Network Analysis Results

Jan Failenschmid (13869159), Daniel van der Meer (12227056), Dana Sleiffer (12767298)

15.12.2021

Summary

We estimated three networks to investigate variable relations for a subset of the General Society Survey (GSS) data set. Specifically, we looked at variables measuring: - attitude towards races
- attitude towards gay marriage
- social behavior
- standard of living
- attitude towards mothers working
- morals
- religiousness
- equal wealth
- political views
- news consumption - happiness

We estimated networks for the years 2006, 2008 and 2010. We then pruned the model, performed bootstraps and investigated model statistics. Resulting bootstrapped confidence intervals and edge differences can be seen below. The bootstrapped samples show good correlations with the original samples in terms of betweenness, closeness and strength for all three networks.
A network comparison between the networks of 2006, 2008 and 2010 shows no significant difference between the overall structure. When comparing the 2006 and 2008 networks, tests for specific edges show significant differences for the edges between "marasian"-"marhomo", "marasian"-"parsol", "marblk"-"fefam", and "rotapple"-"sprtprsn". For the networks of 2008 and 2010, the edges between "goodlife" and "eqwlth" seems to differ between the two networks.

Descriptives

Table 1
Descriptives of the Sample Demographics by Year

```
kable(describe(df_clean[df_clean$yearID == 2006, 3:7]))
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
age	1	633	41.263823	12.3501743	42	41.329389	14.826	18	63	45	-0.0555063	-1.0952998	0.4908754
sex	2	633	1.576619	0.4944854	2	1.595661	0.000	1	2	1	-0.3094054	-1.9072739	0.0196540
wrkstat	3	633	2.668246	2.4177390	1	2.297830	0.000	1	8	7	1.0581120	-0.5707812	0.0960965

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
degree	4	633	1.704581	1.1622740	1	1.611440	0.000	0	4	4	0.6499515	-0.7592092	0.0461962
race	5	633	1.368089	0.6764990	1	1.211045	0.000	1	3	2	1.5718085	0.9728134	0.0268884

```
kable(describe(df_clean[df_clean$yearID == 2008, 3:7]))
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
age	1	633	43.169036	12.3378671	44	43.214990	14.826	20	65	45	-0.0467736	-1.0872303	0.4903862
sex	2	633	1.573460	0.4949654	2	1.591716	0.000	1	2	1	-0.2963588	-1.9151896	0.0196731
wrkstat	3	633	2.712480	2.4474449	1	2.343195	0.000	1	8	7	1.0392817	-0.5905339	0.0972772
degree	4	633	1.733017	1.1609958	1	1.639053	0.000	0	4	4	0.6151438	-0.8469196	0.0461454
race	5	633	1.284360	0.5910931	1	1.140039	0.000	1	3	2	1.9358538	2.4983706	0.0234938

```
kable(describe(df_clean[df_clean$yearID == 2010, 3:7]))
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
age	1	633	45.257504	12.3431313	46	45.321499	14.826	21	67	46	-0.0542344	-1.0852929	0.4905954
sex	2	633	1.573460	0.4949654	2	1.591716	0.000	1	2	1	-0.2963588	-1.9151896	0.0196731
wrkstat	3	632	2.757911	2.3925268	1	2.399209	0.000	1	8	7	0.9734042	-0.6386610	0.0951696
degree	4	633	1.766193	1.1769117	1	1.668639	0.000	0	4	4	0.6048222	-0.9112572	0.0467780
race	5	633	1.285940	0.5916680	1	1.142012	0.000	1	3	2	1.9251902	2.4597264	0.0235167

Table 2
Descriptives of the Analysis Variables by Year

```
kable(describe(df_clean[df_clean$yearID == 2006, 8:ncol(df_clean)]))
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
marblk	1	632	2.863924	1.1931510	3	2.830039	1.4826	1	5	4	0.0114601	-0.6291715	0.0474610
marasian	2	631	2.717908	1.0463752	3	2.704950	1.4826	1	5	4	-0.0250164	-0.2526132	0.0416555
marhisp	3	633	2.698262	1.0975866	3	2.670611	1.4826	1	5	4	0.0281846	-0.4459676	0.0436252
marhomo	4	325	3.298462	1.5051857	4	3.371647	1.4826	1	5	4	-0.2372095	-1.4618306	0.0834927

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
socrel	5	633	3.279621	1.593850	5	3.213018	1.4826	1	7	6	0.3551953	-0.7998337	0.0633499
socommun	6	633	4.521327	2.032941	6	4.595661	2.9652	1	7	6	-0.1633485	-1.3838646	0.0808022
socfrend	7	633	3.747235	1.568861	4	3.656805	1.4826	1	7	6	0.3402966	-0.6712947	0.0623566
fechld	8	629	2.114467	0.857487	2	2.071287	1.4826	1	4	3	0.3239876	-0.6276258	0.0341902
fepresch	9	625	2.673600	0.794022	5	3	2.698603	0.0000	1	4	3	-0.3676646	-0.2337247	0.0317609
fefam	10	625	2.806400	0.845021	6	3	2.852295	1.4826	1	4	3	-0.3703700	-0.4150846	0.0338009
punsin	11	597	2.666667	0.993828	8	3	2.707724	1.4826	1	4	3	-0.1368944	-1.0559341	0.0406747
blkwhite	12	629	1.842607	0.919895	2	1	1.699010	1.4826	1	4	3	0.9647672	0.1105746	0.0366786
rotapple	13	623	2.176565	0.972762	4	2	2.096192	1.4826	1	4	3	0.3536455	-0.9025289	0.0389729
permoral	14	624	2.003205	0.890464	5	2	1.922000	1.4826	1	4	3	0.5521009	-0.4948338	0.0356471
relpersn	15	630	2.379365	0.916490	6	2	2.349206	1.4826	1	4	3	0.2831282	-0.7291065	0.0365139
sprtprsn	16	629	2.068363	0.904747	5	2	2.000000	1.4826	1	4	3	0.4194008	-0.7100082	0.0360746
polviews	17	620	4.130645	1.442723	4	4	4.165323	1.4826	1	7	6	-0.1285610	-0.4327341	0.0579412
news	18	633	2.590837	1.395645	2	2	2.489152	1.4826	1	5	4	0.4076202	-1.1534525	0.0554719
happy	19	632	1.789557	0.602439	7	2	1.739130	0.0000	1	3	2	0.1264360	-0.4820253	0.0239638

```
kable(describe(df_clean[df_clean$yearID == 2008, 8:ncol(df_clean)]))
```

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
marblk	1	633	2.818325	1.123756	3	2	2.783037	0.0000	1	5	4	0.0131172	-0.3403587	0.0446653
marasian	2	631	2.654517	0.992579	3	3	2.639604	0.0000	1	5	4	-0.0853587	-0.0191278	0.0395140
marhisp	3	632	2.666139	1.024698	3	3	2.650198	0.0000	1	5	4	-0.0762910	-0.1802662	0.0407603
marhomo	4	325	3.215385	1.534503	5	3	3.268199	2.9652	1	5	4	-0.1228054	-1.5187620	0.0851189
socrel	5	633	3.355450	1.612365	5	3	3.285996	1.4826	1	7	6	0.3386415	-0.8752943	0.0640858
socommun	6	633	4.526066	1.947011	6	4	4.585799	2.9652	1	7	6	-0.1117785	-1.3071379	0.0773868
socfrend	7	633	3.796209	1.482338	5	4	3.712032	1.4826	1	7	6	0.2877974	-0.6076922	0.0589177
fechld	8	632	2.058544	0.849007	9	2	2.025692	1.4826	1	4	3	0.2611100	-0.8317480	0.0337717

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
fepresch	9	628	2.746815	0.7894274	3	2.773810	0.0000	1	4	3	-0.3949329	-0.1522429	0.0315016
fefam	10	630	2.846032	0.8496376	3	2.902778	1.4826	1	4	3	-0.4476290	-0.3392421	0.0338504
punsin	11	598	2.658863	0.9893978	3	2.697917	1.4826	1	4	3	-0.1280599	-1.0444089	0.0404595
blkwhite	12	628	1.882166	0.9578687	2	1.730159	1.4826	1	4	3	0.9537418	-0.0188122	0.0382231
rotapple	13	627	2.283892	0.9947685	2	2.230616	1.4826	1	4	3	0.3435934	-0.9192311	0.0397272
permoral	14	618	2.074434	0.9480647	2	1.973790	1.4826	1	4	3	0.5239159	-0.6658216	0.0381368
relpersn	15	631	2.413629	0.9587051	2	2.392079	1.4826	1	4	3	0.2622582	-0.8822654	0.0381654
sprtprsn	16	630	2.096825	0.9031773	2	2.025794	1.4826	1	4	3	0.4295725	-0.6411398	0.0359834
polviews	17	623	4.017657	1.3975206	4	4.028056	1.4826	1	7	6	-0.0560742	0.4013459	0.0559905
news	18	633	2.644550	1.4013012	2	2.556213	1.4826	1	5	4	0.3851155	-1.1695039	0.0556967
happy	19	632	1.805380	0.5892079	2	1.762846	0.0000	1	3	2	0.0739527	-0.3776021	0.0234374

```
kable(describe(df_clean[df_clean$yearID == 2010, 8:ncol(df_clean)]))
```

