# Exercise 3: Examiners' Demographics and Advice Networks

Liliana Tretyakova

March 27, 2023

### Introduction

In this assignment, we analyze the demographics of examiners in two selected workgroups and explore the advice networks within those workgroups. Specifically, we will:

1. Load the data files and add the following variables for examiners:

- Gender
- Race
- Tenure

2. Choose two workgroups and compare their demographics through summary statistics and plots.

3. Create advice networks from edges_sample and calculate centrality scores for examiners in the selected workgroups.

```
# Load required libraries
library(tidyverse)
library(lubridate)
library(arrow)
library(gender)
library(wru)
library(igraph)
library(ggplot2)

# Load data
applications <- read_parquet("C:/Users/ulyan/OneDrive - McGill University/Documents/MMA/Winter II 2023/
edges_sample <- read_csv("C:/Users/ulyan/OneDrive - McGill University/Documents/MMA/Winter II 2023/Org 
```

### Add Gender, Race, and Tenure Variables for Examiners

First, we will add the gender, race, and tenure variables to the examiners' data.

Here's how we can add gender variable:

```
# Get unique examiner first names
examiner_names <- applications %>% distinct(examiner_name_first)

# Predict gender based on first names
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)

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

Now, let's add race variable:

```
# Get unique examiner last names
examiner_surnames <- applications %>% select(surname = examiner_name_last) %>% distinct()

# Predict race based on last names
examiner_race <- predict_race(voter.file = examiner_surnames, surname.only = T) %>% as_tibble()
```

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

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

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

```
# Select the race with the highest probability for each last name
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_
  ))

# Join race data back to the main applications dataset
applications <- applications %>%
  left_join(examiner_race, by = c("examiner_name_last" = "surname"))
```

Finally, we estimate and add tenure variable:

```
# Extract examiner IDs and application dates
examiner_dates <- applications %>%
  select(examiner_id, filing_date, appl_status_date)

# Convert dates to a consistent format
examiner_dates <- examiner_dates %>%
  mutate(start_date = ymd(filing_date), end_date = as_date(dmy_hms(appl_status_date)))

# Calculate the earliest and latest dates for each examiner and their tenure in days
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)

# Join tenure data back to the main applications dataset
applications <- applications %>% left_join(examiner_dates, by = "examiner_id")
```

Now that we have added gender, race, and tenure variables to the examiners' data, let's proceed with the analysis.

#### Select and Compare Demographics of Two Workgroups

In this section of the exercise, we analyze and compare the demographics of two selected workgroups, Workgroup 161 and Workgroup 162. We first generate summary statistics and then visualize the demographics using bar plots. The main demographics of interest are gender, race, and tenure.

```
# Choose workgroups
workgroup1 <- "161"
workgroup2 <- "162"
```

## Summary statistics

We start by computing the summary statistics for the demographics of each workgroup. This includes the average tenure in days, the proportion of female examiners, and the total count of examiners in each workgroup.

```
# Filter the applications dataset for the chosen workgroups
workgroups_data <- applications %>%
  filter(substr(examiner_art_unit, 1, 3) %in% c(workgroup1, workgroup2))

# Summary statistics for demographics
summary_stats <- workgroups_data %>%
  group_by(workgroup = substr(examiner_art_unit, 1, 3)) %>%
  summarise(
    avg_tenure_days = mean(tenure_days, na.rm = TRUE),
    proportion_female = mean(proportion_female, na.rm = TRUE),
    count = n()
  ) %>%
  mutate(across(c(avg_tenure_days, proportion_female), round, 2))

# Print summary statistics
print(summary_stats)
```

```
## # A tibble: 2 x 4
##   workgroup avg_tenure_days proportion_female  count
##   <chr>               <dbl>             <dbl>  <int>
## 1 161                 5679.              0.49  89795
## 2 162                 5806.              0.48 141390
```

## Demographic Distribution Tables

Next, we will examine the demographic distributions of the workgroups in more detail by generating tables for gender, race, and tenure distributions.

```
# Filter the applications dataset for the chosen workgroups
workgroups_data <- applications %>%
  filter(substr(examiner_art_unit, 1, 3) %in% c(workgroup1, workgroup2))

# Gender distribution
gender_distribution <- workgroups_data %>%
  group_by(workgroup = substr(examiner_art_unit, 1, 3), gender) %>%
  summarise(count = n()) %>%
  arrange(workgroup, count, .by_group = TRUE)
```

```
## `summarise()` has grouped output by 'workgroup'. You can override using the
## `.groups` argument.
```

```
# Race distribution
race_distribution <- workgroups_data %>%
  group_by(workgroup = substr(examiner_art_unit, 1, 3), race) %>%
  summarise(count = n()) %>%
  arrange(workgroup, count, .by_group = TRUE)
```

```
## `summarise()` has grouped output by 'workgroup'. You can override using the
## `.groups` argument.
```

```
# Tenure distribution (grouped by years)
tenure_distribution <- workgroups_data %>%
  mutate(tenure_years = floor(tenure_days / 365)) %>%
  group_by(workgroup = substr(examiner_art_unit, 1, 3), tenure_years) %>%
  summarise(count = n()) %>%
  arrange(workgroup, tenure_years)
```

```
## `summarise()` has grouped output by 'workgroup'. You can override using the
## `.groups` argument.
```

```
# Display summary tables
print(gender_distribution)
```

```
## # A tibble: 6 x 3
## # Groups:   workgroup [2]
##   workgroup gender count
##   <chr>     <chr>  <int>
## 1 161       <NA>   12966
## 2 161       female 37275
## 3 161       male   39554
## 4 162       <NA>   34598
## 5 162       female 51412
## 6 162       male   55380
```

```r
print(race_distribution)
```

```
## # A tibble: 8 x 3
## # Groups:   workgroup [2]
##   workgroup race     count
##   <chr>     <chr>    <int>
## 1 161       Hispanic  1843
## 2 161       black     2452
## 3 161       Asian    19528
## 4 161       white    65972
## 5 162       Hispanic  3884
## 6 162       black    11023
## 7 162       Asian    35442
## 8 162       white    91041
```

