Exercise 4

Lucie Peccoux

2023-03-31

For this exercise I used a new parquet file called applications 2 that I made based on Exercise 3. Indeed, I created this parquet file with the original app_data_sample file to which I have added the gender, the race, the tenure and the work groups from Exercise 3. The code for how those features were added is available in Exercise 3.

First step is to import all the libraries needed and the files:

```
library(arrow)
```

```
## Warning: le package 'arrow' a été compilé avec la version R 4.2.2

##
## Attachement du package : 'arrow'

## L'objet suivant est masqué depuis 'package:utils':
##
##     timestamp
```

```
library(dplyr)
```

```
##
## Attachement du package : 'dplyr'

## Les objets suivants sont masqués depuis 'package:stats':
##
##     filter, lag

## Les objets suivants sont masqués depuis 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(tidygraph)
```

```
## Warning: le package 'tidygraph' a été compilé avec la version R 4.2.2

##
## Attachement du package : 'tidygraph'

## L'objet suivant est masqué depuis 'package:stats':
##
##     filter
```

```
library(igraph)
```

```
## Warning: le package 'igraph' a été compilé avec la version R 4.2.2

##
## Attachement du package : 'igraph'

## L'objet suivant est masqué depuis 'package:tidygraph':
##
##     groups

## Les objets suivants sont masqués depuis 'package:dplyr':
##
##     as_data_frame, groups, union

## Les objets suivants sont masqués depuis 'package:stats':
##
##     decompose, spectrum

## L'objet suivant est masqué depuis 'package:base':
##
##     union
```

```
library(visNetwork)
```

```
## Warning: le package 'visNetwork' a été compilé avec la version R 4.2.2
```

```
library(visNetwork)
library(tidyr)
```

```
##
## Attachement du package : 'tidyr'

## L'objet suivant est masqué depuis 'package:igraph':
##
##     crossing
```

```
library(readr)
```

```
## Warning: le package 'readr' a été compilé avec la version R 4.2.2
```

```
applications <- read_parquet("C:/Users/33652/Documents/Canada/Mcgill/COURS/Winter semester/Orgaziational
edges <- read_csv("C:/Users/33652/Documents/Canada/Mcgill/COURS/Winter semester/Orgaziational Network/ed
```

```
## 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.
```

Question 1 : Patent Processing time

The first question required to created the processing time features. I made the choice to create 2 new features : the processing time in days and the processing time in weeks. The processing time can be calculated by counting the days between the application filing date and the date a decision was made. For this last part, we have 2 features to look at : either the patent_issue_date if the patent was granted or the abandon_date if it was not.

```
#Making sure the dates are read as such by R
applications$filing_date <- as.Date(applications$filing_date)
applications$patent_issue_date <- as.Date(applications$patent_issue_date)
applications$abandon_date <- as.Date(applications$abandon_date)

#Processing time in days
applications$app_proc_time_days  <- ifelse(!is.na(applications$patent_issue_date),
                                       difftime(applications$patent_issue_date, applications$filing_dat
                                       difftime(applications$abandon_date, applications$filing_date, un
#Processing time in weeks
applications$app_proc_time_weeks <- ifelse(!is.na(applications$patent_issue_date),
                                        difftime(applications$patent_issue_date, applications$filing_da
                                        difftime(applications$abandon_date, applications$filing_date, u
```

Some dates might have been wrongly reported as some processing times appears to be negative. I decided to remove them from the dataset.

```
#Removing outliers
applications <- applications %>%
  filter(app_proc_time_days  >= 0)
```

Question 2: Use linear regression models `lm()` to estimate the relationship between centrality and `app_proc_time`

The first step to answer this question was to calculated the centralities for each examiner. To do so we need to have nodes and edges. I first took care of the edges. As I removed some outliers I made sure that only the examiners within the applications dataset without the outliers were also in the edges dataset.

```
#Edges dataset
edges_df<-edges
edges<-edges_df

#Filtering the data so that only the examiner_id still in the applications dataset stays in the eges da
edges<-edges%>%
  filter(ego_examiner_id %in% applications$examiner_id)%>%
  drop_na()%>%
  mutate(from=ego_examiner_id, to=alter_examiner_id)%>%
  select(from,to)
```