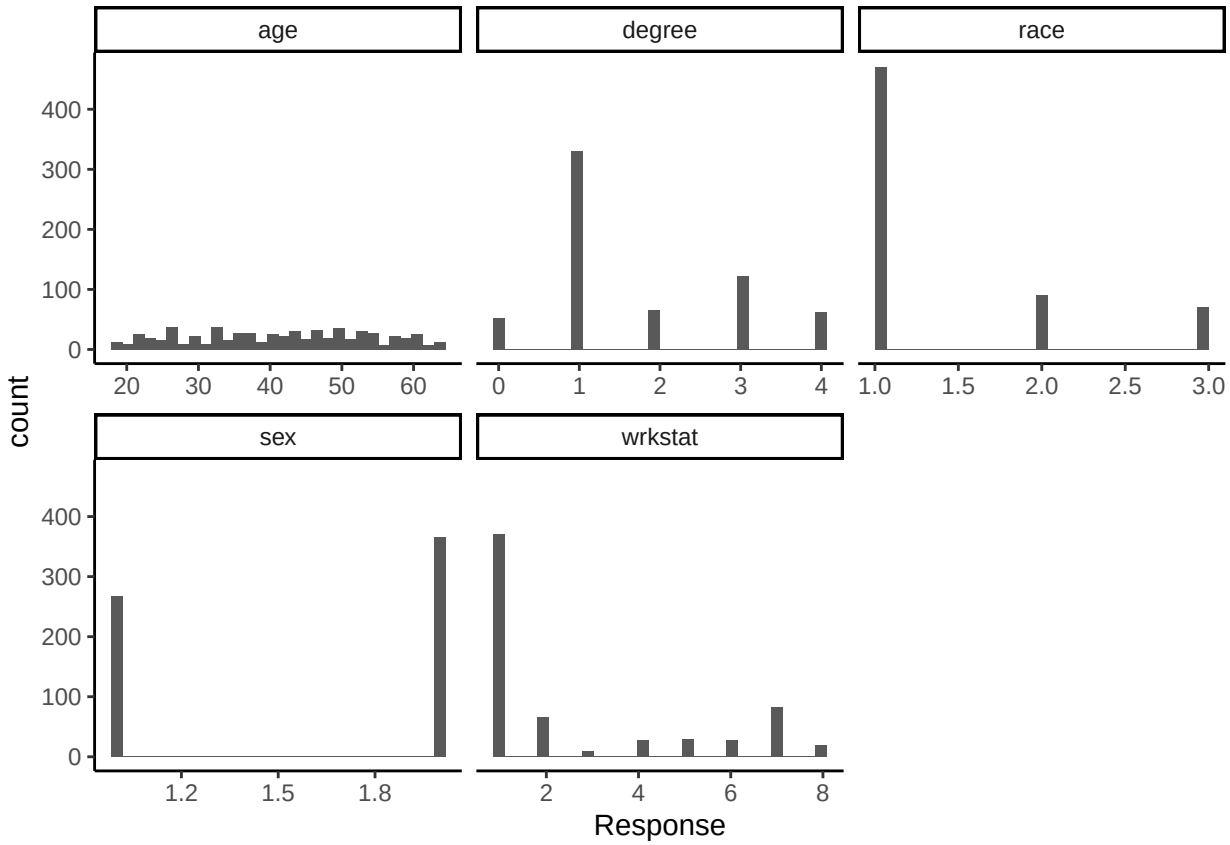
	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
marblk	1	633	2.777251	1.1120687	3	2.739645	0.0000	1	5	4	0.0393912	-0.3006363	0.0442008
marasian	2	633	2.661927	0.9576908	3	2.662722	0.0000	1	5	4	-0.1900111	-0.0114227	0.0380648
marhisp	3	633	2.657188	0.9925432	3	2.646943	0.0000	1	5	4	-0.1182396	-0.0815510	0.0394501
marhomo	4	323	3.148607	1.5209376	3	3.185328	1.4826	1	5	4	-0.0766771	-1.5160156	0.0846272
socrel	5	633	3.320695	1.5880781	3	3.250493	1.4826	1	7	6	0.3355325	-0.8725266	0.0631204
socommun	6	632	4.639240	1.9727730	5	4.731225	2.9652	1	7	6	-0.2226825	-1.3015517	0.0784727
socfrend	7	633	4.007899	1.5078699	4	3.923077	1.4826	1	7	6	0.2824416	-0.6490580	0.0599324
fechld	8	630	2.055556	0.8468172	2	1.998016	1.4826	1	4	3	0.4593057	-0.4145642	0.0337380
fepresch	9	630	2.742857	0.7631652	3	2.742063	0.0000	1	4	3	-0.2166982	-0.2790061	0.0304052
fefam	10	629	2.829889	0.8509999	3	2.883168	1.4826	1	4	3	-0.4122405	-0.3907343	0.0339316
punsin	11	607	2.701812	0.9758030	3	2.751540	1.4826	1	4	3	-0.1856720	-0.9913494	0.0396066
blkwhite	12	628	1.834395	0.9165996	2	1.684524	1.4826	1	4	3	1.0143697	0.2525827	0.0365763

	vars	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
rotapple	13	628	2.286624	0.9866797	2	2.234127	1.4826	1	4	3	0.3685772	-0.8730463	0.0393728
permoral	14	621	2.004831	0.8953153	2	1.913481	1.4826	1	4	3	0.6098533	-0.3844021	0.0359278
relpersn	15	632	2.468354	0.9729846	2	2.460474	1.4826	1	4	3	0.1713168	-0.9723031	0.0387032
sprtprsn	16	633	2.146919	0.9186331	2	2.078895	1.4826	1	4	3	0.3544212	-0.7563220	0.0365124
polviews	17	621	4.114332	1.4078595	4	4.148893	1.4826	1	7	6	-0.1162391	-0.5026698	0.0564955
news	18	633	2.775671	1.4820366	3	2.719921	1.4826	1	5	4	0.2365150	-1.3742949	0.0589057
happy	19	633	1.848341	0.5998955	2	1.810651	0.0000	1	3	2	0.0708944	-0.3633098	0.0238437