```r
print(tenure_distribution, n=37)
```

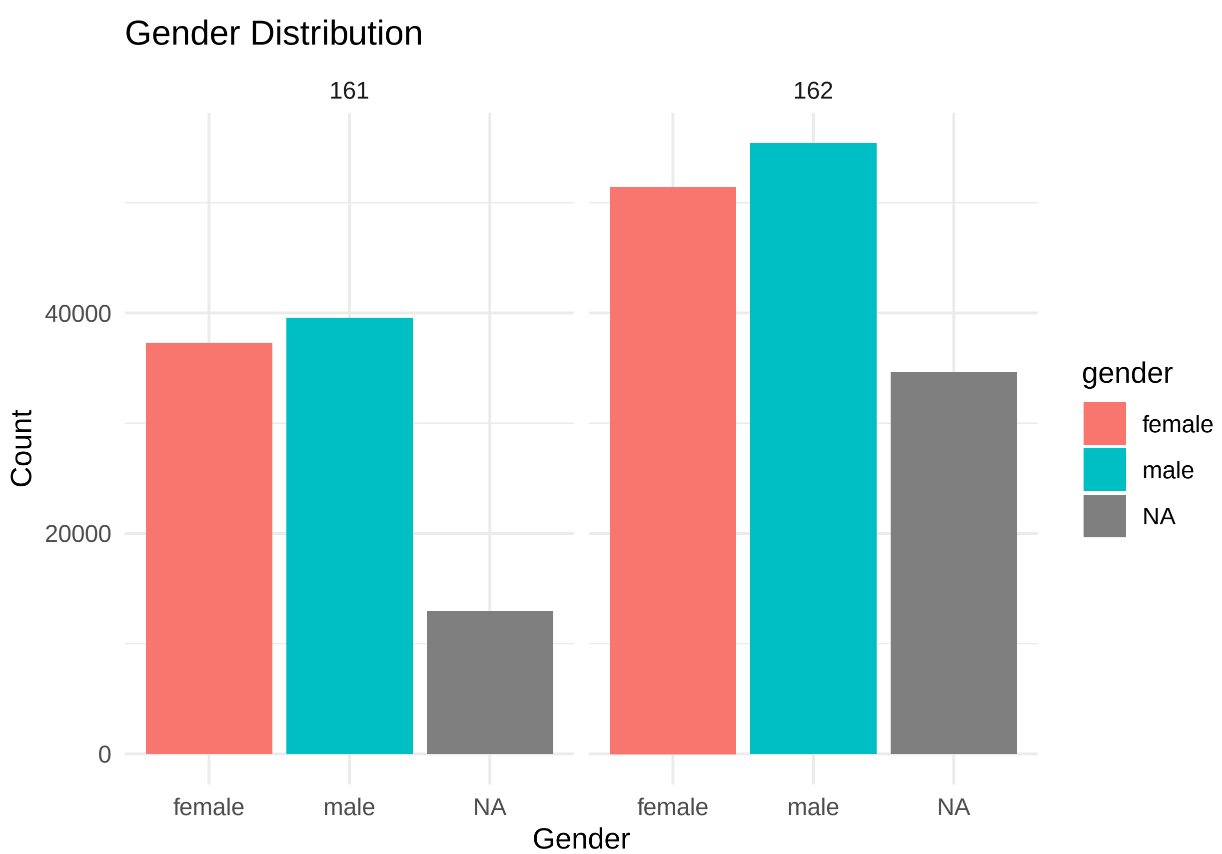
```
## # A tibble: 37 x 3
## # Groups:   workgroup [2]
##    workgroup tenure_years count
##    <chr>            <dbl> <int>
##  1 161                  0     1
##  2 161                  1     2
##  3 161                  2     4
##  4 161                  3   168
##  5 161                  4   233
##  6 161                  5   118
##  7 161                  6    12
##  8 161                  7   141
##  9 161                  8     4
## 10 161                  9   961
## 11 161                 10  1382
## 12 161                 11  2717
## 13 161                 12  4762
## 14 161                 13  7204
## 15 161                 14 10448
## 16 161                 15 11042
## 17 161                 16 14499
## 18 161                 17 32366
## 19 161                 NA  3731
## 20 162                  1     6
## 21 162                  2     1
## 22 162                  3     6
## 23 162                  4   203
## 24 162                  5    26
## 25 162                  6     6
## 26 162                  7   479
## 27 162                  8   759
## 28 162                  9  1308
## 29 162                 10  2412
## 30 162                 11  1594
## 31 162                 12  5134
```

```
## 32 162                13 10135
## 33 162                14 16627
## 34 162                15 10559
## 35 162                16 15547
## 36 162                17 72199
## 37 162                NA  4389
```

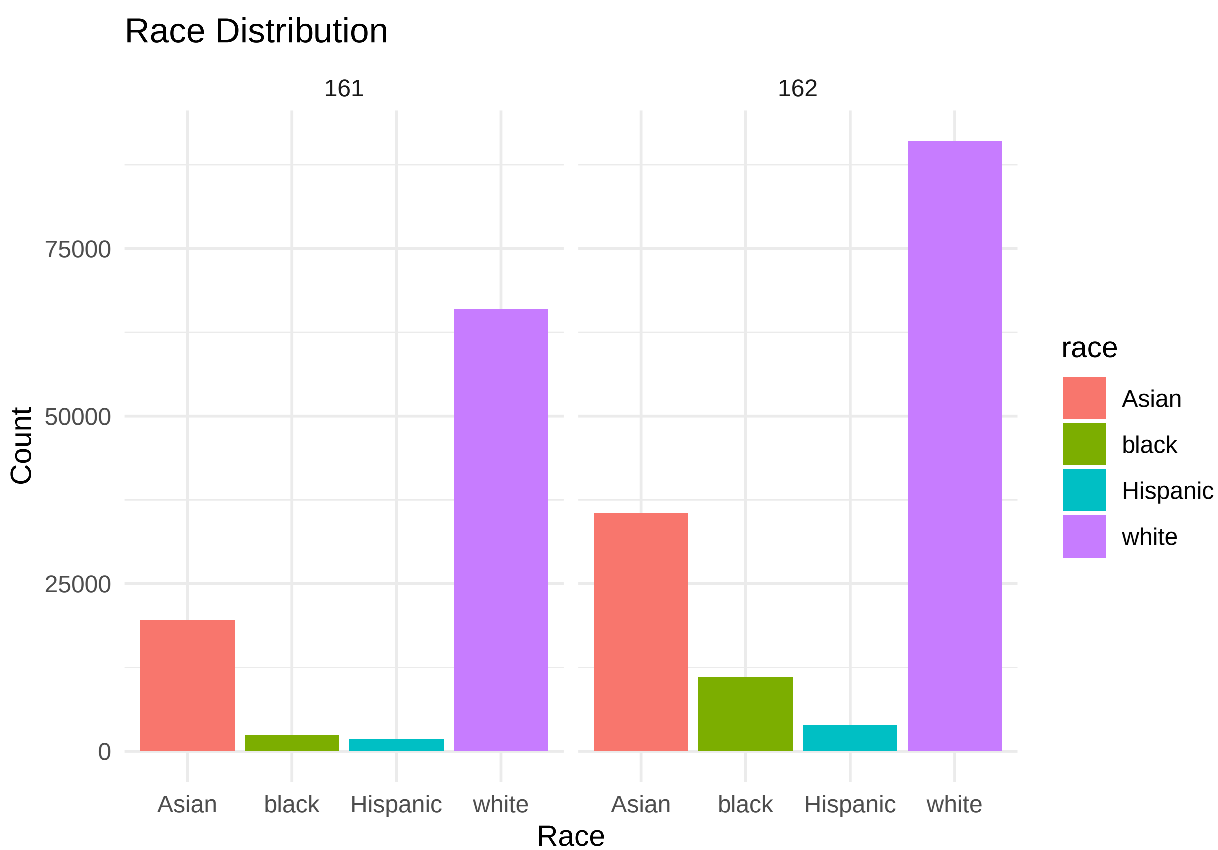
### Demographic Distribution Plots

To visualize the demographic distributions of the workgroups, we will create bar plots for gender and race distributions, as well as a histogram for the tenure distribution.