I then created a new dataset for the nodes with only the examiner_id from the application dataset.

```
nodes<-select(applications, examiner_id)
```

After that I used a graph to get the centrality scores with the degree function

```
graph <- graph_from_data_frame(edges, directed = TRUE)
#Degree centrality
deg <- degree(graph)
centrality_table <- data.frame(node = V(graph)$name, degree_centrality = deg)
centrality_table <- centrality_table[order(-centrality_table$degree_centrality),]

#Closeness centrality
closeness<-closeness(graph, mode='out')
closeness_table <- data.frame(node = V(graph)$name, closeness_centrality = closeness)
closeness_table <- closeness_table[order(-closeness_table$closeness_centrality),]

#Betweenness Centrality
betweenness <- betweenness(graph, directed = TRUE)
betweenness_table <- data.frame(node = V(graph)$name, betweenness_centrality = betweenness)
betweenness_table <- betweenness_table[order(-betweenness_table$betweenness_centrality),]
```

Next step was to add the centrality to the applications dataset

```
#Adding it back to the applications dataset
deg_centrality <- data.frame(examiner_id = V(graph)$name, degree_centrality = deg)
close_centrality <- data.frame(examiner_id = V(graph)$name, closeness_centrality = closeness)
between_centrality <- data.frame(examiner_id = V(graph)$name, betweenness_centrality = betweenness)

# Merge new_dataset with degree centrality, closeness centrality, and betweenness centrality data frame
applications2 <- merge(applications, deg_centrality, by = "examiner_id")
applications2 <- merge(applications2, close_centrality, by = "examiner_id")
applications2 <- merge(applications2, between_centrality, by = "examiner_id")
```

Let's run some linear regression.

```
model <- lm(app_proc_time_days ~ degree_centrality, data = applications2)
summary(model)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ degree_centrality, data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1498.2  -431.9  -113.9   296.1  4909.6
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       1.212e+03  7.051e-01 1718.61   <2e-16 ***
## degree_centrality 5.485e-01  2.044e-02   26.84   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 630.7 on 1052120 degrees of freedom
## Multiple R-squared:  0.000684,   Adjusted R-squared:  0.000683
## F-statistic: 720.1 on 1 and 1052120 DF,  p-value: < 2.2e-16
```

We can see that there is a significant postive relationship between the degree centrality and the application process time in days. With the intercept we can understand that when the degree centrality is 0, the estimated value of the application process time is 1212 days (which seems to be a lot). We can also understand that, on average, a one unit increase in degree centrality is associated with a 0.5485-unit increase in the process time.

```
model2 <- lm(app_proc_time_weeks ~ degree_centrality, data = applications2)
summary(model2)
```

```
##
## Call:
## lm(formula = app_proc_time_weeks ~ degree_centrality, data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -214.03  -61.70  -16.28   42.30  701.37
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)       173.11916    0.10073 1718.61   <2e-16 ***
## degree_centrality   0.07835    0.00292   26.84   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 90.1 on 1052120 degrees of freedom
## Multiple R-squared:  0.000684,   Adjusted R-squared:  0.000683
## F-statistic: 720.1 on 1 and 1052120 DF,  p-value: < 2.2e-16
```

When looking at the the process time in weeks not in days, we can see the same phenomenon. The process time is positively correlated with the degree centrality and for one unit increase in the degree centrality, the process time increase on average by 0.078 unit.