Figure 1
Histograms of Sample Demographics by Year

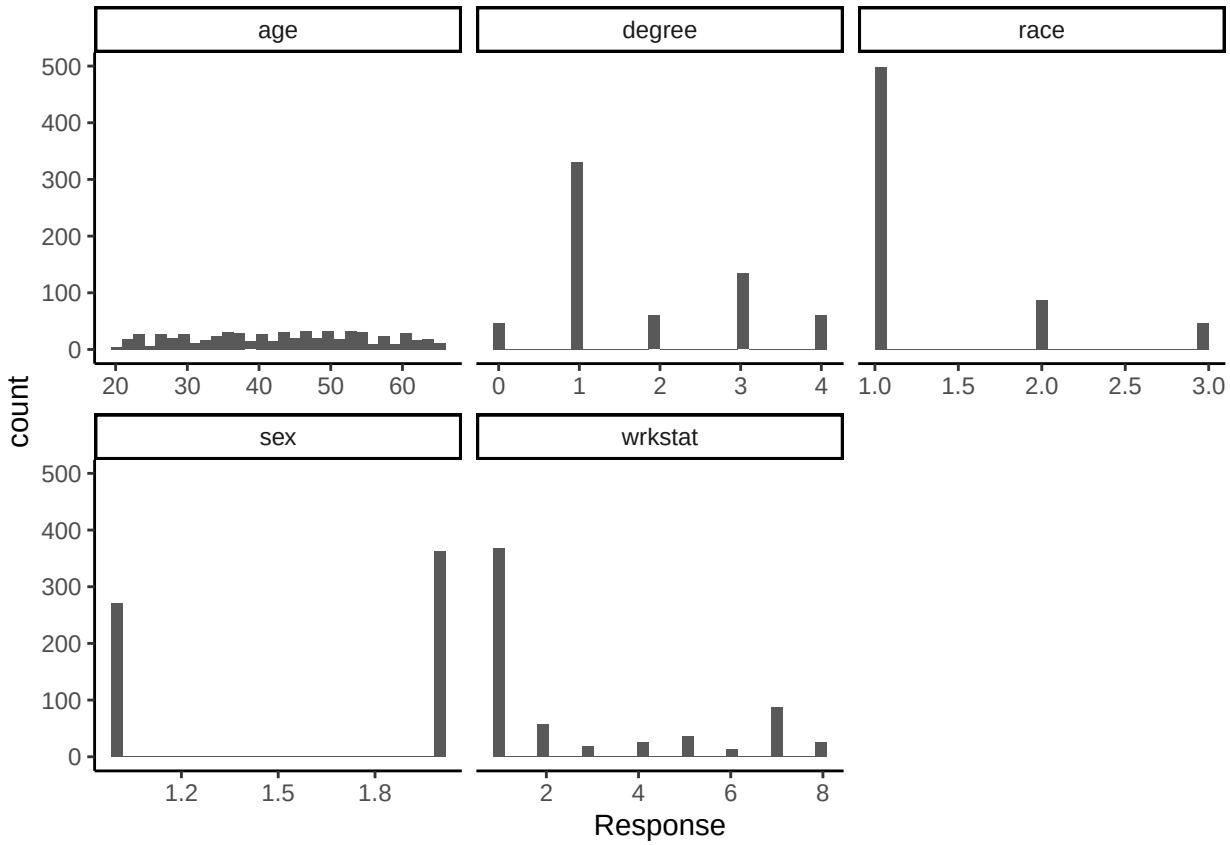
```
ggplot(gather(df_clean[df_clean$yearID == 2006, 3:7]), aes(value)) +
  geom_histogram() +
  facet_wrap(~key, scales = "free_x") +
  theme_classic() +
  xlab("Response")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(gather(df_clean[df_clean$yearID == 2008, 3:7]), aes(value)) +
  geom_histogram() +
  facet_wrap(~key, scales = "free_x") +
  theme_classic() +
  xlab("Response")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(gather(df_clean[df_clean$yearID == 2010, 3:7]), aes(value)) +
  geom_histogram() +
  facet_wrap(~key, scales = "free_x") +
  theme_classic() +
  xlab("Response")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

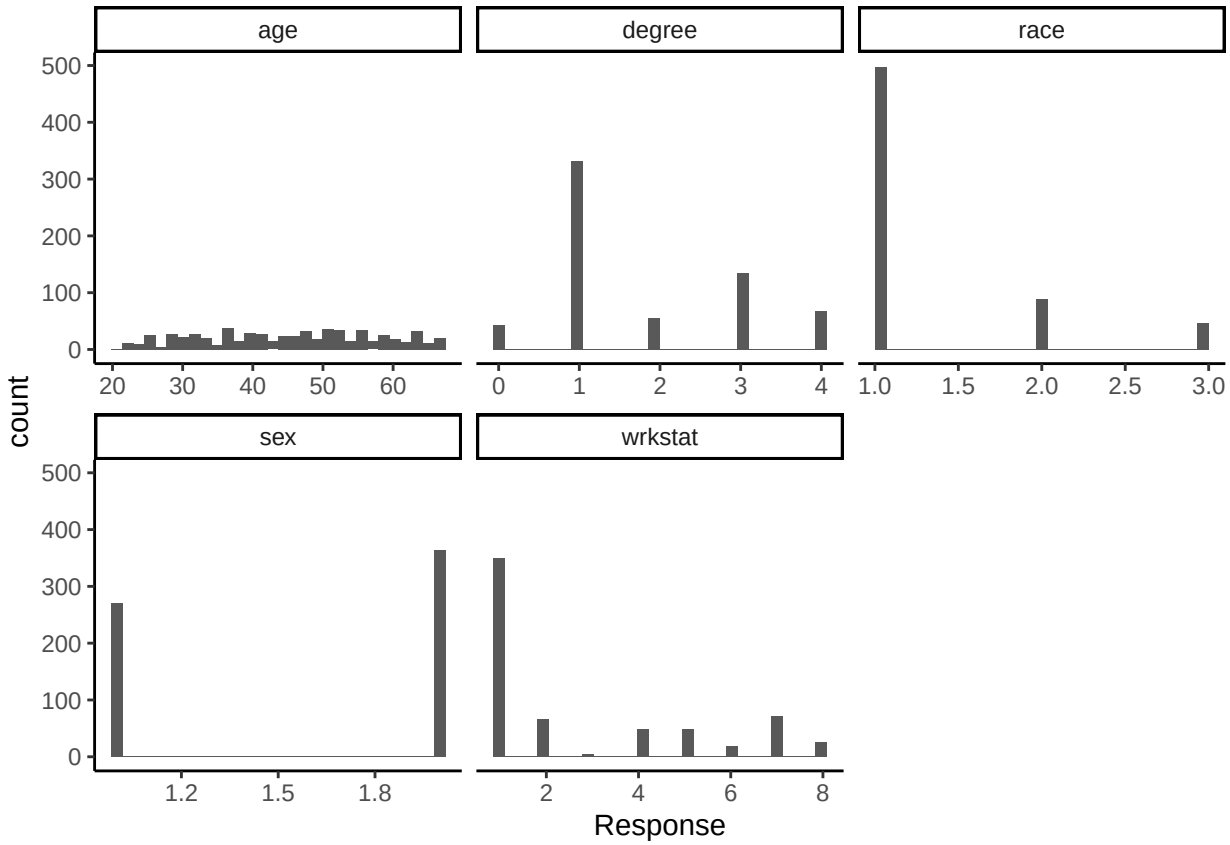
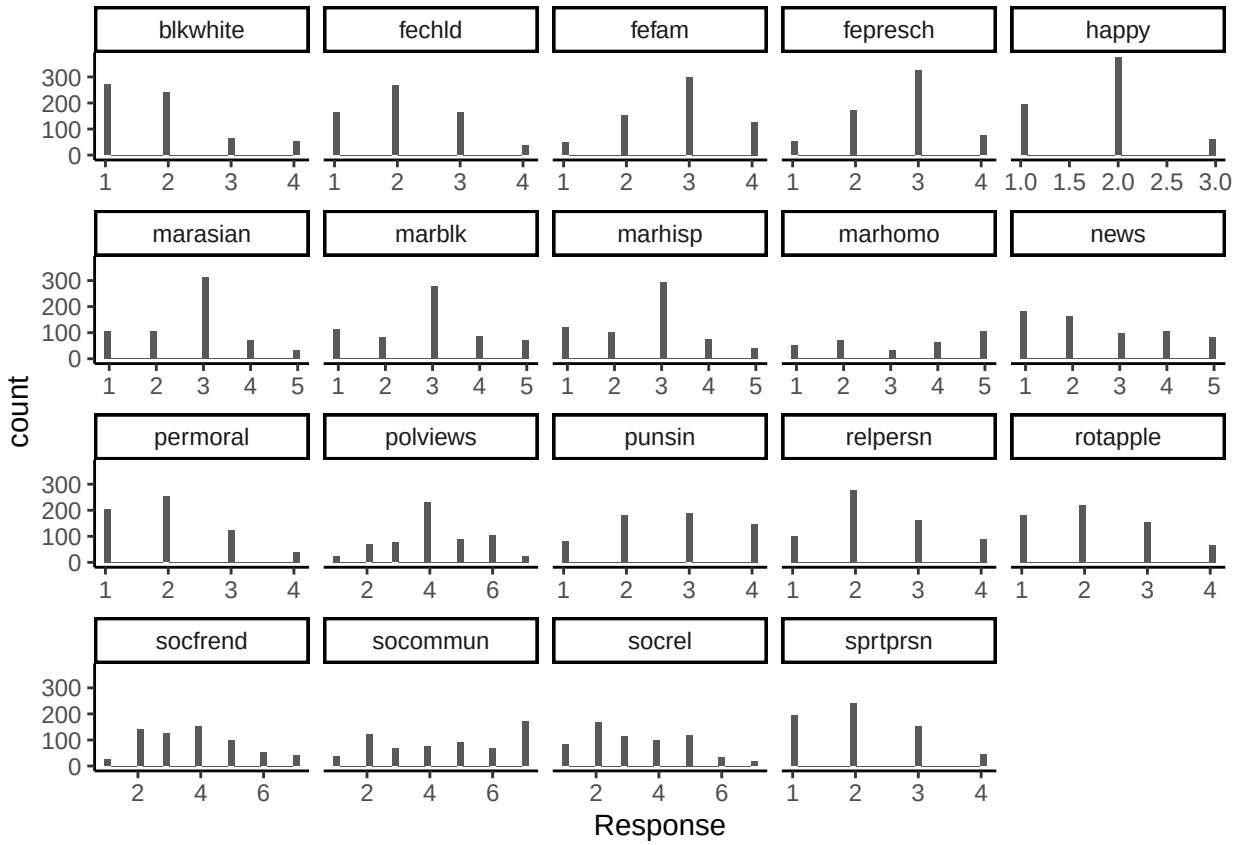


Figure 2
Histograms of Analysis Variables by Year

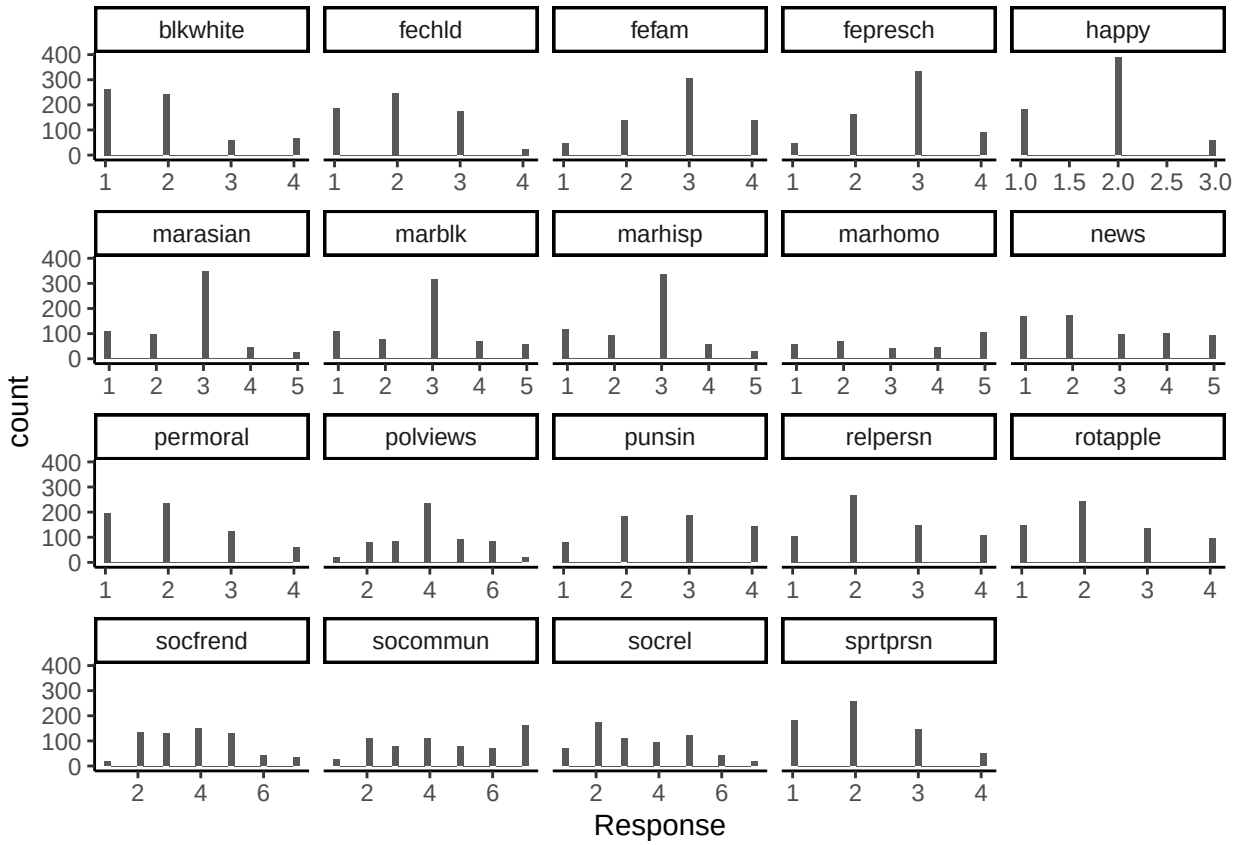
```
ggplot(gather(df_clean[df_clean$yearID == 2006, 8:ncol(df_clean)]), aes(value)) +
  geom_histogram() +
  facet_wrap(~key, scales = "free_x") +
  theme_classic() +
  xlab("Response")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



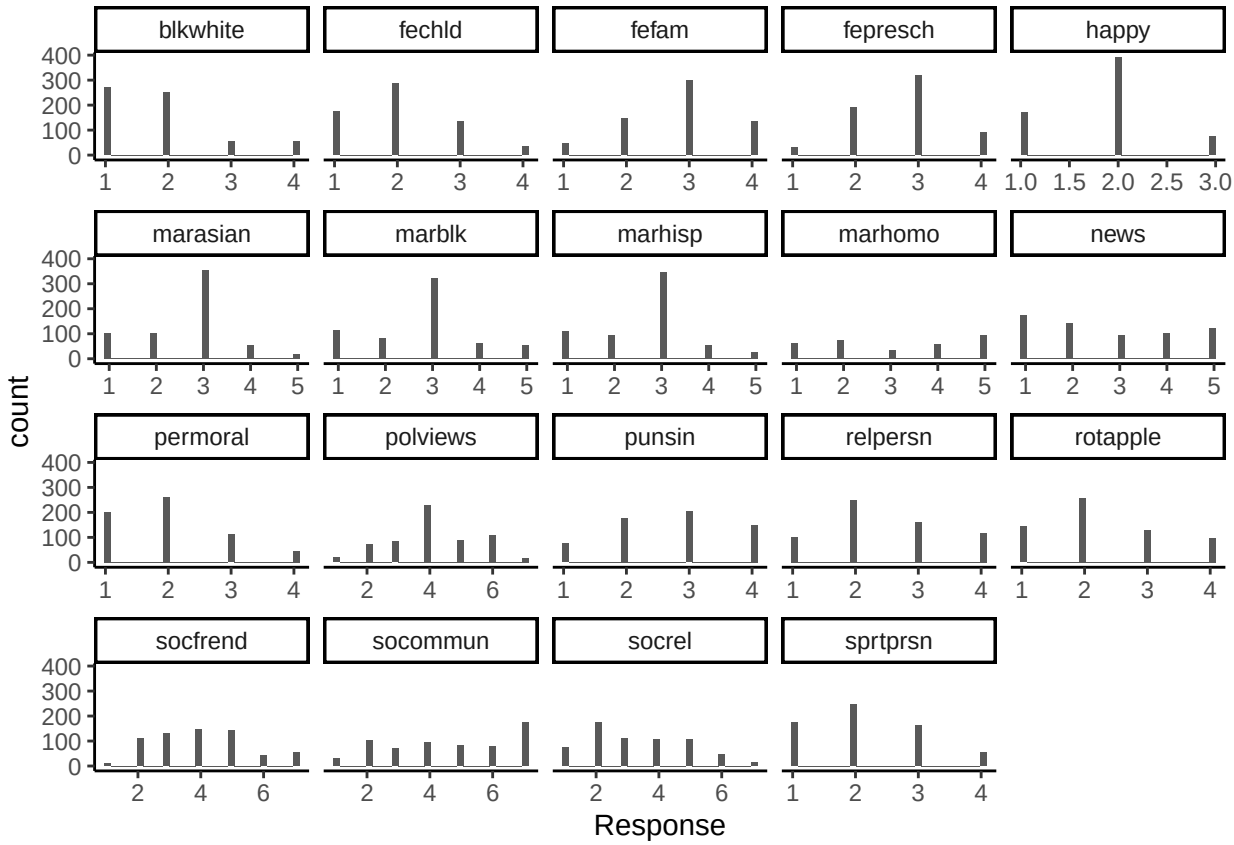

```
ggplot(gather(df_clean[df_clean$yearID == 2008, 8:ncol(df_clean)]), aes(value)) +
  geom_histogram() +
  facet_wrap(~key, scales = "free_x") +
  theme_classic() +
  xlab("Response")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
ggplot(gather(df_clean[df_clean$yearID == 2010, 8:ncol(df_clean)]), aes(value)) +
  geom_histogram() +
  facet_wrap(~key, scales = "free_x") +
  theme_classic() +
  xlab("Response")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```


blkwhite
fechld
fefam
fepresch
happy
marasian
marblk
marhisp
marhomo
news
permoral
polviews
punsin
relpersn
rotapple
socfrend
socommun
socrel
sprtprsn
count
Response

Network Analysis

Network for 2006

Figure 3
Network for 2006

