ORGB-672: Organization Network Analysis

Final Report

Submitted to:

Dr. Roman Galperin

Submitted by:

Emery Dittmer

Hai Nguyen

Ximena Rodriguez Miranda

Will Stephenson

April 18th, 2023

Table of Contents

1. Introduction 2

2. Background 2

3. Methodology 3

4. Analysis 4

5. Conclusion 6

6. Bibliography 7

7. Appendix 7

A. Linear Models 7

B. Survivor Analysis 7

C. Tree plot details and Analysis 8

D. Cluster Analysis 8

E. Correlation Plots 10

1. Introduction

US Patent Office

In this research paper we analyze the social connections of the United States Patent Office (USPTO). The USPTO plays a vital role in the American intellectual property system. They provide advice to the U.S. government, at levels all the way up to the president. This advice is led predominantly by the patent examiners who are on the ground floor for the patent review process. This process is not always fully independent however, formal, and informal relationships and advice can play a part within this review process.

In this paper we aim to understand how these networks can affect the patent review process, and more specifically, do patent advisors with better connected formal advise networks have a noticeable difference in their patent processing time?

Our hypothesis is that the more connected the advisor, the speedier the patent review will occur. This is the belief that networks can aid with more accurate decisions, in a phenomenon known as the wisdom of the crowds. While accuracy and speed are not always correlated, there is other evidence in the literature about the benefits of network centrality.

2. Background

Before moving any forward, it is important to clarify what centrality measures we are analyzing and what these measures of centrality mean. There are three types of centralities that we analyze, degree centrality, closeness centrality, and betweenness centrality. To understand these concepts let us look at a simple network analysis.

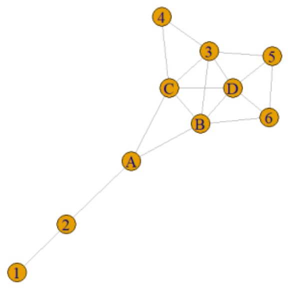


Figure 2.1 Example of network

In the above network we can see 10 nodes that have varying degrees of interconnectedness. For the network connections we will first turn to degree centrality. Degree centrality is the simplest of the three measures as it is just the number of edges coming out of each node[1]. Looking at our above example the score for A is 3. It is connected to nodes 2, B, and C. The highest scores are a tie of B, C, and D.

Our second score is closeness centrality. We calculate closeness based on the average distance from a node to all other nodes[2]. What this means is A is a distance of 2 from node D, but a distance of 3 from

[1] Newman 2018

[2] Newman 2018

node 5. Then we sum up all these calculations and average it by the total connects within the network. This would give A a closeness score of .0625. The highest score on this network is a tie between B and C.

Finally, we look at the betweenness scores. Betweenness score does not look solely at the node in play but rather looks at what node does the best job of filling in a gap[3]. Looking at the above, A has the highest betweenness score at 14. Looking at the graph, we can see that node A is the only way of linking nodes 1 and 2 to the group so it scores the highest here.

These measures of centrality allow a mathematical approach to informal advice networks within the USPTO. Using these measures we will evaluate the strength and interconnectedness of patent examiners and what role that may play in the speed of application decisions to be issued or abandoned.

3. Methodology

We will use two datasets to examine the role networks play in application speed. The first dataset, *applications*, contains a record of all application transactions (issued, abandoned or otherwise) with a time stamp and associated patent examiner ID. The second dataset, edges, contains a list of all advice sought between examiners regarding applications. The dataset matches two examiner ids, a date and patent application id.

For the analysis we first investigated the datasets to determine the general parameters such as data types, min-max and missing values. The investigation found 2 data issues that we needed to address. The *edges* data goes from 1-jan 2008 to 31 dec 2008 and the *application* data has a right (or future) filtering problem where decisions after 2016 drop off exponentially. Therefore, we filtered the dataset to contain data from 2008 to 2016 before removing all null values for the application status date.