Let’s look at the betweenness cetrality and lead the same analysis.

```
model3 <- lm(app_proc_time_days ~ betweenness_centrality, data = applications2)
summary(model3)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ betweenness_centrality, data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1400.8  -430.9  -113.8   295.3  4906.2
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            1.219e+03  6.193e-01 1968.15   <2e-16 ***
## betweenness_centrality 4.244e-03  1.363e-04   31.14   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 630.6 on 1052120 degrees of freedom
## Multiple R-squared:  0.0009208,  Adjusted R-squared:  0.0009198
## F-statistic: 969.7 on 1 and 1052120 DF,  p-value: < 2.2e-16
```

```
model4 <- lm(app_proc_time_weeks ~ betweenness_centrality, data = applications2)
summary(model4)
```

```
##
## Call:
## lm(formula = app_proc_time_weeks ~ betweenness_centrality, data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -200.12  -61.56  -16.26   42.18  700.89
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            1.741e+02  8.847e-02 1968.15   <2e-16 ***
## betweenness_centrality 6.063e-04  1.947e-05   31.14   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 90.09 on 1052120 degrees of freedom
## Multiple R-squared:  0.0009208,  Adjusted R-squared:  0.0009198
## F-statistic: 969.7 on 1 and 1052120 DF,  p-value: < 2.2e-16
```

Here again we can witness a positive relationship between the betweenness centrality and the application process time. It seems like an increase of one unit in the betweeness centrality will most likely increase the application process time.

Let's look at other factor as well.

For this next try, I added the parameter Race, setting White as the reference for this categorical variable and checked the effect with degree centrality

```
#Using the race, setting the reference on White and comparing the results
applications2$race<-factor(applications2$race)
applications2$race<-relevel(applications2$race, ref='white')
attach(applications2)
model5 <- lm(app_proc_time_days ~ degree_centrality+race, data = applications2)
summary(model5)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ degree_centrality + race, data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1526.9  -431.2  -113.3   295.0  4934.0
```

```
##
## Coefficients:
##                    Estimate Std. Error  t value Pr(>|t|)
## (Intercept)       1.187e+03  8.543e-01 1389.983   <2e-16 ***
## degree_centrality 5.284e-01  2.043e-02   25.869   <2e-16 ***
## raceAsian         6.709e+01  1.338e+00   50.154   <2e-16 ***
## raceblack         2.825e+01  3.052e+00    9.254   <2e-16 ***
## raceHispanic      7.613e+01  4.323e+00   17.609   <2e-16 ***
## raceother         2.197e+02  1.544e+01   14.227   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 629.9 on 1052116 degrees of freedom
## Multiple R-squared:  0.003365,   Adjusted R-squared:  0.00336
## F-statistic: 710.5 on 5 and 1052116 DF,  p-value: < 2.2e-16
```

We can see with the output that all coefficient are significant as p-value <0.05 which lean that the differences in application processing time between the reference category (white) and the each of the other race categories are unlikely to be due to chance. We can notice that all the coefficient are positive which mean that in our case, examiner that are not white tend to take more time to process the application.

```
model6 <- lm(app_proc_time_days ~ betweenness_centrality+race, data = applications2)
summary(model6)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ betweenness_centrality + race,
##     data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1370.9  -431.1  -113.4   294.9  4927.9
##
## Coefficients:
##                         Estimate Std. Error  t value Pr(>|t|)
## (Intercept)            1.194e+03  7.875e-01 1516.290   <2e-16 ***
## betweenness_centrality 4.169e-03  1.361e-04   30.626   <2e-16 ***
## raceAsian              6.732e+01  1.337e+00   50.343   <2e-16 ***
## raceblack              2.720e+01  3.051e+00    8.915   <2e-16 ***
## raceHispanic           7.700e+01  4.323e+00   17.813   <2e-16 ***
## raceother              2.178e+02  1.544e+01   14.107   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 629.8 on 1052116 degrees of freedom
## Multiple R-squared:  0.003619,   Adjusted R-squared:  0.003615
## F-statistic: 764.4 on 5 and 1052116 DF,  p-value: < 2.2e-16
```

Interestingly the same thing happens with white as reference for the betweenness centrality.