```
#Plot the networks
plot(network2006, layout = avrgLayout, maximum = maxi)
```

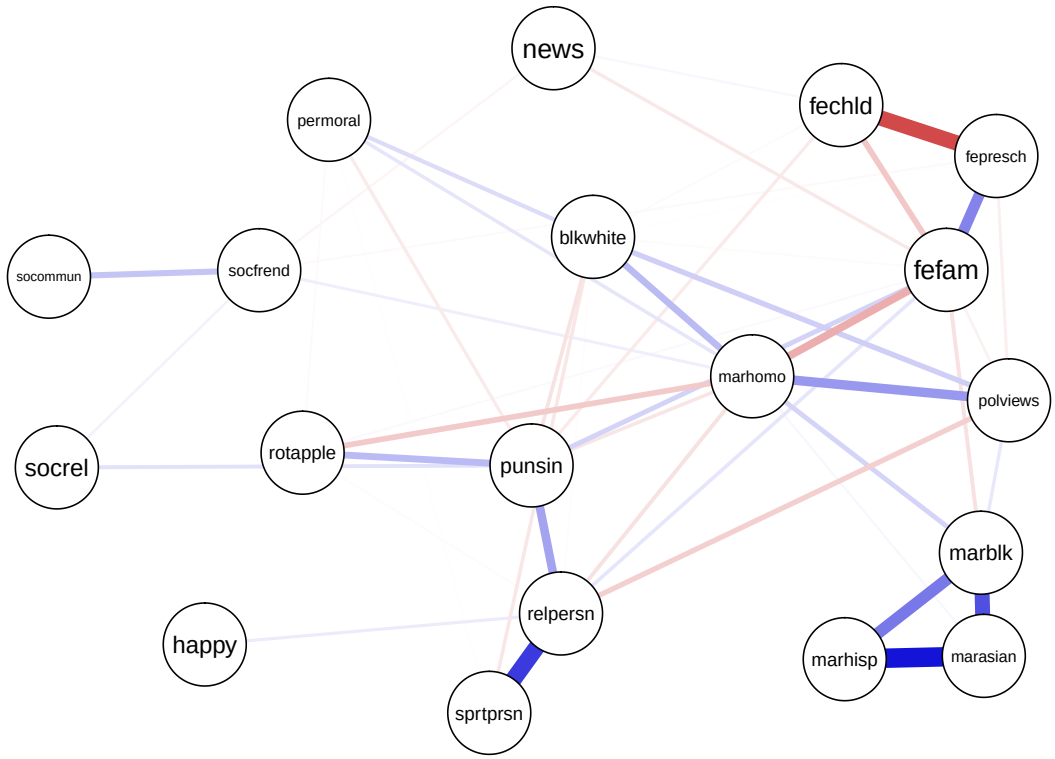


Figure 4
Bootstrapped Confidence Intervals for the Edges

```
#Plot Bootstrapped CI's
plot(boot_nonparametric_2006, order = "sample", labels = F)
```

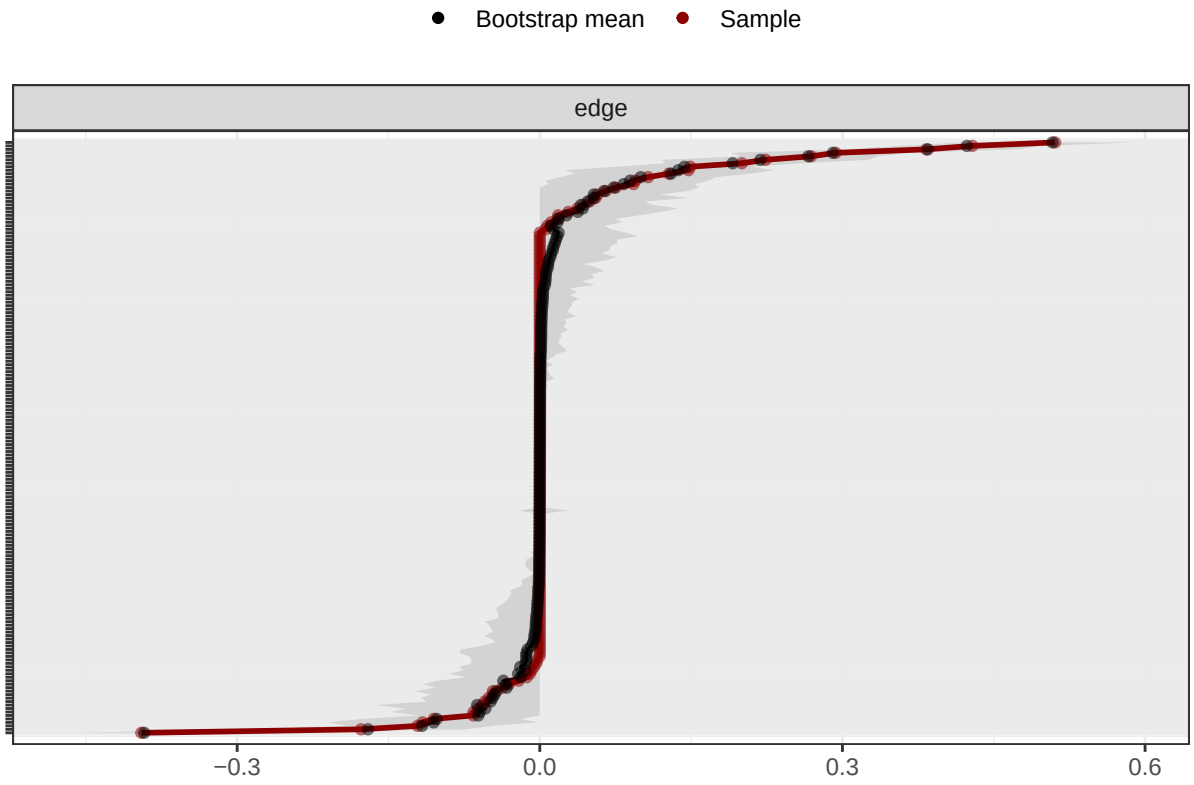


Figure 5

Bootstrapped Edge Differrence

```
#Plot Bootstrapped Differnece
plot(boot_nonparametric_2006, plot = "difference", onlyNonZero = TRUE, order = "sample")
```

```
## Expected significance level given number of bootstrap samples is approximately: 0.05
```

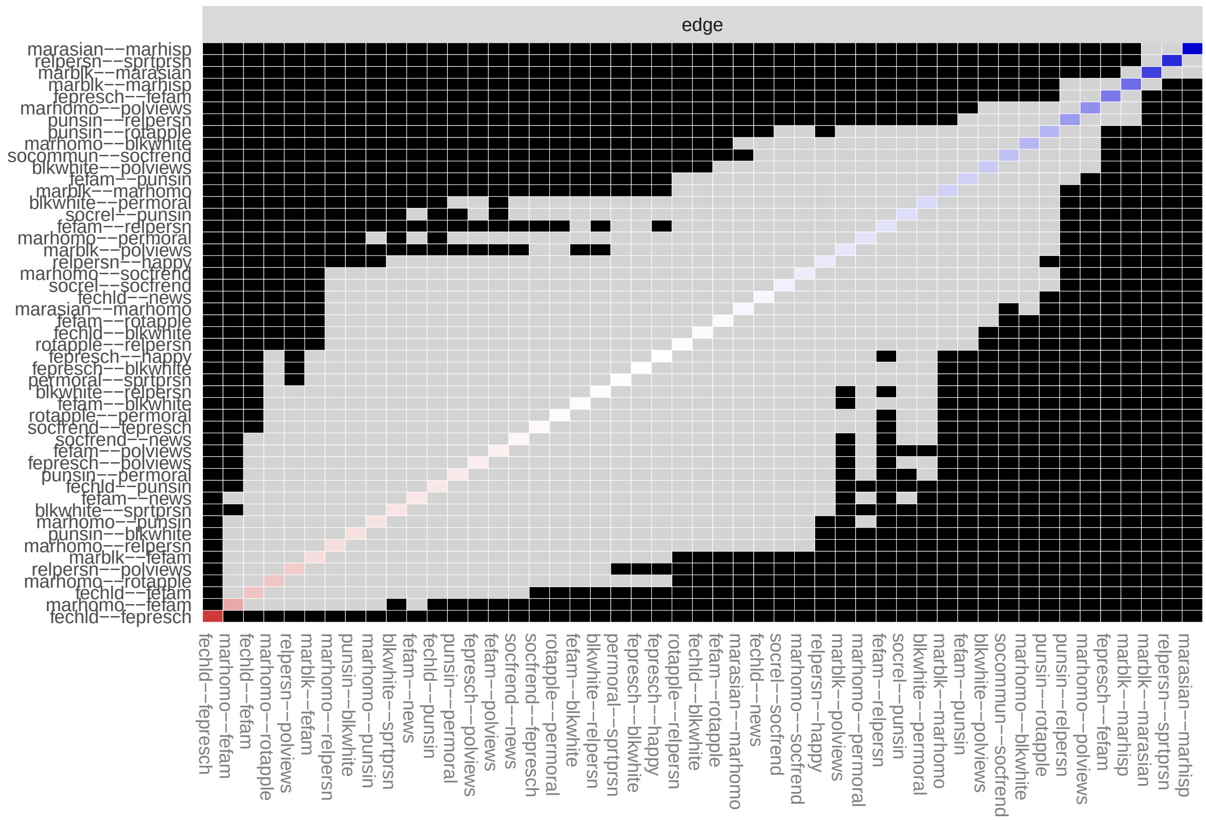
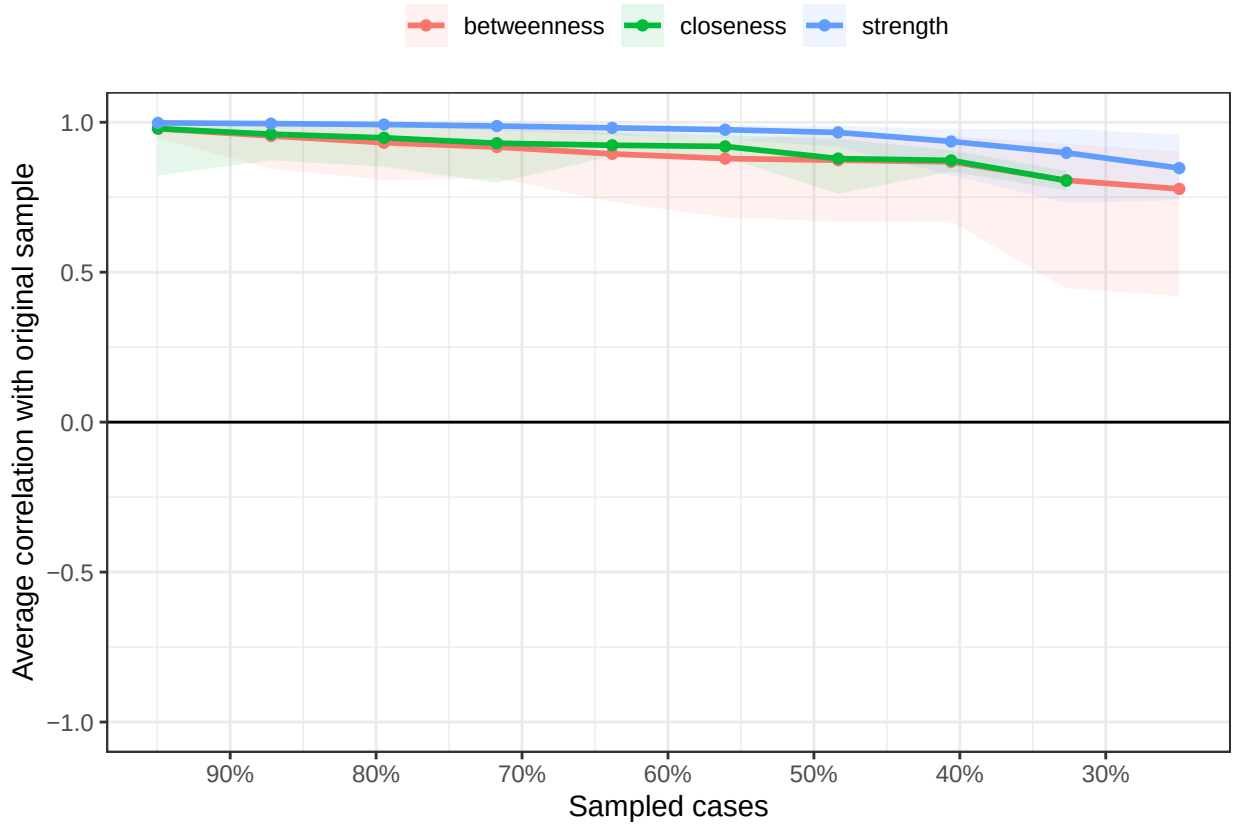


Figure 6
Case-Drop Bootstrapp Centrality Measures Stability Plot