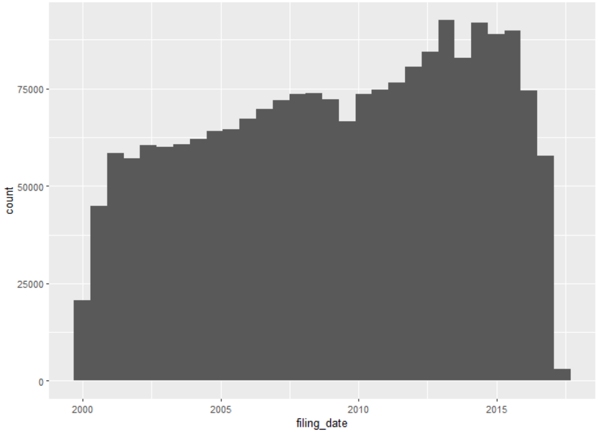


Figure 3.1 Histogram of patent applications post data cleaning

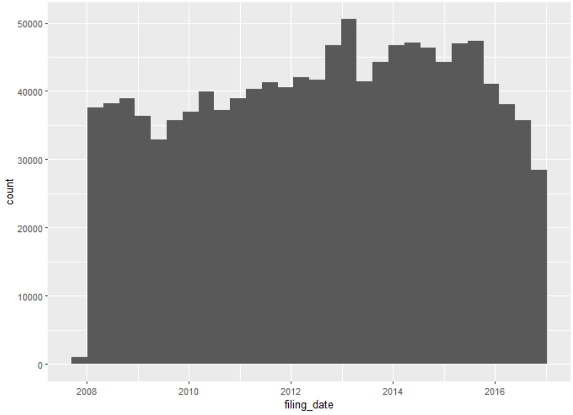


Figure 3.2 Histogram of Filing date for patent applications pre-data cleaning

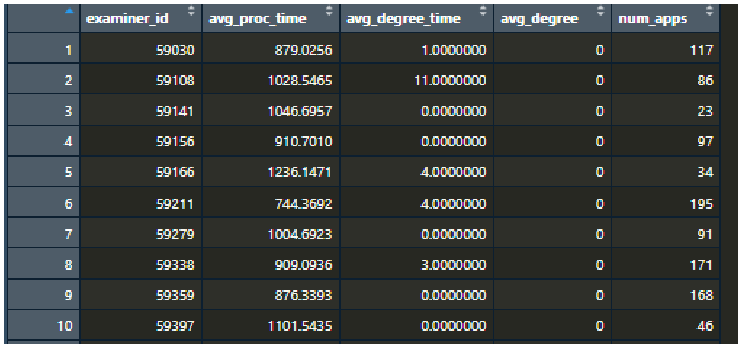
After data filtering, we added the gender and race of the examiners using the wru and gender packages. Then, we added the centrality, closeness and betweenness scores as of December 31, 2008, to the dataset. While not complete data these serve as indicators of advice networks.

[3] Newman 2018

To attempt better to capture the time series nature of the data, we added the centrality score as a measure through time to the data. The degree over time measure matched the data advice data to the application data vase on the status update date and a running total of all advice sought prior to that date. Here we assume that the cumulative sum of the centrality is an indicator of the centrality of an examiner within the network. This dataset is designated as the applications_tc.

From the application_tc data we also created a data subset called applications_ts_exam where the average application processing time per examiner is measured as well as their average application processing time and degree is measured. The data is filtered from 2008 to 2013 (inclusive) as the majority of apps are done after 1500 days (5 years) it is assumed that the application they receive is complete. This dataset should best represent the association of application time and degree centrality.

Based on the data available we considered investigating on an application basis as well as the examiner basis. Therefore applications_request_counts is a dataset that counts the application centrality degree. The edges data contains application numbers; therefore each application was given a degree based on the unique combination of alter and ego examiners, which were grouped by application and counted as the application degree. The number of appearances of applications produces a centrality score per application independent of examiner. This dataset will reveal if an application is speed is proportional to number of advice requests, independent of examiner.