```
# Plot for gender distribution
gender_plot <- workgroups_data %>%
  ggplot(aes(x = gender, fill = gender)) +
  geom_bar() +
  facet_wrap(~substr(examiner_art_unit, 1, 3)) +
  labs(title = "Gender Distribution",
       x = "Gender",
       y = "Count") +
  theme_minimal()

# Display plot
print(gender_plot)
```

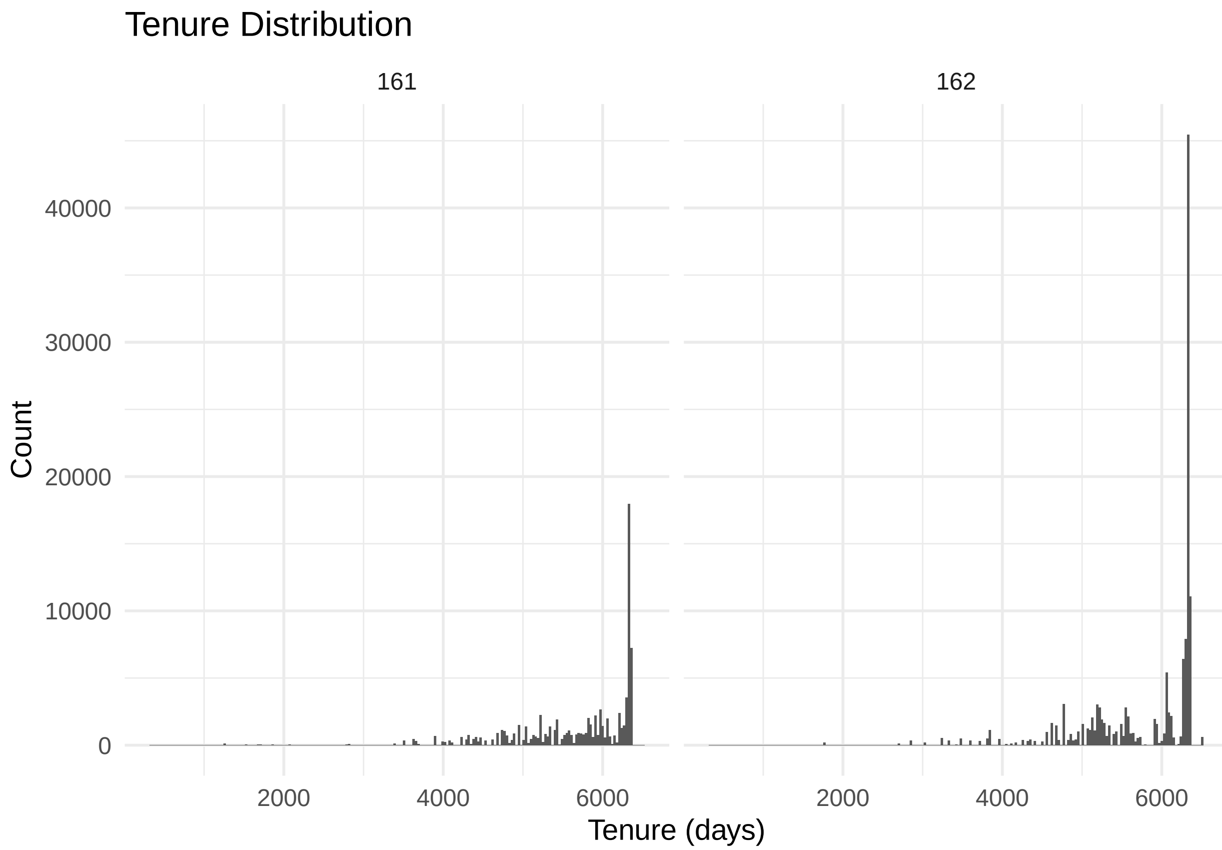


```r
# Plot for race distribution
race_plot <- workgroups_data %>%
  ggplot(aes(x = race, fill = race)) +
  geom_bar() +
  facet_wrap(~substr(examiner_art_unit, 1, 3)) +
  labs(title = "Race Distribution",
       x = "Race",
       y = "Count") +
  theme_minimal()

# Display plot
print(race_plot)
```




```r
# Plot for tenure distribution
tenure_plot <- workgroups_data %>%
  ggplot(aes(x = tenure_days)) +
  geom_histogram(binwidth = 30) +
  facet_wrap(~substr(examiner_art_unit, 1, 3)) +
  labs(title = "Tenure Distribution",
       x = "Tenure (days)",
       y = "Count") +
  theme_minimal()

# Display plot
print(tenure_plot)
```

```
## Warning: Removed 8120 rows containing non-finite values (stat_bin).
```




In comparing workgroups 161 and 162 on examiners' demographics, we can observe the following:

1. Gender Distribution: Both workgroups have slightly more male examiners than female examiners. However, there are also a considerable number of examiners with unknown gender in both workgroups.
2. Race Distribution: In both workgroups, the majority of examiners are White, followed by Asian, Black, and Hispanic examiners. Workgroup 162 has a larger number of examiners for each race compared to Workgroup 161.
3. Tenure Distribution: Both workgroups show a similar trend in tenure distribution, with the number of examiners generally increasing as the tenure in years increases. For both workgroups, the largest number of examiners fall into the 17-year tenure category. A considerable number of examiners in both workgroups have unknown tenure.

In summary, Workgroups 161 and 162 have similar demographic trends. The main difference between them is that Workgroup 162 has a larger number of examiners for each demographic category compared to Workgroup 161.

### Create Advice Networks and Calculate Centrality Scores

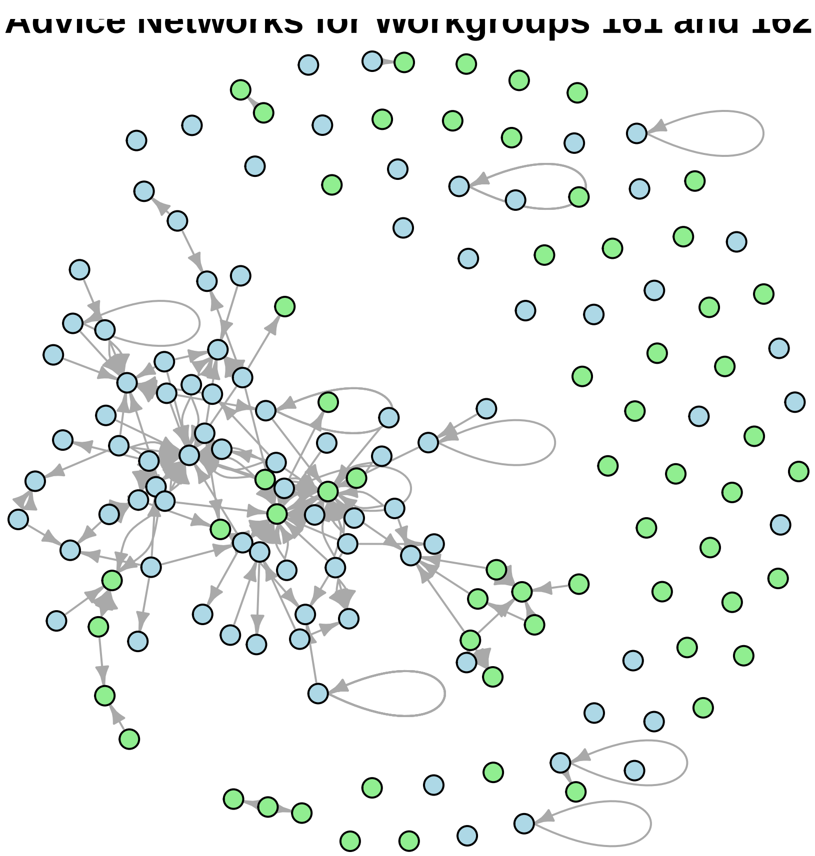
Next, we will create advice networks for the selected workgroups using the edges_sample dataset and calculate centrality scores for the examiners.

We will start with creating and plotting advice networks:

```r
# Create an igraph object from the edges_sample data
g <- graph_from_data_frame(edges_sample[, c("ego_examiner_id", "alter_examiner_id")], directed = TRUE)
```

```
## Warning in graph_from_data_frame(edges_sample[, c("ego_examiner_id",
## "alter_examiner_id")], : In ‘d’ ‘NA’ elements were replaced with string "NA"
```

```r
# Extract the first 3 digits of examiner_art_unit values
applications$workgroup <- substr(applications$examiner_art_unit, 1, 3)

# Create a mapping between examiner_id and workgroup in the applications dataset
examiner_workgroup_mapping <- applications %>%
  select(examiner_id, workgroup) %>%
  distinct()

# Add attributes to vertices in the network
V(g)$workgroup <- examiner_workgroup_mapping$workgroup[match(V(g)$name, examiner_workgroup_mapping$exami

# Filter the network to only include the two selected workgroups
g_filtered <- g %>%
  induced_subgraph(V(g)[V(g)$workgroup %in% c(workgroup1, workgroup2)])

# Set plot options
par(mar = c(0, 0, 0, 0))
set.seed(123)

# Create the plot
plot(g_filtered,
     vertex.color = ifelse(V(g_filtered)$workgroup == workgroup1, "lightblue", "lightgreen"),
     vertex.label = NA,
     vertex.size = 5,
     edge.arrow.size = 0.5,
     main = "Advice Networks for Workgroups 161 and 162")
```




## Calculate centralities

Now, we will calculate centrality scores for examiners in selected workgroups.

Since we need to ensure examiner_id has the same data type in both data frames, we will convert examiner_id to numeric in examiner_workgroup_mapping.

For this exercise, we have chosen to use Degree Centrality and Betweenness Centrality as our measures of centrality. Our choice is based on the following justifications:

1. Degree Centrality measures the number of direct connections an examiner has within the network. A higher degree centrality indicates that an examiner is directly connected to more examiners, which could imply that they collaborate frequently or share information with a large number of colleagues. As a result, examiners with high degree centrality can be considered influential or well-connected within the workgroup. This measure provides a straightforward way to quantify the local importance of an examiner in the network.

2. Betweenness Centrality, on the other hand, measures the extent to which an examiner lies on the shortest paths between other examiners in the network. Examiners with high betweenness centrality act as bridges or intermediaries between other examiners, connecting different parts of the network. This measure provides insight into the global importance of an examiner, as it considers their role in the overall network structure. High betweenness centrality may indicate that an examiner is crucial for communication or information flow within the workgroup.

By combining both Degree Centrality and Betweenness Centrality, we can gain a comprehensive understanding of an examiner's influence and connectivity within the workgroup. While degree centrality focuses on an examiner's local connections, betweenness centrality highlights their global role in the network. Using these measures together allows us to identify influential examiners in the network and better understand the overall structure and dynamics of the workgroup.

```
# Calculate Degree Centrality and Betweenness Centrality
degree_centrality <- degree(g_filtered, mode = "all")
betweenness_centrality <- betweenness(g_filtered, directed = TRUE)

# Add the centrality scores to the vertex attributes
V(g_filtered)$degree_centrality <- degree_centrality
V(g_filtered)$betweenness_centrality <- betweenness_centrality

# Merge centrality scores with the examiners' characteristics
centrality_scores <- data.frame(
  examiner_id = as.numeric(V(g_filtered)$name), # Convert examiner_id to numeric
  workgroup = V(g_filtered)$workgroup,
  degree_centrality = V(g_filtered)$degree_centrality,
  betweenness_centrality = V(g_filtered)$betweenness_centrality
)

applications_centrality <- applications %>%
  select(examiner_id, gender, race, tenure_days) %>%
  mutate(examiner_id = as.numeric(examiner_id)) %>% # Convert examiner_id to numeric
  inner_join(centrality_scores, by = "examiner_id")