Let's try it with a different reference. Let's make the race Asian the reference

```
#Asian as reference
applications2$race<-relevel(applications2$race, ref='Asian')
attach(applications2)
```

```
## Les objets suivants sont masqués depuis applications2 (pos = 3):
##
##     abandon_date, app_proc_time_days, app_proc_time_weeks,
##     appl_status_code, appl_status_date, application_number,
##     betweenness_centrality, closeness_centrality, degree_centrality,
##     disposal_type, earliest_date, examiner_art_unit, examiner_id,
##     examiner_name_first, examiner_name_last, examiner_name_middle,
##     filing_date, gender, latest_date, patent_issue_date, patent_number,
##     race, tc, tenure_days, uspc_class, uspc_subclass, workgroups
```

```
model7<- lm(app_proc_time_days ~ degree_centrality+race, data = applications2)
summary(model7)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ degree_centrality + race, data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1526.9  -431.2  -113.3   295.0  4934.0
##
## Coefficients:
##                     Estimate Std. Error  t value Pr(>|t|)
## (Intercept)       1254.53352    1.14517 1095.500   <2e-16 ***
## degree_centrality    0.52845    0.02043   25.869   <2e-16 ***
## racewhite          -67.09227    1.33772  -50.154   <2e-16 ***
## raceblack          -38.84629    3.14314  -12.359   <2e-16 ***
## raceHispanic         9.03689    4.38765    2.060   0.0394 *
## raceother          152.61978   15.46144    9.871   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 629.9 on 1052116 degrees of freedom
## Multiple R-squared:  0.003365,   Adjusted R-squared:  0.00336
## F-statistic: 710.5 on 5 and 1052116 DF,  p-value: < 2.2e-16
```

Here we can see that some coefficient are negative. Race as White as a negative coefficient which make sens compared to our previous findings. But we can also notice that race as balck also have a negative coefficient. This means that for the same degree centrality, a black examiner will tend to process the application faster.

```
model8 <- lm(app_proc_time_days ~ betweenness_centrality+race, data = applications2)
summary(model8)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ betweenness_centrality + race,
##     data = applications2)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1370.9  -431.1  -113.4   294.9  4927.9
##
## Coefficients:
##                          Estimate Std. Error  t value Pr(>|t|)
## (Intercept)             1.261e+03  1.087e+00 1160.768   <2e-16 ***
## betweenness_centrality  4.169e-03  1.361e-04   30.626   <2e-16 ***
## racewhite              -6.732e+01  1.337e+00  -50.343   <2e-16 ***
## raceblack              -4.012e+01  3.141e+00  -12.773   <2e-16 ***
## raceHispanic            9.682e+00  4.387e+00    2.207   0.0273 *
## raceother               1.505e+02  1.546e+01    9.736   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 629.8 on 1052116 degrees of freedom
## Multiple R-squared:  0.003619,   Adjusted R-squared:  0.003615
## F-statistic: 764.4 on 5 and 1052116 DF,  p-value: < 2.2e-16
```

Here again the same thing is happening for the betweenness centrality.

Let's check a last factor, the tenure days.

```
model11<- lm(app_proc_time_days ~ degree_centrality+tenure_days, data = applications2)
summary(model11)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ degree_centrality + tenure_days,
##     data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1503.4  -430.6  -114.3   295.7  4927.9
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        1.519e+03  5.335e+00  284.79   <2e-16 ***
## degree_centrality  5.479e-01  2.049e-02   26.74   <2e-16 ***
## tenure_days       -5.204e-02  8.951e-04  -58.13   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 629.5 on 1044754 degrees of freedom
##   (7365 observations effacées parce que manquantes)
## Multiple R-squared:  0.003942,   Adjusted R-squared:  0.00394
## F-statistic:  2067 on 2 and 1044754 DF,  p-value: < 2.2e-16
```

In this first linear regression with the degree centrality and the tenure days we can notice for a fixed degree centrality, for a one unit increase in tenure days, the application processing time decrease by 0.05 units. The p-value is also very small which lead us to think that this result is unlikely due to chance.

Let's check for the betweenness centrality

```
model12<- lm(app_proc_time_days ~ betweenness_centrality+tenure_days, data = applications2)
summary(model12)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ betweenness_centrality + tenure_days,
##     data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1401.4  -430.6  -113.9   295.2  4929.1
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             1.533e+03  5.318e+00  288.27   <2e-16 ***
## betweenness_centrality  4.490e-03  1.361e-04   32.99   <2e-16 ***
## tenure_days            -5.325e-02  8.953e-04  -59.48   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 629.4 on 1044754 degrees of freedom
##   (7365 observations effacées parce que manquantes)
## Multiple R-squared:  0.004298,   Adjusted R-squared:  0.004296
## F-statistic:  2255 on 2 and 1044754 DF,  p-value: < 2.2e-16
```

The same is happening with almost the same values.