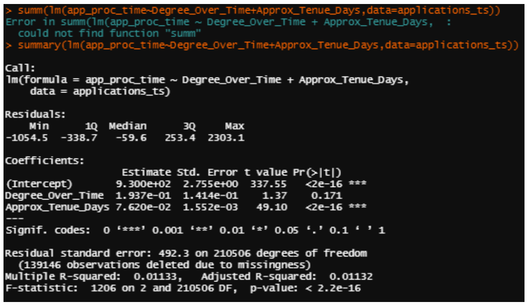


	examiner_id	avg_proc_time	avg_degree_time	avg_degree	num_apps
1	59030	879.0256	1.0000000	0	117
2	59108	1028.5465	11.0000000	0	86
3	59141	1046.6957	0.0000000	0	23
4	59156	910.7010	0.0000000	0	97
5	59166	1236.1471	4.0000000	0	34
6	59211	744.3692	4.0000000	0	195
7	59279	1004.6923	0.0000000	0	91
8	59338	909.0936	3.0000000	0	171
9	59359	876.3393	0.0000000	0	168
10	59397	1101.5435	0.0000000	0	46

Figure 3.3 application_ts_exam dataset

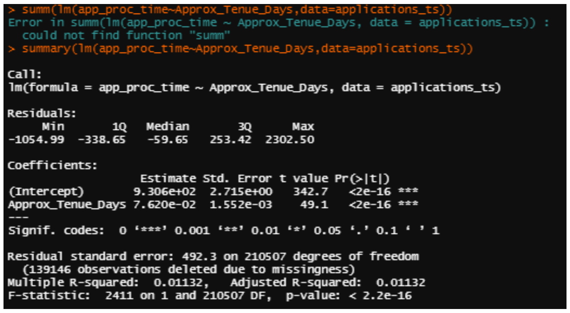
4. Analysis

To better investigate the relationship between processing time and centrality we used a series of linear regressions to compare the effects on processing time based on independent variables. We attempted several correlations only 2 proved insightful, the remaining are available in appendix A. We found that degree over time does not have a significant impact on application processing time, but tenure over time does. Based on figure 4.1 the approximate tenure days, or number of tenure days from the examiner's first decision until the current decisions, significantly influence application processing time. So based on this, we followed up with a second linear regression solely on tenure. As their tenure increases so does the application processing time. Although at first counterintuitive, it may be reasonable to assume that as the examiner is more senior, they receive more difficult applications and therefore require more time to complete the examination. The number of connections an examiner has is not linearly related to their application processing time.



```
> summ(lm(app_proc_time~Degree_Over_Time+Approx_Tenue_Days,data=applications_ts))
Error in summ(lm(app_proc_time ~ Degree_Over_Time + Approx_Tenue_Days,  :
  could not find function "summ"
> summary(lm(app_proc_time~Degree_Over_Time+Approx_Tenue_Days,data=applications_ts))

Call:
lm(formula = app_proc_time ~ Degree_Over_Time + Approx_Tenue_Days,
    data = applications_ts)

Residuals:
    Min      1Q  Median      3Q     Max
-1054.5  -338.7   -59.6   253.4  2303.1

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       9.300e+02  2.755e+00  337.55   <2e-16 ***
Degree_Over_Time  1.937e-01  1.414e-01    1.37    0.171
Approx_Tenue_Days 7.620e-02  1.552e-03   49.10   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 492.3 on 210506 degrees of freedom
  (139146 observations deleted due to missingness)
Multiple R-squared:  0.01133,   Adjusted R-squared:  0.01132
F-statistic:  1206 on 2 and 210506 DF,  p-value: < 2.2e-16
```

Figure 4.1 Relationship between degree over time, tenure, and application processing time