# Examine the results
print(applications_centrality)
```

```
## # A tibble: 100,951 x 7
##    examiner_id gender race  tenure_days workgroup degree_centrality betweennes~1
##          <dbl> <chr>  <chr>       <dbl> <chr>                 <dbl>        <dbl>
##  1       70017 female Asian        6283 162                       0            0
##  2       69138 <NA>   white        6348 161                       4            1
##  3       64839 male   white        6254 161                       0            0
##  4       94939 <NA>   Asian        6336 162                       0            0
##  5       65737 female white        6129 162                       9            0
##  6       95225 male   white        6332 162                       0            0
##  7       68694 male   white        6350 161                       2            0
##  8       90588 female white        6343 161                       1            0
##  9       65536 female Asian        6345 162                       1            0
## 10       59399 male   white        6339 161                       1            0
## # ... with 100,941 more rows, and abbreviated variable name
## #   1: betweenness_centrality
```

### Characterize and discuss the relationship between centrality and other examiners' characteristics

In this section, we investigate the relationship between centrality measures (Degree and Betweenness Centrality) and examiners' characteristics such as tenure, race, and gender.

## Tenure

First, we calculate the correlations between centrality measures (Degree and Betweenness Centrality) and tenure_days.

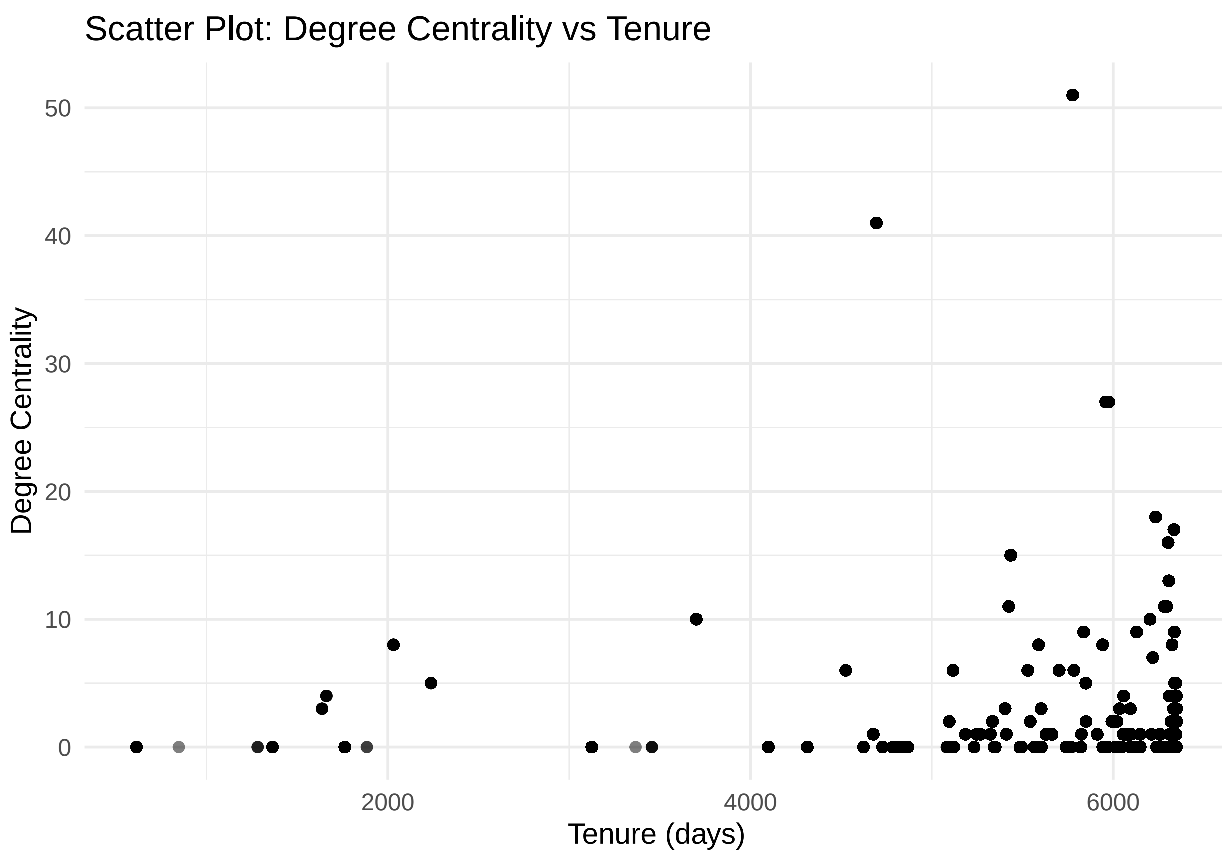
```
# Calculate correlations between centrality measures and tenure_days
correlation_results <- applications_centrality %>%
  select(degree_centrality, betweenness_centrality, tenure_days) %>%
  cor(use = "pairwise.complete.obs")

# Print the correlation results
print(correlation_results)
```

```
##                        degree_centrality betweenness_centrality tenure_days
## degree_centrality              1.0000000             0.45178302  0.05183140
## betweenness_centrality         0.4517830             1.00000000  0.03868923
## tenure_days                    0.0518314             0.03868923  1.00000000
```

Then, we visualize the relationships using scatter plots.

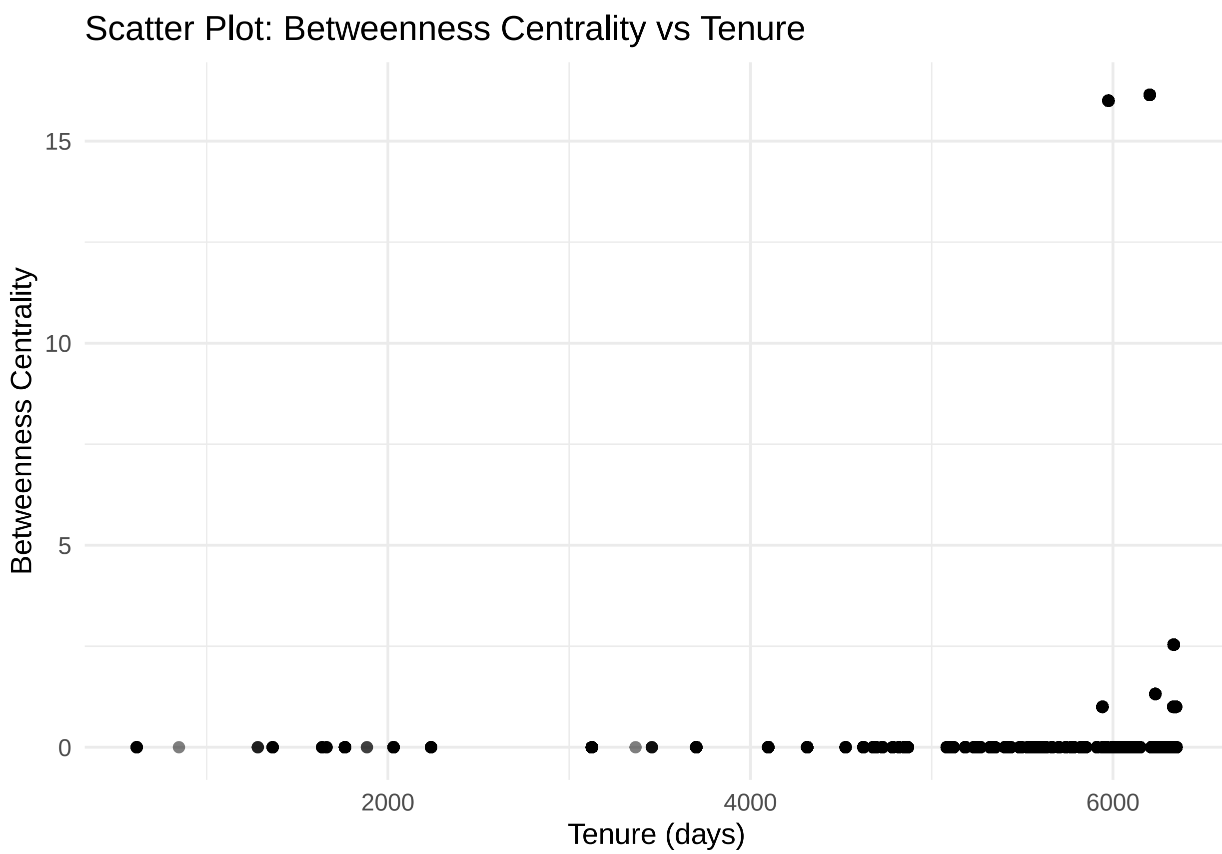
```
# Plot scatter plots to visualize the relationships
scatter_plot_degree_tenure <- ggplot(applications_centrality, aes(x = tenure_days, y = degree_centralit
  geom_point(alpha = 0.5) +
  labs(title = "Scatter Plot: Degree Centrality vs Tenure",
       x = "Tenure (days)",
       y = "Degree Centrality") +
  theme_minimal()

scatter_plot_betweenness_tenure <- ggplot(applications_centrality, aes(x = tenure_days, y = betweenness_
  geom_point(alpha = 0.5) +
  labs(title = "Scatter Plot: Betweenness Centrality vs Tenure",
       x = "Tenure (days)",
       y = "Betweenness Centrality") +
  theme_minimal()