```
#Stability Plot
plot(boot_casedrop_2006, statistics = c("strength", "betweenness", "closeness"))
```



Cor-Stability Analysis

```
corStability(boot_casedrop_2006)
```

```
## === Correlation Stability Analysis ===
##
## Sampling levels tested:
##    nPerson Drop%  n
## 1      158  75.0 22
## 2      207  67.3 24
## 3      257  59.4 38
## 4      306  51.7 36
## 5      355  43.9 25
## 6      404  36.2 28
## 7      454  28.3 28
## 8      503  20.5 44
## 9      552  12.8 34
## 10     601   5.1 21
##
```

```
## Maximum drop proportions to retain correlation of 0.7 in at least 95% of the samples:
##
## betweenness: 0.517
##   - For more accuracy, run bootnet(..., caseMin = 0.439, caseMax = 0.594)
##
## closeness: 0.051 (CS-coefficient is lowest level tested)
##   - For more accuracy, run bootnet(..., caseMin = 0, caseMax = 0.128)
##
## strength: 0.75 (CS-coefficient is highest level tested)
##   - For more accuracy, run bootnet(..., caseMin = 0.673, caseMax = 1)
##
## Accuracy can also be increased by increasing both 'nBoots' and 'caseN'.
```

Network for 2008

Figure 7
Network for 2008

```
#Plot the networks
plot(network2008, layout = avrgLayout, maximum = maxi)
```

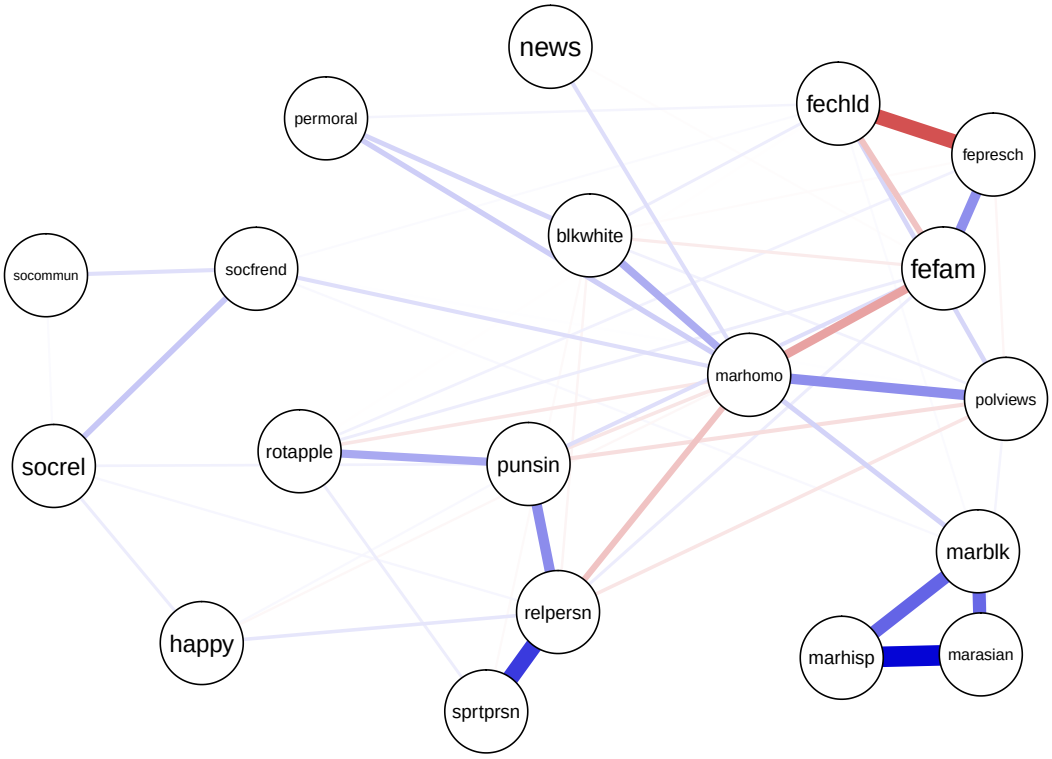


Figure 8
Bootstrapped Confidence Intervals for the Edges

```
#Plot Bootstrapped CI's
plot(boot_nonparametric_2008, order = "sample", labels = F)
```

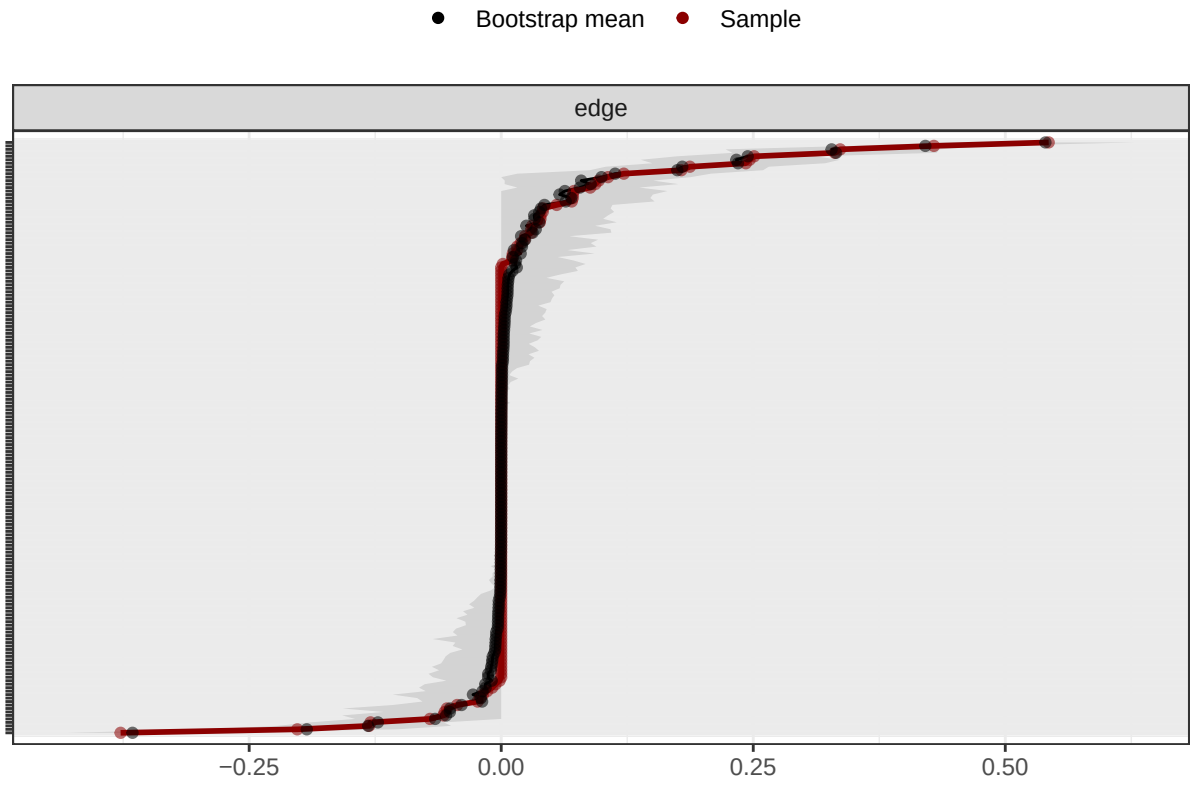


Figure 9
Bootstrapped Edge Differrence

```
#Plot Bootstrapped Differnece
plot(boot_nonparametric_2008, plot = "difference", onlyNonZero = TRUE, order = "sample")
```