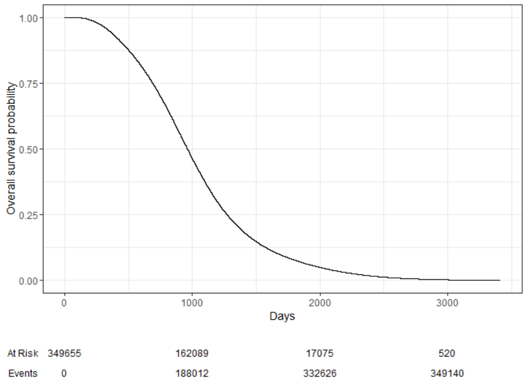


```
> summ(lm(app_proc_time~Approx_Tenue_Days,data=applications_ts))
Error in summ(lm(app_proc_time ~ Approx_Tenue_Days, data = applications_ts)) :
  could not find function "summ"
> summary(lm(app_proc_time~Approx_Tenue_Days,data=applications_ts))

Call:
lm(formula = app_proc_time ~ Approx_Tenue_Days, data = applications_ts)

Residuals:
     Min       1Q   Median       3Q      Max
-1054.99  -338.65   -59.65   253.42  2302.50

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)       9.306e+02  2.715e+00   342.7   <2e-16 ***
Approx_Tenue_Days 7.620e-02  1.552e-03    49.1   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 492.3 on 210507 degrees of freedom
  (139146 observations deleted due to missingness)
Multiple R-squared:  0.01132,   Adjusted R-squared:  0.01132
F-statistic:  2411 on 1 and 210507 DF,  p-value: < 2.2e-16
```

Figure 4.2 Relationship between tenure and application processing time

However, the application data only contains the name of one examiner for an application. There may be hidden relationships where the examiner signing off on the application is not the same as doing the bulk of the processing. We investigated whether or there was any association between the application processing time and the application centrality. While it may be reasonable to measure the correlation between application degree and application processing time, the linear model did not find a significance between the application centrality and processing time. The details of these findings are in appendix A.

Since the relationships are not linear, we look at the overall survival probability for the applications to understand if there is any visible or significant departure for application processing time based on the connection degree. Figures 4.3 and 4.4 indicate the number of days spent on the applications and the probability that they finish before a certain number of days. The survival graph in figure 4.4 seems to indicate that there is a difference in application time based on degree connections. There seems to be a slight benefit given to an application processing time with increasing degree connections. However, the relationship is very unclear as there are overlapping lines and there is a distribution of processing times for centralities that are proximal. 2 groups can be distinguished based on this data, the group with lower centrality (below 24 – green and blue) and above. The lower centrality has a consistent application processing time. This might be as application complexity increases, the number of requests for advice and help increases as well, presumably a higher number of requests for help would delay the application for longer than without the request for help.



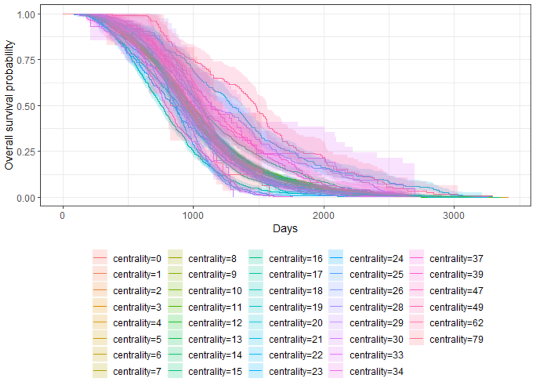


Figure 4.3: Survival plot with all data

Figure 4.4: Survivor plot of application_tc data. centrality score effect on survival (centrality is max centrality as seen from edges data)

Lastly, we applied a decision tree method on the data to understand hidden relationships between tenure days, the technology center (tc) and the connection degree over time. A decision tree with a control parameter of 0.001 found 3 primary factors determining application processing time. tenure, tc and centrality degree. A lower tenure was an indication of lower application time similarly some tcs had lower application time than others. Both may be indications of complexity. Some tcs may have less complex applications, while higher tenure staff may receive more difficult cases which require additional time . The degree over time impacts the application processing time. Connection degree between 24 and 9 seem to have an application processing time of 1139 days whereas under 9 has an average application processing time of 1024 days. While 100 days is a significant amount of time for application processing time this factor only effects specific tcs. The factor is far down the decision tree dendrogram indicating It is not as relevant as other factors. 26% of overall applications that are affected by the degree centrality (above 24 or below 9) which is a significant finding, but it is not generalizable. Therefore, while there may be an indication that there is some effect of degree on application time, other factors like tc or tenure dates are more prevalent factors to determine application processing time.