# Display the scatter plots
print(scatter_plot_degree_tenure)
```

```
## Warning: Removed 2735 rows containing missing values (geom_point).
```




```
print(scatter_plot_betweenness_tenure)
```

```
## Warning: Removed 2735 rows containing missing values (geom_point).
```




Based on the correlation results, we can make the following conclusions:

1. Degree Centrality and Tenure Days: There is a weak positive correlation (0.0518) between degree centrality and tenure days. This suggests that examiners with longer tenure may have slightly more connections within the workgroup, possibly due to their longer presence and more interactions within the organization.

2. Betweenness Centrality and Tenure Days: There is an even weaker positive correlation (0.0387) between betweenness centrality and tenure days. This implies that examiners with longer tenure might be slightly more likely to lie on the shortest paths between other examiners, although the effect is not strong.

3. Degree Centrality and Betweenness Centrality: There is a moderate positive correlation (0.4518) between degree centrality and betweenness centrality. This indicates that examiners with a higher degree centrality (more direct connections) are more likely to have a higher betweenness centrality (lie on more shortest paths between other examiners). This is expected, as more connected individuals tend to have a higher chance of being on the shortest paths between others in the network.

Overall, the correlations between centrality measures and tenure days are weak, suggesting that the relationship between examiners' tenure and their centrality in the network is not strong. However, the positive correlation between degree centrality and betweenness centrality indicates that these two centrality measures are related, as expected.

## Race and gender

Next, we examine the relationships between centrality measures and race or gender. First, we convert race and gender to numerical values (dummy coding). Then, we calculate the correlation matrix and visualize the relationships using scatter plots.

```
# Convert race and gender to numerical values (dummy coding)
dummy_coded_data <- applications_centrality %>%
  mutate(
    race_num = as.numeric(factor(race, levels = unique(race))),
    gender_num = as.numeric(factor(gender, levels = unique(gender)))
  )

# Calculate the correlation matrix
correlation_matrix <- cor(dummy_coded_data[, c("degree_centrality", "betweenness_centrality", "race_num