```
## Expected significance level given number of bootstrap samples is approximately: 0.05
```

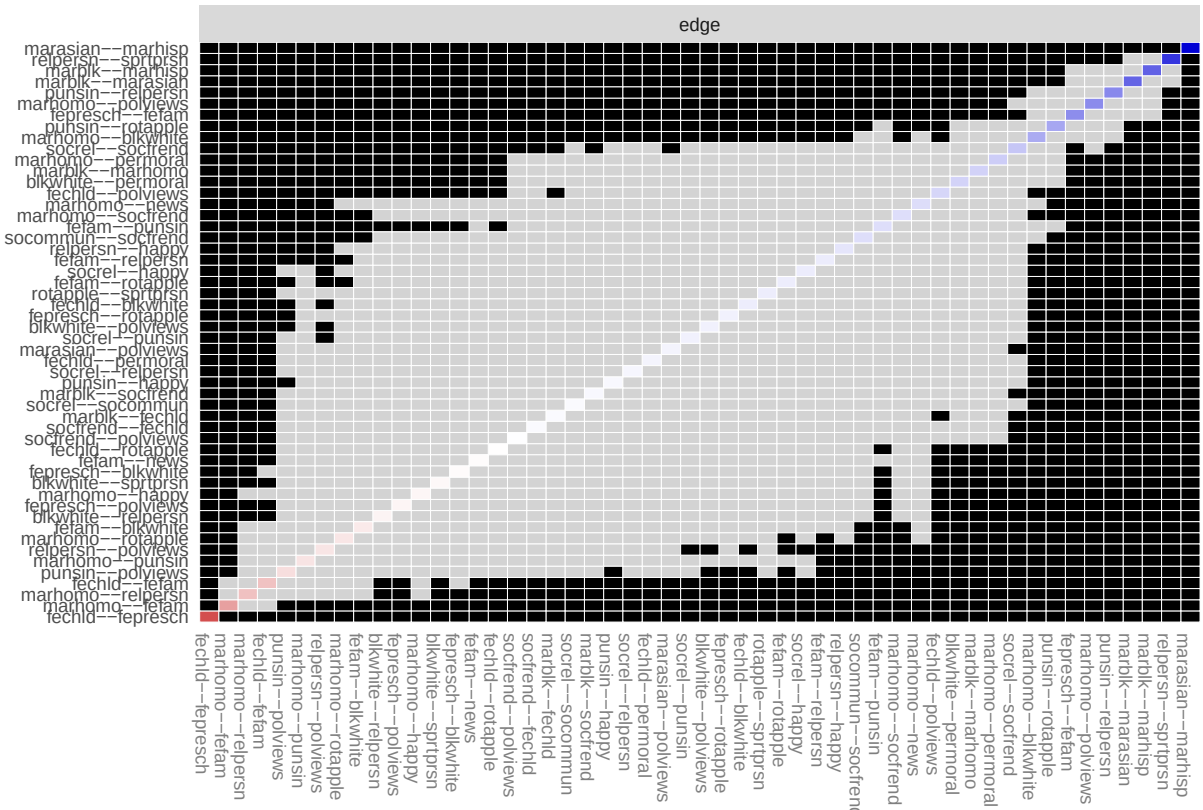
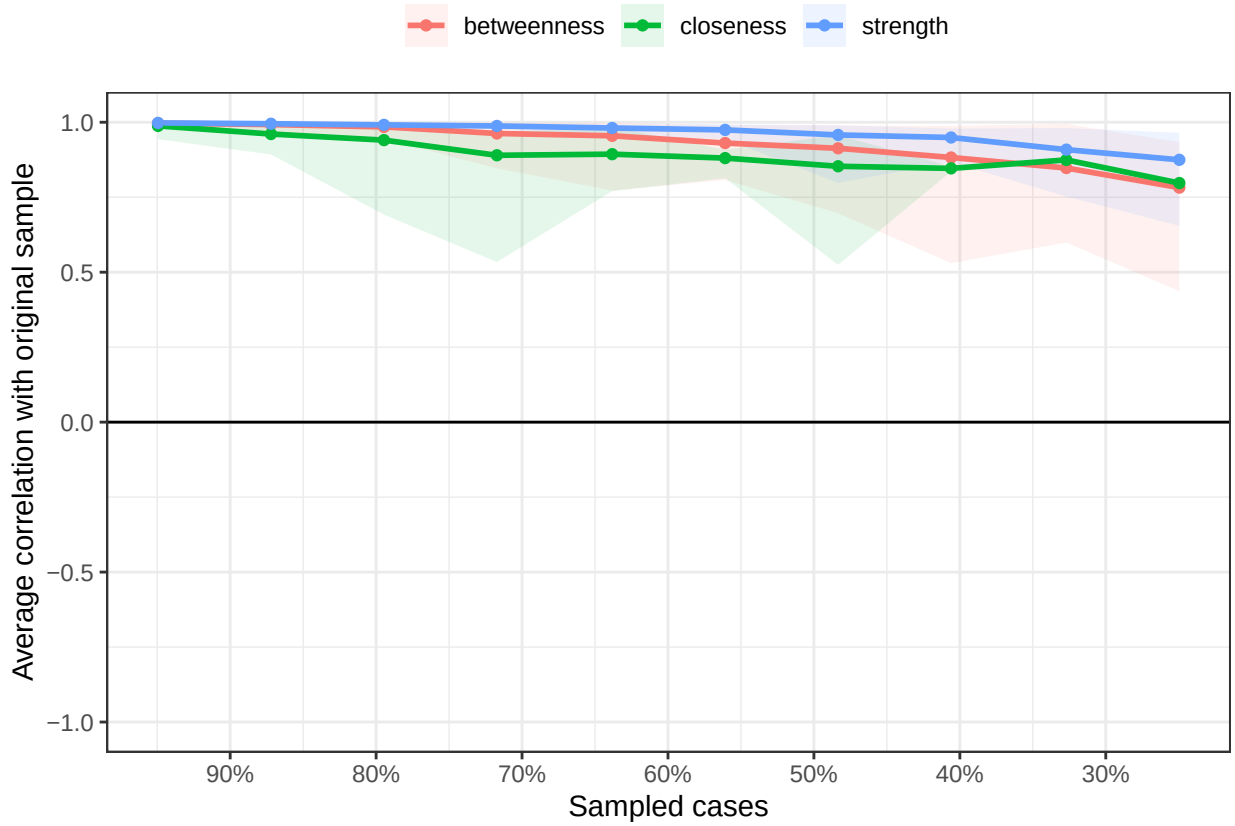


Figure 10
Case-Drop Bootstrapp Centrality Measures Stability Plot

```
#Stability Plot
plot(boot_casedrop_2008, statistics = c("strength", "betweenness", "closeness"))
```



Cor-Stability Analysis

```
corStability(boot_casedrop_2008)
```

```
## === Correlation Stability Analysis ===
##
## Sampling levels tested:
##    nPerson Drop%  n
## 1      158  75.0 31
## 2      207  67.3 28
## 3      257  59.4 23
## 4      306  51.7 25
## 5      355  43.9 28
## 6      404  36.2 39
## 7      454  28.3 26
## 8      503  20.5 37
## 9      552  12.8 29
## 10     601   5.1 34
##
```

```
## Maximum drop proportions to retain correlation of 0.7 in at least 95% of the samples:
##
## betweenness: 0.594
##   - For more accuracy, run bootnet(..., caseMin = 0.517, caseMax = 0.673)
##
## closeness: 0.128
##   - For more accuracy, run bootnet(..., caseMin = 0.051, caseMax = 0.205)
##
## strength: 0.75 (CS-coefficient is highest level tested)
##   - For more accuracy, run bootnet(..., caseMin = 0.673, caseMax = 1)
##
## Accuracy can also be increased by increasing both 'nBoots' and 'caseN'.
```

Network for 2010

Figure 11
Network for 2010

```
#Plot the networks
plot(network2010, layout = avrgLayout, maximum = maxi)
```

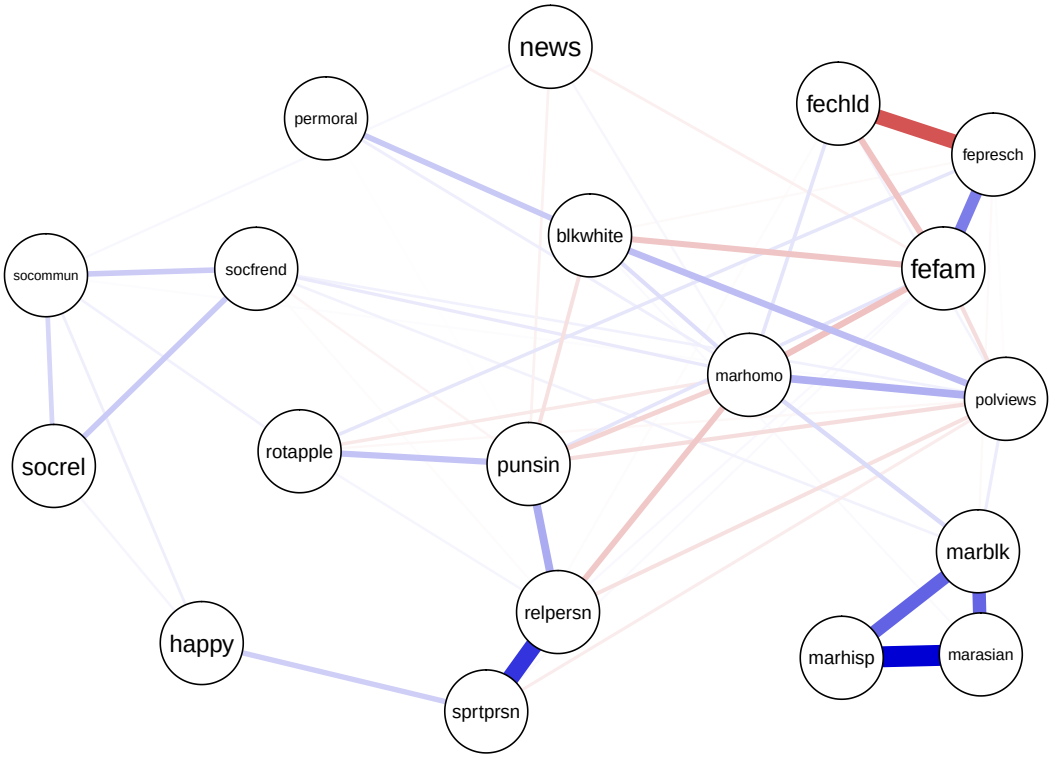


Figure 12
Bootstrapped Confidence Intervals for the Edges

```
#Plot Bootstrapped CI's
plot(boot_nonparametric_2010, order = "sample", labels = F)
```

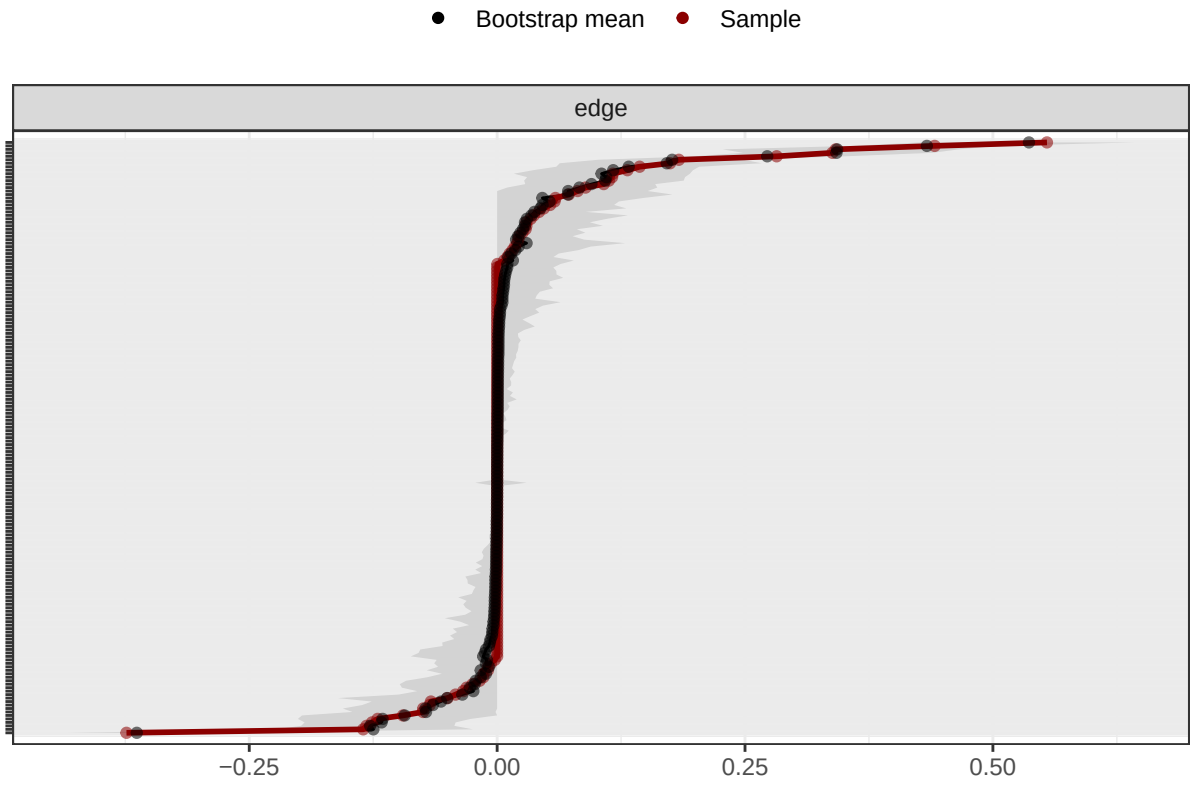


Figure 13

Bootstrapped Edge Differrence