Finally let’s see the degree centrality with the workgroups

```
model13<- lm(app_proc_time_days ~ betweenness_centrality+workgroups, data = applications2)
summary(model13)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ betweenness_centrality + workgroups,
##     data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1551.3  -405.6   -95.8   282.7  4913.0
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             1.264e+03  5.859e+01  21.566  < 2e-16 ***
## betweenness_centrality  2.428e-03  1.329e-04  18.273  < 2e-16 ***
## workgroups161          -2.765e+01  5.867e+01  -0.471 0.637493
## workgroups162          -2.660e+02  5.866e+01  -4.534 5.80e-06 ***
## workgroups163          -3.915e+01  5.867e+01  -0.667 0.504644
## workgroups164          -6.898e+01  5.865e+01  -1.176 0.239543
## workgroups165          -1.437e+02  5.868e+01  -2.449 0.014311 *
## workgroups166          -2.180e+02  5.996e+01  -3.636 0.000277 ***
## workgroups167          -2.113e+02  5.891e+01  -3.588 0.000334 ***
## workgroups170          -2.426e+02  1.515e+02  -1.601 0.109311
## workgroups171          -1.370e+02  5.867e+01  -2.336 0.019496 *  
## workgroups172          -2.209e+02  5.867e+01  -3.766 0.000166 ***
## workgroups173          -2.264e+02  5.867e+01  -3.859 0.000114 ***
## workgroups174          -1.877e+02  5.867e+01  -3.199 0.001380 ** 
## workgroups175          -3.037e+02  5.870e+01  -5.174 2.29e-07 ***
## workgroups176          -2.613e+02  5.866e+01  -4.455 8.39e-06 ***
## workgroups177          -1.889e+02  5.867e+01  -3.219 0.001285 ** 
## workgroups178           4.409e+01  5.872e+01   0.751 0.452761    
## workgroups179          -3.025e+00  5.863e+01  -0.052 0.958851    
## workgroups210           2.548e+02  1.138e+02   2.240 0.025076 *  
## workgroups211          -1.320e+02  5.867e+01  -2.250 0.024467 *  
## workgroups212          -1.581e+01  5.870e+01  -0.269 0.787688    
## workgroups213           9.671e+01  5.879e+01   1.645 0.099976 .  
## workgroups214           4.120e+02  5.897e+01   6.987 2.82e-12 ***
## workgroups215           1.471e+02  5.873e+01   2.505 0.012234 *  
## workgroups216           1.232e+02  5.868e+01   2.099 0.035784 *  
## workgroups217           2.856e+02  5.874e+01   4.862 1.16e-06 ***
## workgroups218          -5.165e+01  5.868e+01  -0.880 0.378767    
## workgroups219           2.410e+02  5.872e+01   4.104 4.06e-05 ***
## workgroups240          -1.404e+02  9.753e+01  -1.439 0.150023    
## workgroups241           1.379e+02  5.891e+01   2.341 0.019253 *  
## workgroups242           2.192e+02  5.877e+01   3.731 0.000191 ***
## workgroups243           1.717e+02  5.872e+01   2.924 0.003450 ** 
## workgroups244           2.314e+02  5.873e+01   3.941 8.13e-05 ***
## workgroups245           2.204e+02  5.870e+01   3.755 0.000173 ***
## workgroups246           4.675e+01  5.871e+01   0.796 0.425878    
## workgroups247          -1.062e+01  5.870e+01  -0.181 0.856411    
## workgroups248           9.762e+01  5.887e+01   1.658 0.097228 .  
## workgroups249           7.597e+01  5.901e+01   1.287 0.197929    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 608.9 on 1052083 degrees of freedom
## Multiple R-squared:  0.06854,    Adjusted R-squared:  0.06851 
## F-statistic:  2037 on 38 and 1052083 DF,  p-value: < 2.2e-16
```

The work group of reference is the workgroup 160. What we can see this workgroup tends to be slower at processing application than other groups. Indeed, we can see a lot a negative value from the coefficient from other groups. Still some coefficient are positive which it's not the slowest group.