# Print the correlation matrix
print(correlation_matrix)
```

```
##                        degree_centrality betweenness_centrality   race_num
## degree_centrality              1.0000000              0.4644826 0.11384053
## betweenness_centrality         0.4644826              1.0000000 0.22670001
## race_num                       0.1138405              0.2267000 1.00000000
## gender_num                     0.1216593              0.1261357 0.08928328
##                        gender_num
## degree_centrality      0.12165935
## betweenness_centrality 0.12613569
## race_num               0.08928328
## gender_num             1.00000000
```

Based on the correlation matrix output, the relationships between centrality measures (degree and betweenness) and gender or race can be characterized as follows:

1. Degree centrality and race: There is a weak positive correlation (0.11384053) between degree centrality and race. This suggests that as the race variable increases in value, degree centrality tends to increase slightly. However, the relationship is weak and might not be practically significant.

2. Degree centrality and gender: There is a weak positive correlation (0.12165935) between degree centrality and gender. This suggests that as the gender variable increases in value, degree centrality tends to increase slightly. Similar to the relationship between degree centrality and race, the relationship is weak and might not be practically significant.

3. Betweenness centrality and race: There is a weak positive correlation (0.22670001) between betweenness centrality and race. This suggests that as the race variable increases in value, betweenness centrality tends to increase slightly. The relationship is weak but slightly stronger compared to the relationships between degree centrality and race or gender.

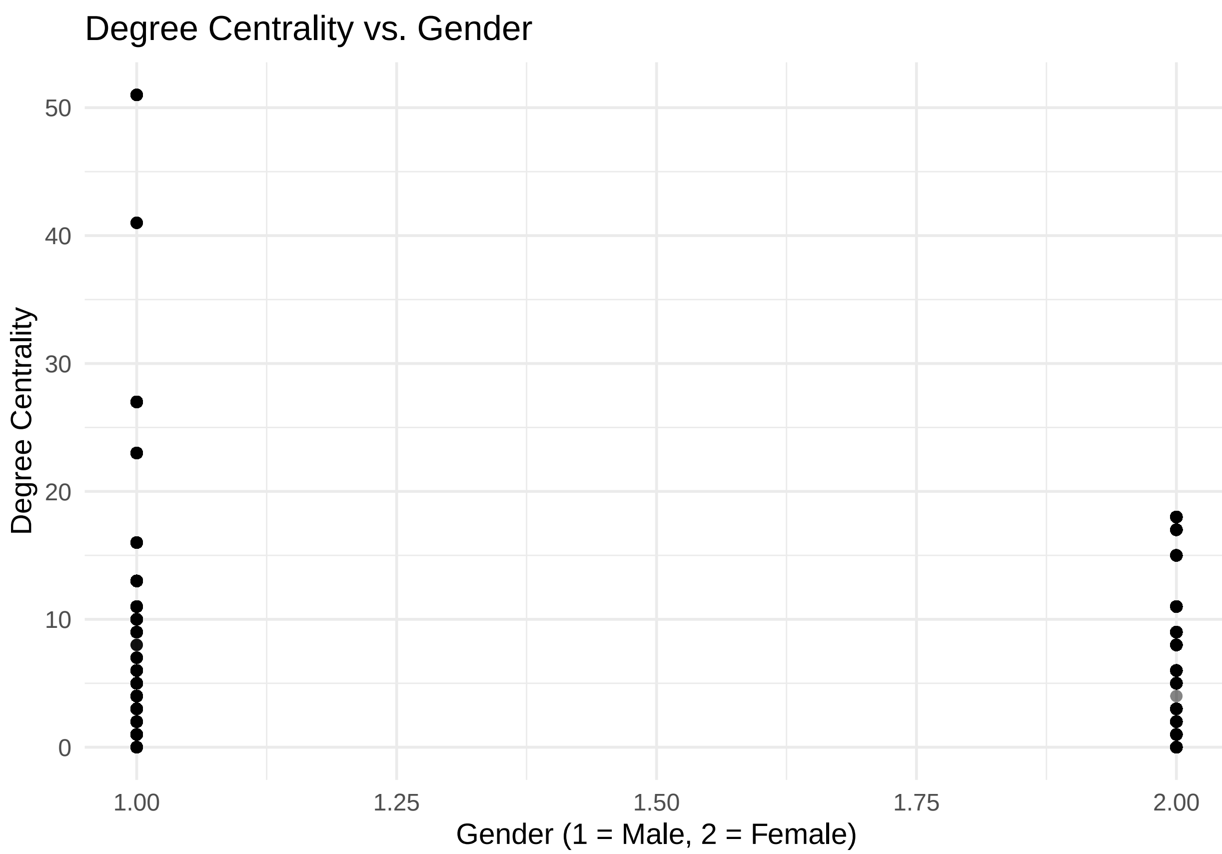
4. Betweenness centrality and gender: There is a weak positive correlation (0.12613569) between betweenness centrality and gender. This suggests that as the gender variable increases in value, betweenness centrality tends to increase slightly. The relationship is weak and might not be practically significant.

In summary, the relationships between centrality measures and gender or race are weak. This indicates that the centrality of examiners in the network might not be strongly influenced by their race or gender. However, it's essential to note that correlation does not imply causation, and other factors might be contributing to the observed relationships.

Let's visualize these relationships:

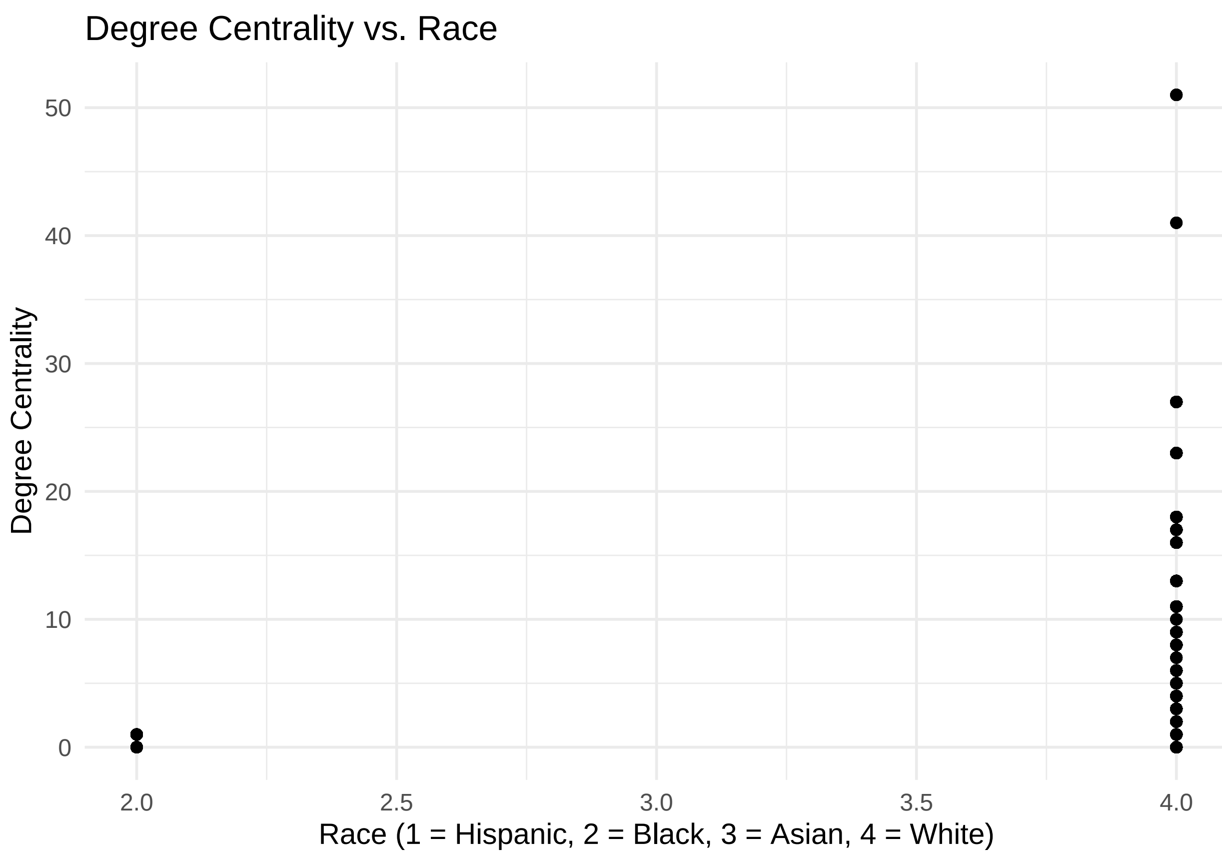
```
# Add 'gender_num' and 'race_num' columns to the dataframe
applications_centrality <- applications_centrality %>%
  mutate(
    gender_num = case_when(
      gender == "male" ~ 1,
      gender == "female" ~ 2,
      TRUE ~ NA_real_
    ),
    race_num = case_when(
      race == "hispanic" ~ 1,
      race == "black" ~ 2,
      race == "asian" ~ 3,
      race == "white" ~ 4,
      TRUE ~ NA_real_
    )
  )

# Scatter plot for Degree Centrality vs. Gender
degree_gender_plot <- applications_centrality %>%
  ggplot(aes(x = gender_num, y = degree_centrality)) +
  geom_point(alpha = 0.1) +
  labs(title = "Degree Centrality vs. Gender",
       x = "Gender (1 = Male, 2 = Female)",
       y = "Degree Centrality") +
  theme_minimal()

# Display plot
print(degree_gender_plot)
```

```
## Warning: Removed 16646 rows containing missing values (geom_point).
```



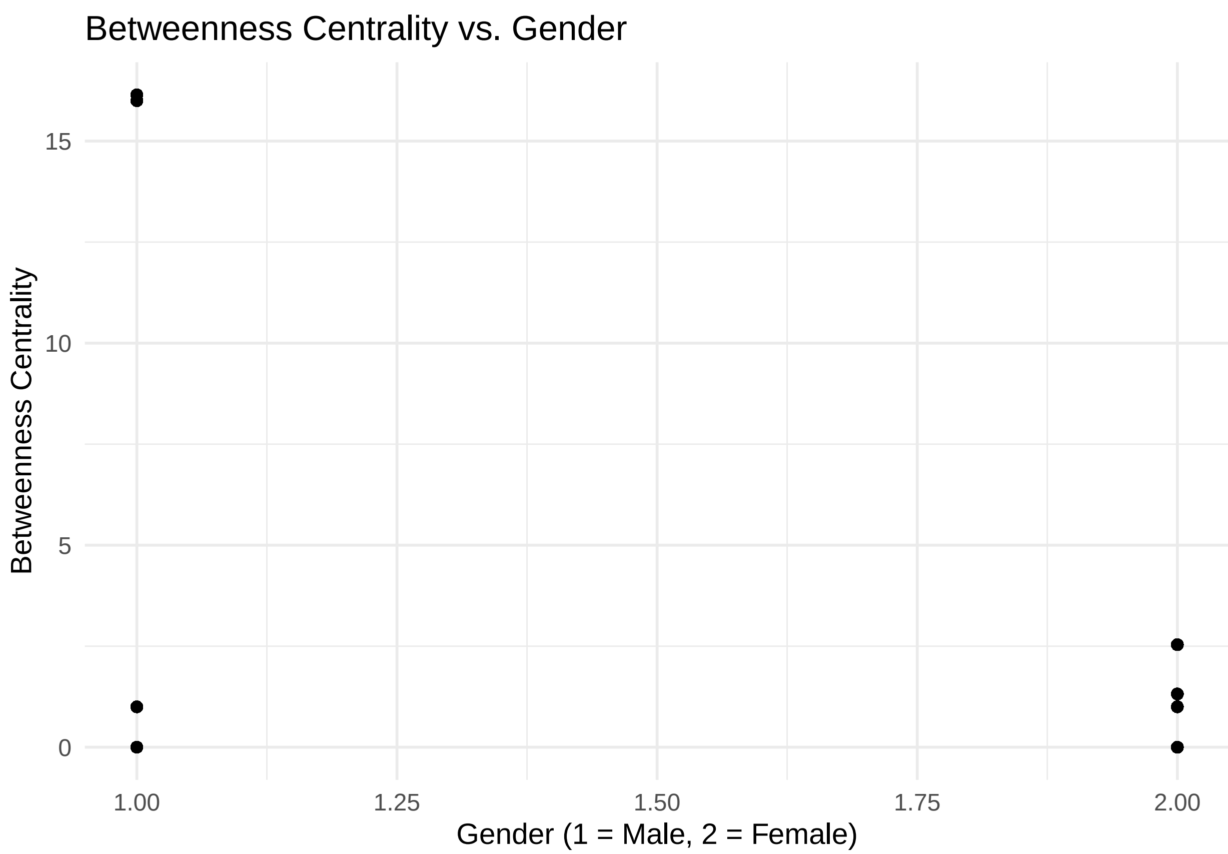