```r
#Plot Bootstrapped Differnece
plot(boot_nonparametric_2010, plot = "difference", onlyNonZero = TRUE, order = "sample")
```

```
## Expected significance level given number of bootstrap samples is approximately: 0.05
```

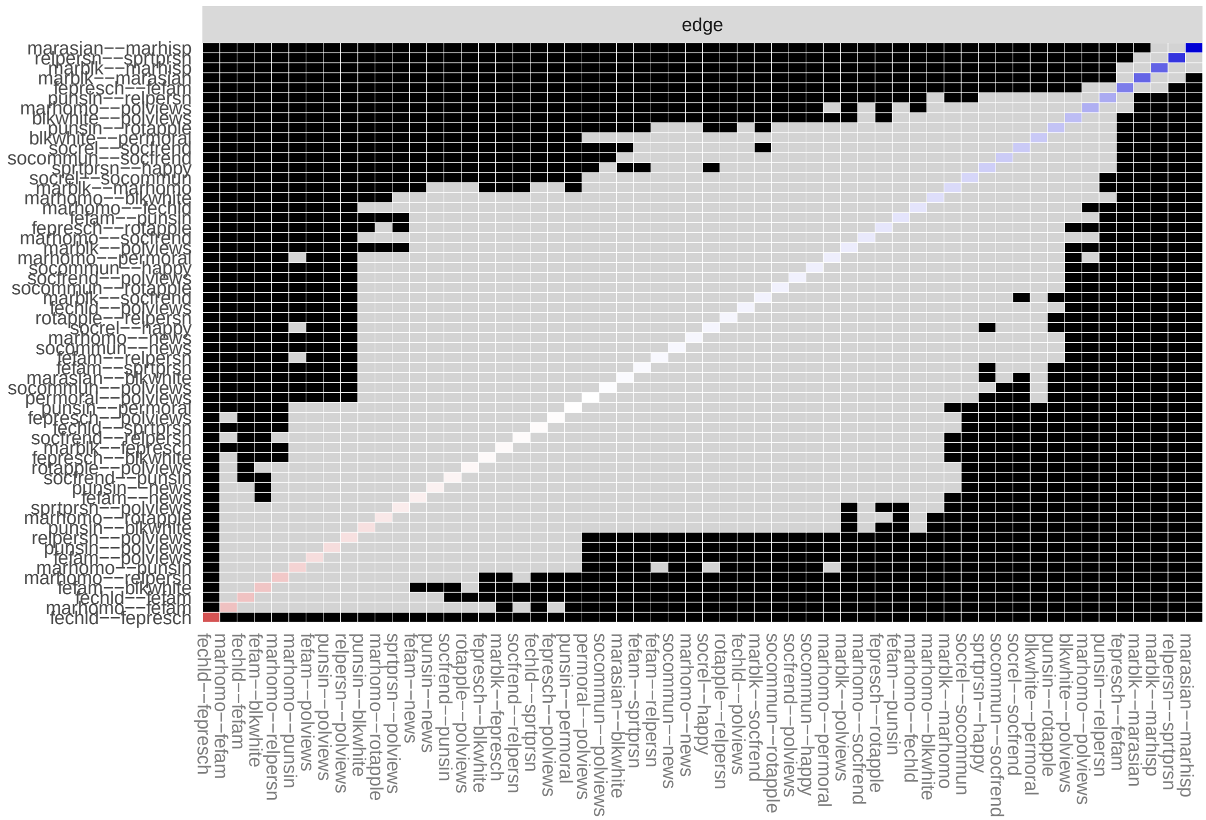
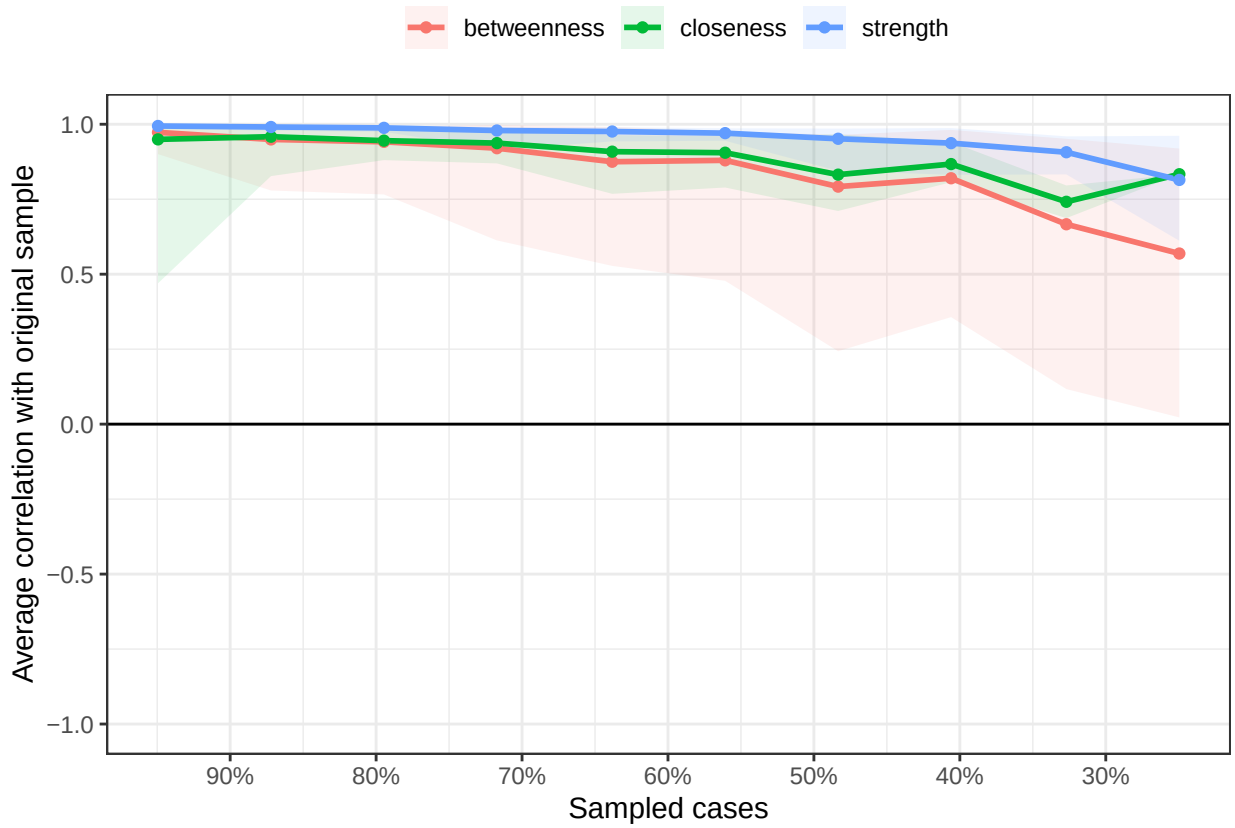


Figure 14
Case-Drop Bootstrapp Centrality Measures Stability Plot

```
#Stability Plot
plot(boot_casedrop_2010, statistics = c("strength", "betweenness", "closeness"))
```



Cor-Stability Analysis

```
corStability(boot_casedrop_2010)
```

```
## === Correlation Stability Analysis ===
##
## Sampling levels tested:
##    nPerson Drop%  n
## 1      158  75.0 21
## 2      207  67.3 33
## 3      257  59.4 35
## 4      306  51.7 36
## 5      355  43.9 23
## 6      404  36.2 26
## 7      454  28.3 34
## 8      503  20.5 27
## 9      552  12.8 28
## 10     601   5.1 37
##
```

```
## Maximum drop proportions to retain correlation of 0.7 in at least 95% of the samples:
##
## betweenness: 0.439
##   - For more accuracy, run bootnet(..., caseMin = 0.362, caseMax = 0.517)
##
## closeness: 0
##   - For more accuracy, run bootnet(..., caseMin = 0, caseMax = 0.051)
##
## strength: 0.673
##   - For more accuracy, run bootnet(..., caseMin = 0.594, caseMax = 0.75)
##
## Accuracy can also be increased by increasing both 'nBoots' and 'caseN'.
```

Network Comparison

Network Comparison Test 1

Figure 14
Global Strength Test

```
# Global Strenght Test
plot(Comparison1, what = "strength")
```

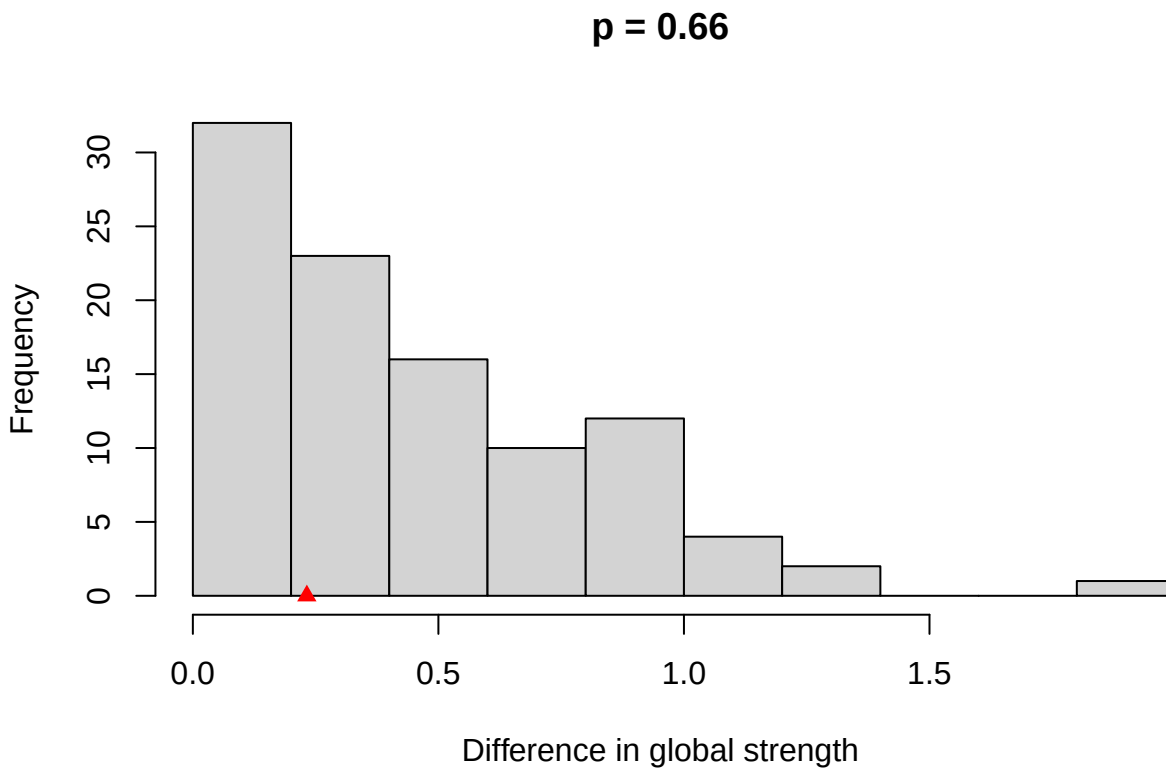


Figure 15
Omnibus Test

