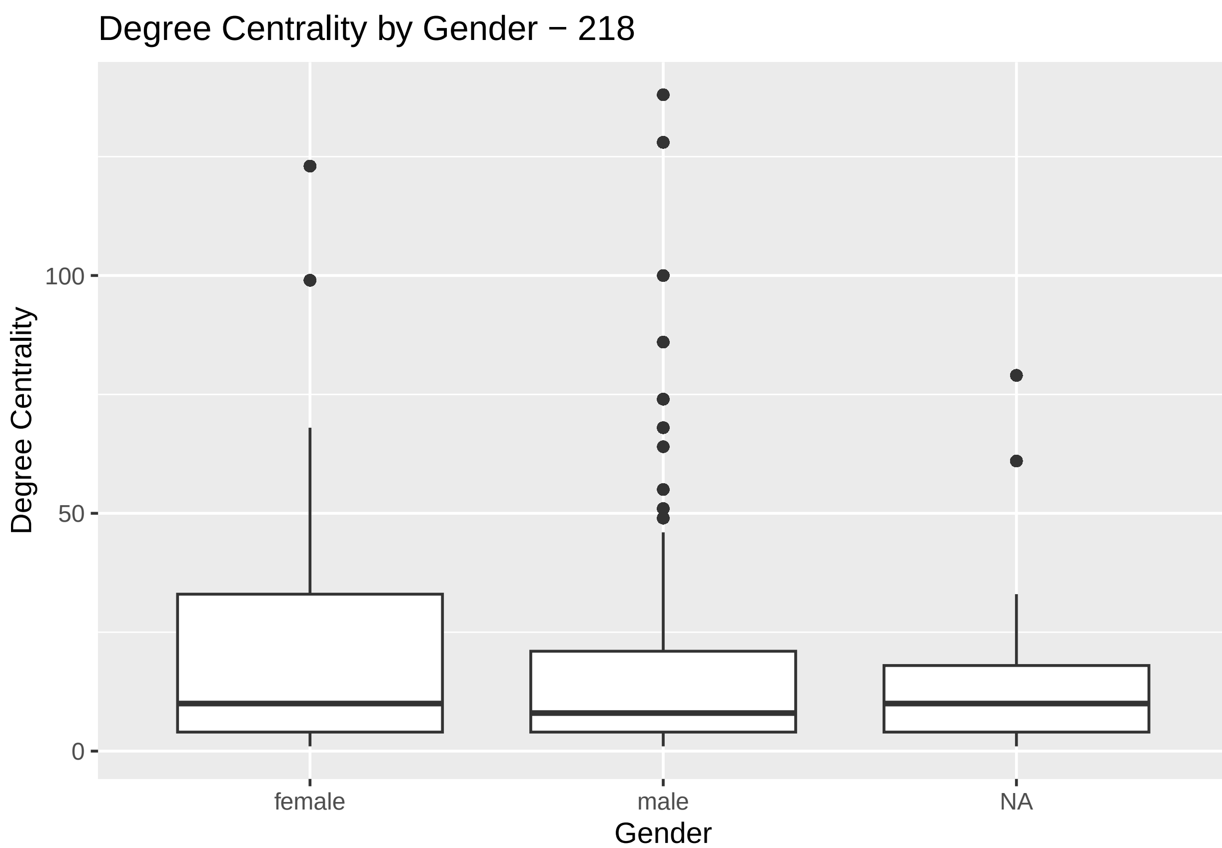
```
## Warning: Removed 30347 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).
```



- Higher degree centrality for female but NA (unidentified gender) is more hihger > This could indicate several things: if you think about how the gender became unidentified, it could show those groups who > were not identified are more likely to be more central in the network. This could be a good thing or a bad thing depending on the context.

```
# Create a boxplot of betweenness centrality by gender
ggplot(applications_218_with_scores, aes(x = gender, y = degree)) +
  geom_boxplot() +
  labs(title = "Degree Centrality by Gender - 218",
       x = "Gender", y = "Degree Centrality")
```

```
## Warning: Removed 16092 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).
```



- higher degree centrality for female examiners in workgroup 218
- For both, male is low but in this unit, NA is the lowest -> lower representation of non-US born examiners

It would be nice to look how these differ by race but it is getting too much for an exercise so i will end here.