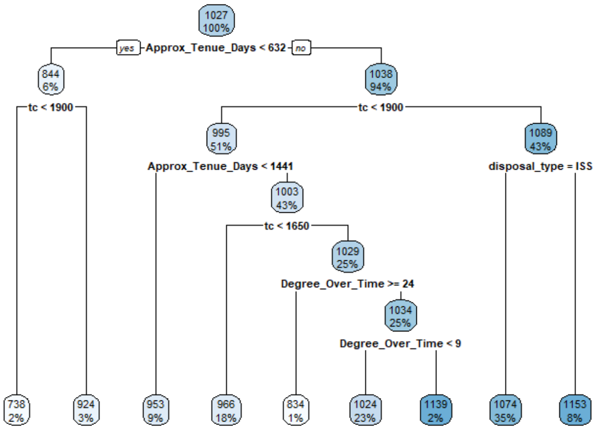


Figure 4.5: applications_tc decision tree for impact on application processing time

5. Conclusion

Looking at the analysis, and the data that we have available, we fail to reject our null hypothesis. We can find little evidence that our examiners' connectivity impacts the time taken in reviewing a patent application. From the data and evidence gathered and analyzed here, we fail to reject our null hypothesis.

Note that we are looking at one year of formalized advice interactions. This data may not be fully representative of all informal advice and relationships. Similarly, as examiners have been there for longer than simply the year 2008, we are unable to see how large people's existing networks were. Everyone was starting from 0 on January 1st. However this could still be beneficial for setting the ground work for future study. As there was a slight difference with the speed and connectivity in 2008, it would be interesting to see what the effect was over a longer period of time. Further if we were able to follow reviewers throughout their professional career we may see more benefit from the network. If someone established a well-connected formalized network in the years leading up to 2008, we do not have visibility on that. As such we may think that they are only connected to 1 person when really they are connected to almost all but know the best person to go to for specific questions.

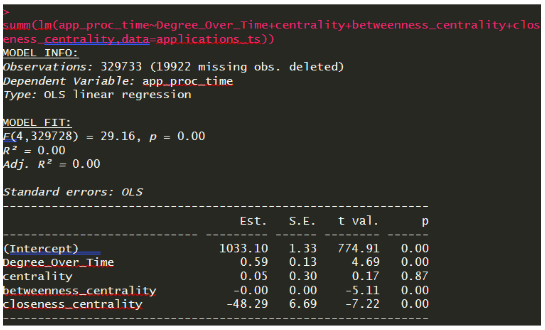
6. Bibliography

Newman, Mark. *Networks*. Vol. 1. Oxford University Press, 2018. https://doi.org/10.1093/oso/9780198805090.001.0001.

7. Appendix

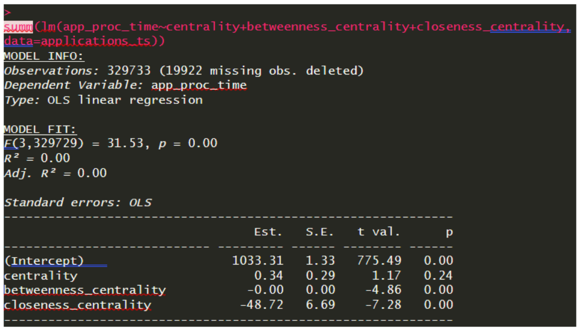
A. Linear Models

In this appendix we investigate the linear relationships.