Question 3: Does this relationship differ by examiner gender?

Now let's check the relationship with centrality and the gender.

```
#Gender
model14 <- lm(app_proc_time_days ~ degree_centrality + gender + degree_centrality*gender, data = applica
summary(model14)
```

```
## 
## Call:
## lm(formula = app_proc_time_days ~ degree_centrality + gender + 
##     degree_centrality * gender, data = applications2)
## 
## Residuals:
```

```
##     Min      1Q  Median      3Q     Max
## -1370.1  -429.0  -113.8   292.0  4934.1
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 1187.06904    1.38415 857.616   <2e-16 ***
## degree_centrality              0.86209    0.04391  19.634   <2e-16 ***
## gendermale                    25.53078    1.64903  15.482   <2e-16 ***
## degree_centrality:gendermale  -0.49806    0.05015  -9.932   <2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 627.4 on 905310 degrees of freedom
##   (146808 observations effacées parce que manquantes)
## Multiple R-squared:  0.0008482,  Adjusted R-squared:  0.0008449
## F-statistic: 256.2 on 3 and 905310 DF,  p-value: < 2.2e-16
```

The results of this output are interesting. With have an intercept of 1187, which means that for a female examiner for degree centrality of 0, the application process time 1187 days. Holding the gender constant (as female), for one unit increase in the degree centrality the application process time increases by 0.86. We have a coefficient of 25 for the gendermale. This means that a for a degree centrality of 0, a male exmainer will take on average 25 more days to process an application. The coefficient degree_centrality:gendermale indicates that the effects if the degree centrality on the application process time is weaker on the male examiner than on the female. Which mean the degree centrality affect male less.

Looking at the betweenness centrality now:

```
model15 <- lm(app_proc_time_days ~ betweenness_centrality + gender + betweenness_centrality*gender, data
summary(model15)
```

```
##
## Call:
## lm(formula = app_proc_time_days ~ betweenness_centrality + gender +
##     betweenness_centrality * gender, data = applications2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1441.7  -428.1  -113.5   291.5  4921.6
##
## Coefficients:
##                                     Estimate Std. Error  t value Pr(>|t|)
## (Intercept)                        1.200e+03  1.196e+00 1004.075  < 2e-16 ***
## betweenness_centrality             1.584e-03  3.554e-04    4.456 8.37e-06 ***
## gendermale                         1.509e+01  1.438e+00   10.494  < 2e-16 ***
## betweenness_centrality:gendermale 3.298e-03  3.854e-04    8.556  < 2e-16 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 627.2 on 905310 degrees of freedom
##   (146808 observations effacées parce que manquantes)
## Multiple R-squared:  0.001377,   Adjusted R-squared:  0.001374
## F-statistic: 416.2 on 3 and 905310 DF,  p-value: < 2.2e-16
```

The results show that for a female with a betweenness centrality of 0, it takes around 1200 days to process an application. For every one unit increase in betweenness centrality, this processinf time increase by 0.0016.

For a betweennness degree of 0, a male examiner will take around 1215 days to process an application. In this case the coefficient for the betweenness_centrality:gendermale is positive. This means that the impact of the betweenness centrality is more significant on the male than it is on the female examiners.

Question 4:

Overall what we have learned is the following : Independently of the other feature studied, it seems like a higher centrality degree leads to a longer processing time. If we think about the centrality degree as someone seeking advice or some sought for advice we can make the link that maybe those people are loosing on efficiency while spending time helping other or looking for information to solve their problem. The same thing goes for the betweenness centrality and the explanation can be pretty similar. We've seen that white examiners for the same degree and betweenness centrality seems to be faster at processing application, followed by black examiners and then Asian examiners.

There was also a difference between male and female examiner. Overall female examiners tend to be faster at processing application, nevertheless the impact of the centrality is different. Indeed, the degree centrality has a less significant impact on men than on women, wehereas the betweenness centrality has a more significant impact on men than it has on women.