```r
# Scatter plot for Degree Centrality vs. Race
degree_race_plot <- applications_centrality %>%
  ggplot(aes(x = race_num, y = degree_centrality)) +
  geom_point(alpha = 0.1) +
  labs(title = "Degree Centrality vs. Race",
       x = "Race (1 = Hispanic, 2 = Black, 3 = Asian, 4 = White)",
       y = "Degree Centrality") +
  theme_minimal()

# Display plot
print(degree_race_plot)
```

```
## Warning: Removed 30541 rows containing missing values (geom_point).
```



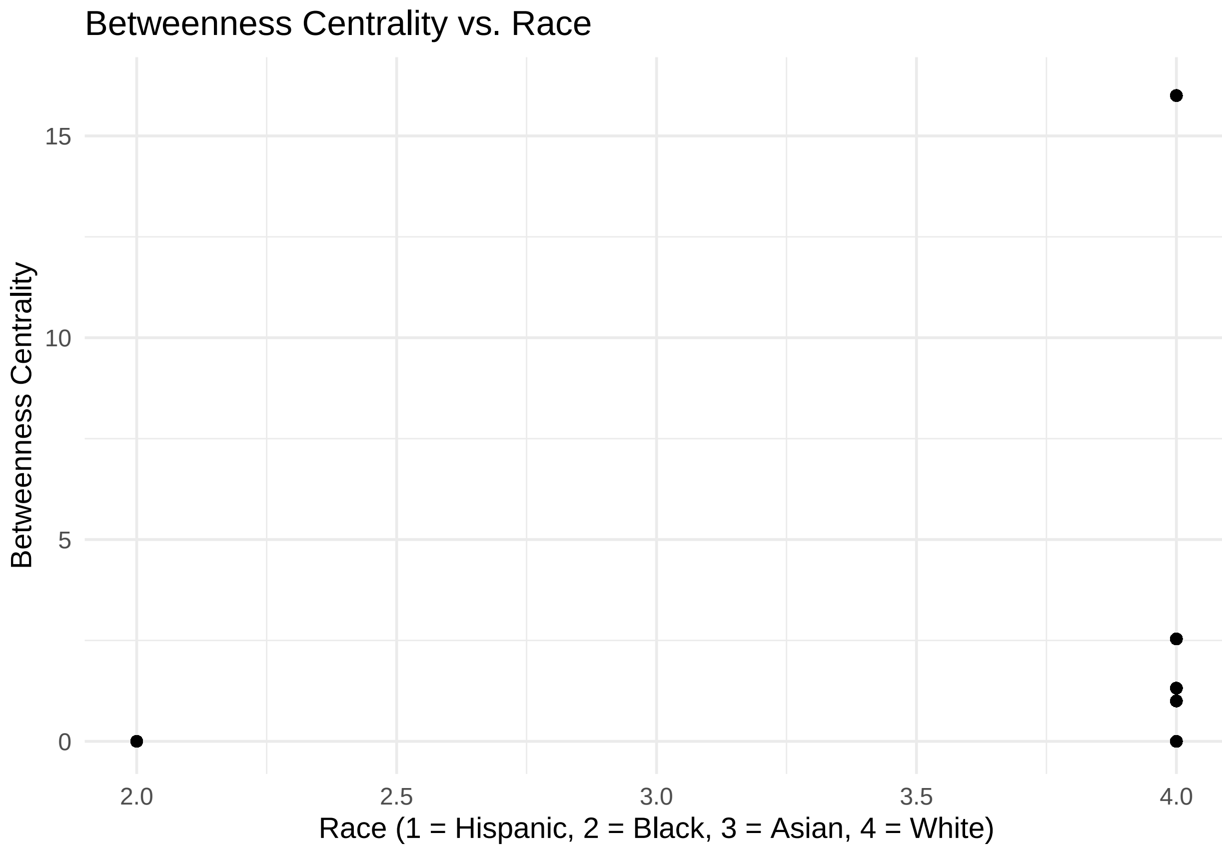

```r
# Scatter plot for Betweenness Centrality vs. Gender
betweenness_gender_plot <- applications_centrality %>%
  ggplot(aes(x = gender_num, y = betweenness_centrality)) +
  geom_point(alpha = 0.1) +
  labs(title = "Betweenness Centrality vs. Gender",
       x = "Gender (1 = Male, 2 = Female)",
       y = "Betweenness Centrality") +
  theme_minimal()

# Display plot
print(betweenness_gender_plot)
```

```
## Warning: Removed 16646 rows containing missing values (geom_point).
```




```r
# Scatter plot for Betweenness Centrality vs. Race
betweenness_race_plot <- applications_centrality %>%
  ggplot(aes(x = race_num, y = betweenness_centrality)) +
  geom_point(alpha = 0.1) +
  labs(title = "Betweenness Centrality vs. Race",
       x = "Race (1 = Hispanic, 2 = Black, 3 = Asian, 4 = White)",
       y = "Betweenness Centrality") +
  theme_minimal()

# Display plot
print(betweenness_race_plot)
```

```
## Warning: Removed 30541 rows containing missing values (geom_point).
```




In this analysis, we have successfully loaded the data, added demographic variables, selected two workgroups, compared their demographics, and created advice networks with centrality scores for the examiners. This information can be used to explore the relationships between examiners' demographics and their advice networks.