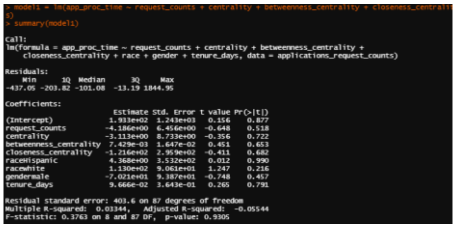
```
>
summ(lm(app_proc_time~Degree_Over_Time+centrality+betweenness_centrality+clos
eness_centrality,data=applications_ts))
MODEL INFO:
Observations: 329733 (19922 missing obs. deleted)
Dependent Variable: app_proc_time
Type: OLS linear regression

MODEL FIT:
F(4,329728) = 29.16, p = 0.00
R² = 0.00
Adj. R² = 0.00

Standard errors: OLS
-------------------------------------------------------------
                                 Est.    S.E.    t val.      p
---------------------------- --------- ------ -------- ------
(Intercept)                   1033.10    1.33    774.91   0.00
Degree_Over_Time                 0.59    0.13      4.69   0.00
centrality                       0.05    0.30      0.17   0.87
betweenness_centrality          -0.00    0.00     -5.11   0.00
closeness_centrality           -48.29    6.69     -7.22   0.00
-------------------------------------------------------------
```

Figure A.1: linear relationship between centrality scores over time and maximum observed centrality.



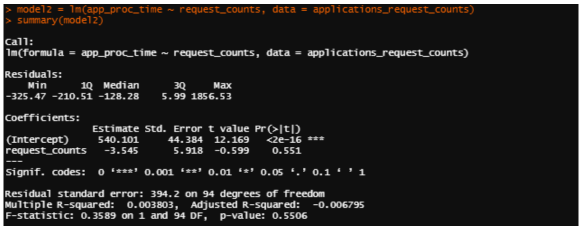
```
>
summ(lm(app_proc_time~centrality+betweenness_centrality+closeness_centrality,
data=applications_ts))
MODEL INFO:
Observations: 329733 (19922 missing obs. deleted)
Dependent Variable: app_proc_time
Type: OLS linear regression

MODEL FIT:
F(3,329729) = 31.53, p = 0.00
R² = 0.00
Adj. R² = 0.00

Standard errors: OLS
-------------------------------------------------------------
                                 Est.    S.E.    t val.      p
---------------------------- --------- ------ -------- ------
(Intercept)                   1033.31    1.33    775.49   0.00
centrality                       0.34    0.29      1.17   0.24
betweenness_centrality          -0.00    0.00     -4.86   0.00
closeness_centrality           -48.72    6.69     -7.28   0.00
-------------------------------------------------------------
```

Figure A.2: linear relationship between max centrality score observed in dataset.



```
> model1 = lm(app_proc_time ~ request_counts + centrality + betweenness_centrality + closeness_centrali
s)
> summary(model1)

Call:
lm(formula = app_proc_time ~ request_counts + centrality + betweenness_centrality +
    closeness_centrality + race + gender + tenure_days, data = applications_request_counts)

Residuals:
    Min      1Q  Median      3Q     Max
-437.05 -203.82 -101.08  -13.19 1844.95

Coefficients:
                         Estimate Std. Error t value Pr(>|t|)
(Intercept)             1.933e+02  1.243e+03   0.156    0.877
request_counts         -4.186e+00  6.456e+00  -0.648    0.518
centrality             -3.113e+00  8.733e+00  -0.356    0.722
betweenness_centrality  7.429e-03  1.647e-02   0.451    0.653
closeness_centrality   -1.216e+02  2.959e+02  -0.411    0.682
raceHispanic            4.368e+00  3.532e+02   0.012    0.990
racewhite               1.130e+02  9.061e+01   1.247    0.216
gendermale             -7.021e+01  9.387e+01  -0.748    0.457
tenure_days             9.666e-02  3.643e-01   0.265    0.791

Residual standard error: 403.6 on 87 degrees of freedom
Multiple R-squared:  0.03344,   Adjusted R-squared:  -0.05544
F-statistic: 0.3763 on 8 and 87 DF,  p-value: 0.9305
```