```
# Omnibus Test
plot(Comparison1, what = "network")
```

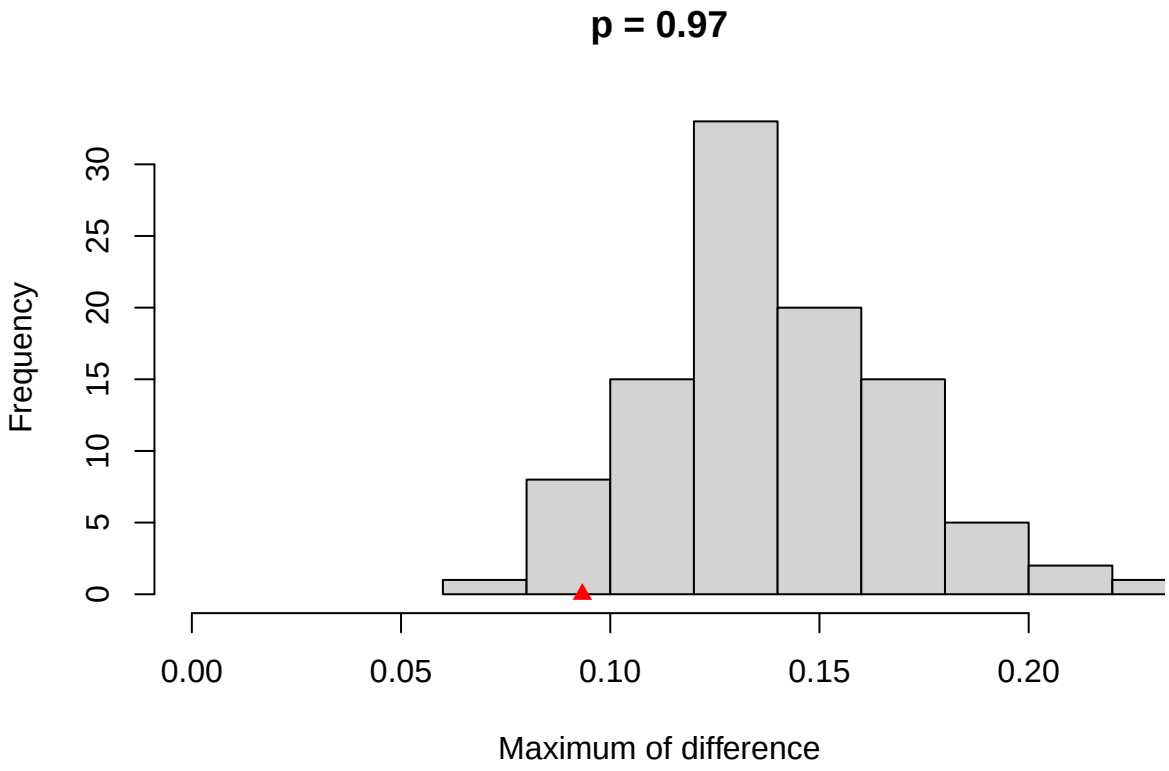


Figure 16
Edge Difference Test Significant Edges

```
#Edge Difference Test
kable(Comparison1$einv.pvals[which(Comparison1$einv.pvals[,3]<.0500),])
```

	Var1	Var2	p-value
140	socfrend	fechld	0.04
172	marblk	fefam	0.04
315	punsin	polviews	0.04
330	socfrend	news	0.02

Network Comparison Test 2

Figure 17
Global Strength Test

```
# Global Strength Test
plot(Comparison2, what = "strength")
```

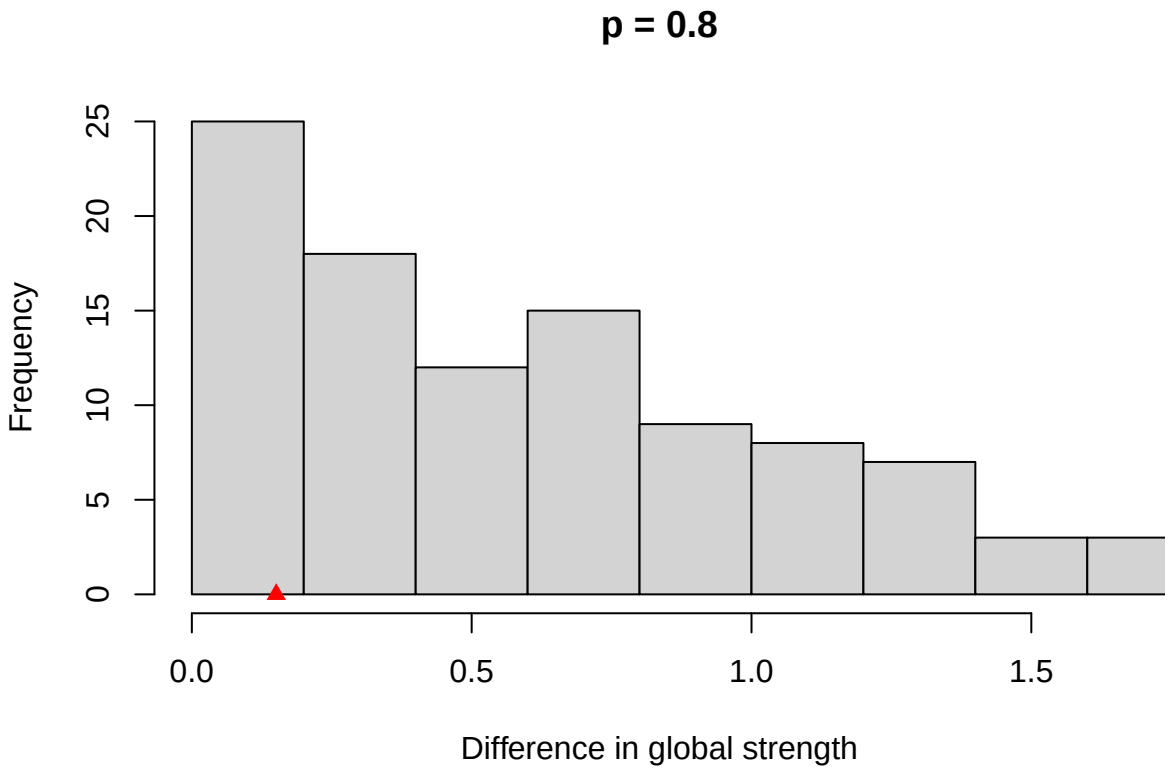


Figure 18
Omnibus Test

```
# Omnibus Test
plot(Comparison2, what = "network")
```

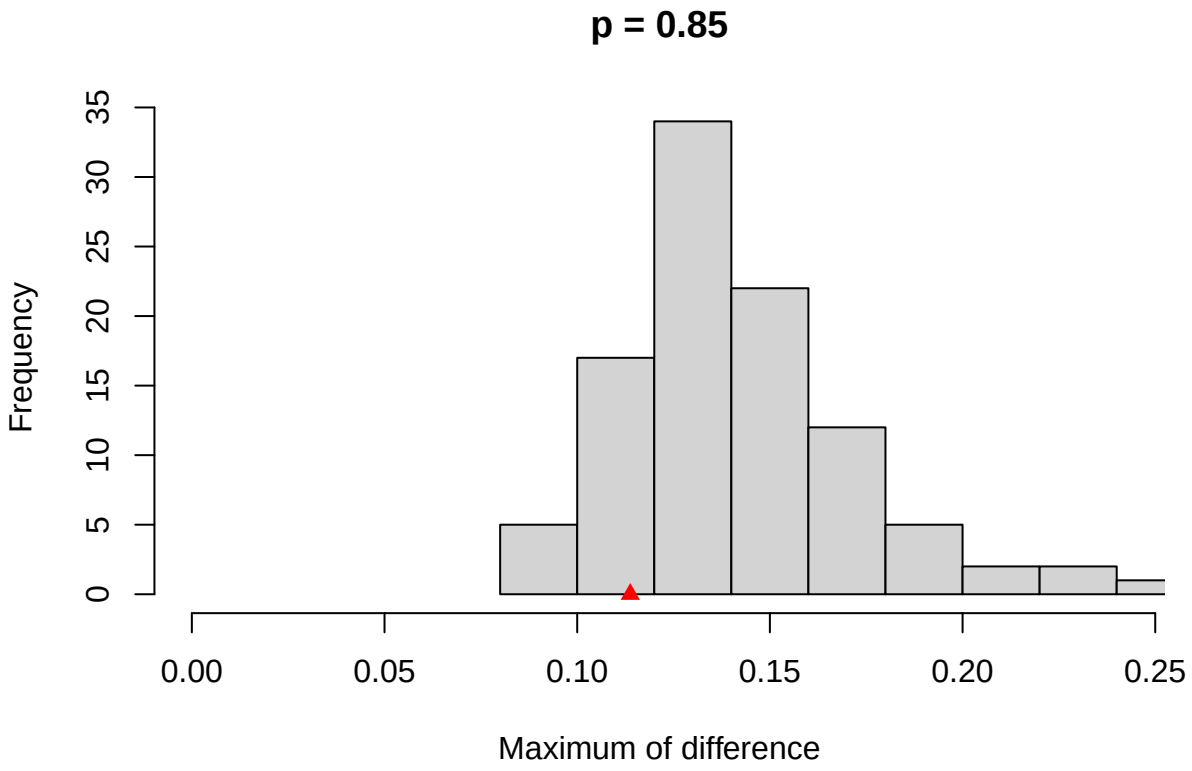


Figure 19
Edge Difference Test Significant Edges

```
#Edge Difference Test
kable(Comparison2$einv.pvals[which(Comparison2$einv.pvals[,3]<.0500),])
```

	Var1	Var2	p-value
234	socommun	rotapple	0.01
293	fechld	sprtprsn	0.00
306	marasian	polviews	0.03
320	sprtprsn	polviews	0.02
358	sprtprsn	happy	0.03