Figure A.3: Relationship between degree over time, tenure, and application processing time based on edges dataset



```
> model2 = lm(app_proc_time ~ request_counts, data = applications_request_counts)
> summary(model2)

Call:
lm(formula = app_proc_time ~ request_counts, data = applications_request_counts)

Residuals:
    Min      1Q  Median      3Q     Max
-325.47 -210.51 -128.28    5.99 1856.53

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)     540.101     44.384  12.169   <2e-16 ***
request_counts   -3.545      5.918  -0.599    0.551
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 394.2 on 94 degrees of freedom
Multiple R-squared:  0.003803,  Adjusted R-squared:  -0.006795
F-statistic: 0.3589 on 1 and 94 DF,  p-value: 0.5506
```

Figure A.4 Relationship between tenure and application processing time based on edges dataset

B. Survivor Analysis

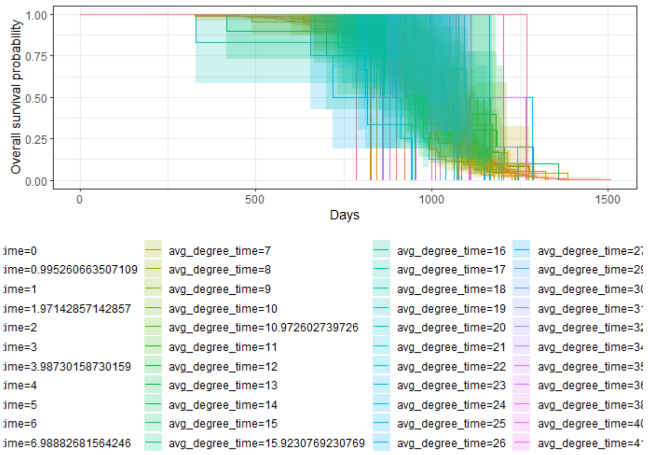


Figure B.1: Survival plot with application_tc_exam data

C. Tree plot details and Analysis

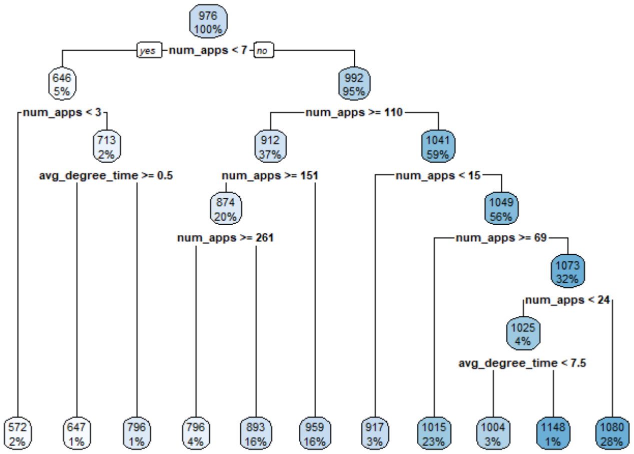


Figure C.1: applications_tc_exam decision tree fpr application time average

D. Cluster Analysis

This appendix contains an unused cluster analysis preformed on the application data. No meaningful clusters were found in the dataset.

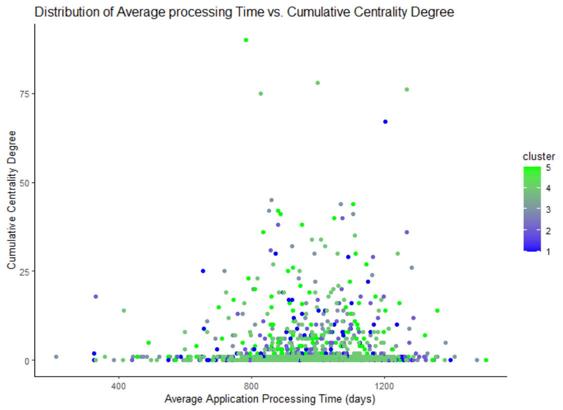


Figure D.1: Kmeans clustering k=5 with application_tc_exam dataset.

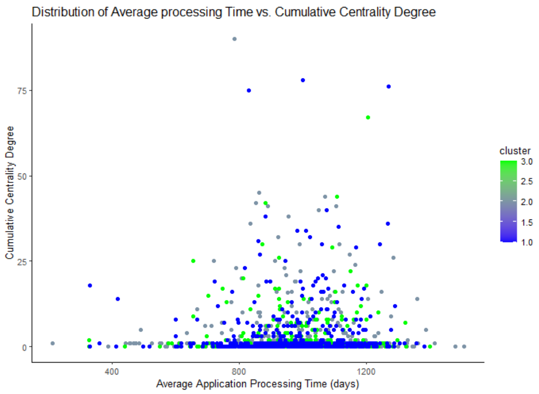


Figure D.2: Kmeans clustering k=3 with application_tc_exam dataset.

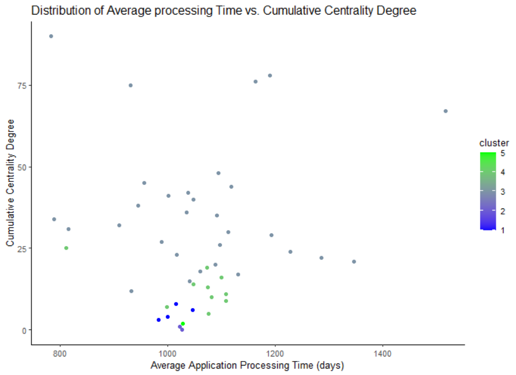


Figure D.3: Kmeans clustering k=5 using average application processing time for each centrality score.

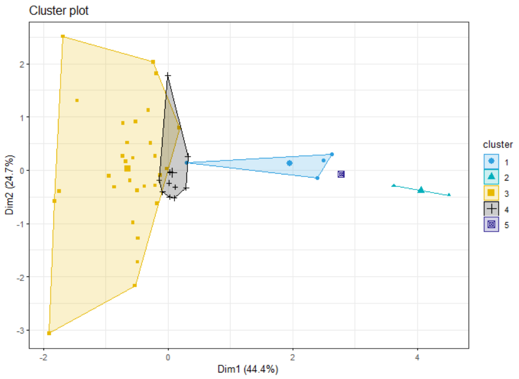
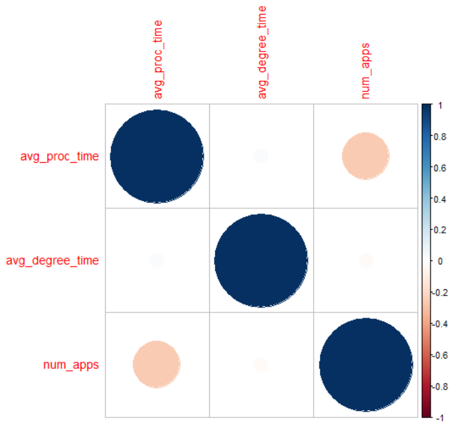


Figure D.4: Kmeans analysis with application_tc_exam dataset in reduced dimensional space. K=5

E. Correlation Plots



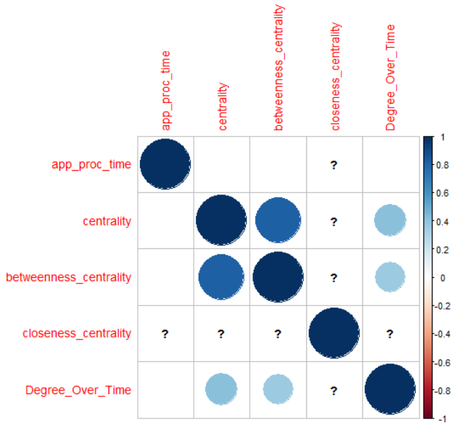


Figure E.1: Correlation Plot with applications_ts_exam

Figure E.2: Correlation plot with application_tc