Extended Comment on ‘Statistical Modelling of Citation Exchange between Statistics Journals’

1) Introduction:

In the *Discussion on ‘Statistical Modelling of Citation Exchange Between Statistics Journals’ by Cristiano Varin, Manuela Cattelan and David Firth*, we briefly consider a networks analysis of the statististics journal citation data discussed in the article.

The following is an extended discussion, expanding on the network models and adding visualizations and details of implementation in R.

2) Visualization:

Julyan Arbel writes in a blog post that he is surprised to see no “networks representation” of the citation data presented in the article by Varin, Cattelan, and Firth (2016; Arbel 2015). Plotting the network of journals offers a succinct summary of relationships, clustering and centrality. Of course, our interpretation of such a plot is sensitive to the plotting algorithm, parameters, and any underlying model.

The first figure in Arbel’s post is a network plot prepared using the Gephi software (http://gephi.github.io/). It shows the traffic between journals via the edge weights, although for visual clarity only the top decile of weightiest edges is included. However, the distances between journals are not meaningful and are instead (I believe) optimized for appearance.

Similarly, Figure 1 below presents a thinned down version of the citation network. An edge ($i \to j$) is only visible if j receives at least seven percent of the citations given by i. It shows that several journals are “parents” of some beneath them, but many journals are on the same level. It is evident why *Journal of the Royal Statistical Society, Series B* (JRSS-B), *Annals of Statistics* (AoS), *Biometrika* (Bka) and *Journal of the American Statistical Association* (JASA) are consistently the top four ranked journals, affirming “diffuse opinion within the statistical community” (Varin, Cattelan, and Firth 2016).

Figures ? and ? below provide examples of citation network plots in which distances between journals are model-based. The discussion describes the underlying models and corresponding code.

3) Network Modelling:

The Stigler model estimates “export scores”, u_i, such that c_{ij} is assumed binomially distributed with $E(c_{ij}) = t_{ij} \exp(\alpha_i + \beta_j)$ and $u_i = \alpha_i - \beta_i$, as in ‘quasi-symmetry’ formulation (4). Note that in Varin et al. the data is stored in an 47 x 47 cross-citation matrix C, in which c_{ij} is the number of times journal i is cited *by* journal j excluding self-citations. T is the symmetric total citation matrix, i.e. $t_{ij} = c_{ij} + c_{ji}$ for $i \neq j$ and 0 otherwise. To faciliate modeling in this paper I have transposed the data to fit a standard network framework such that c_{ij} is the number of times journal i cites journal j. This change is reflected in the notation below. In this notation, the ‘quasi-symmetry’ formulation (4) is expressed $E(c_{ij}) = t_{ij} \exp(\alpha_j + \beta_i)$ and $u_i = \alpha_i - \beta_i$.

We can place these assumptions in the context of a valued exponential random graph model (valued ERGM), where edge weights are directed citation counts. (See Krivitsky and Butts (2015) and Krivitsky (2012).) A direct extension of the Stigler model would retain the assumption of binomially distributed citations. However, to facilitate modelling we assume Poisson-distributed citatons with mean c_{ij}, thereby modelling a count instead of a proportion. We include `sender` (β_i) and `receiver` (α_i) effects, so that our assumption is $c_{ij} \sim pois(\lambda_{ij} = exp(\alpha_j + \beta_i))$.

We implement this model using the `latentnet` package in R (Krivitsky and Handcock 2015). The Poisson model is used because binomial families in `latentnet` demand a constant number of trials across dyads. It

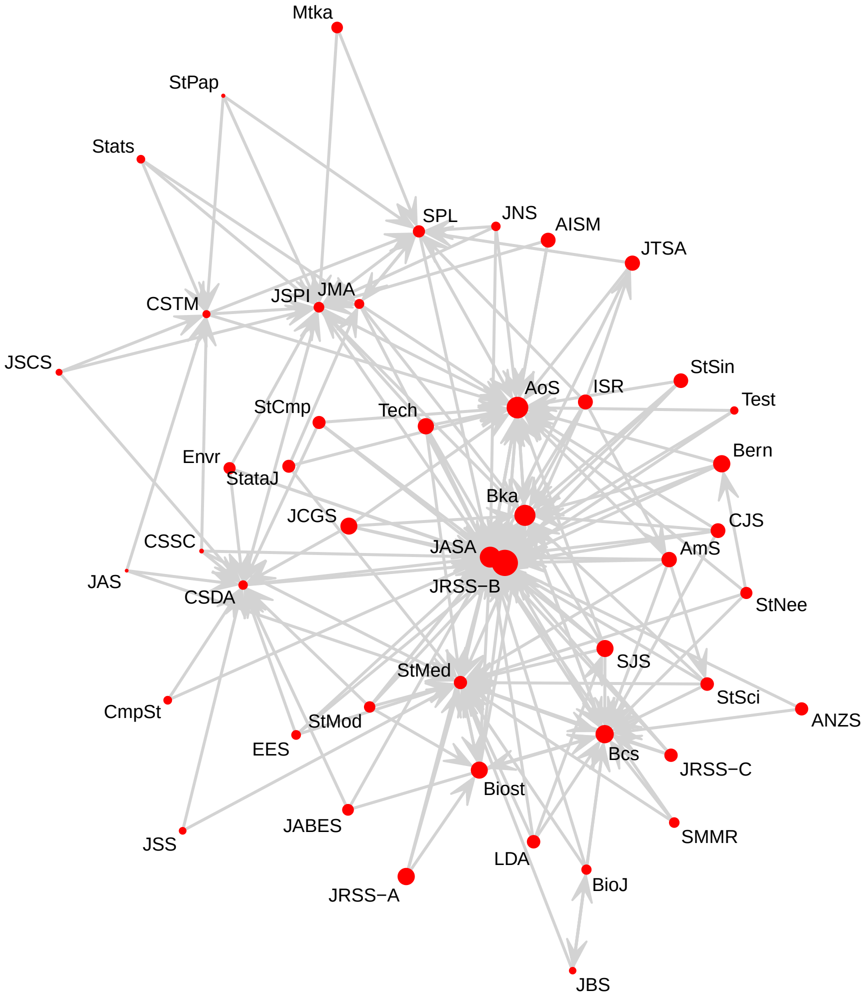


Figure 1: Statistics journal citation network. Edges only pictured if receiver accounts for at least seven percent of sender's total citations sent. Vertex sizes are scaled to export scores under the Stigler model.

may seem natural to use the `ergm.count` (Krivitsky 2015) package for weighted networks instead, but that package does not have `sender` and `receiver` terms implemented (see Section 5.2.5 of Krivitsky (2012)). In addition, this preliminary network model is equivalent to a Poisson GLM (see `glm` code chunk), but using the `latentnet` package sets up later analysis. The model formula is below, `latent.srp1`.

```
latent.srp1 = ergmm(Cnet ~ sender(base=0) + receiver(base=0) - 1, response = "citations",
  family="Poisson.log", control = control.ergmm(pilot.runs=1), seed = 123)
```

The Stigler export scores reported in Table 5 of Varin et al. are highly correlated (.95) with the corresponding estimates from this model ($\alpha_i - \beta_i$). The plot below compares rankings by the two methods, showing their similarity. Differences may be due in most part to the switch from binomial to Poisson expectations. (Code for a plot comparing scores instead of ranks is in the `latent_sr_analysis` code chunk in the markdown version of this document.)

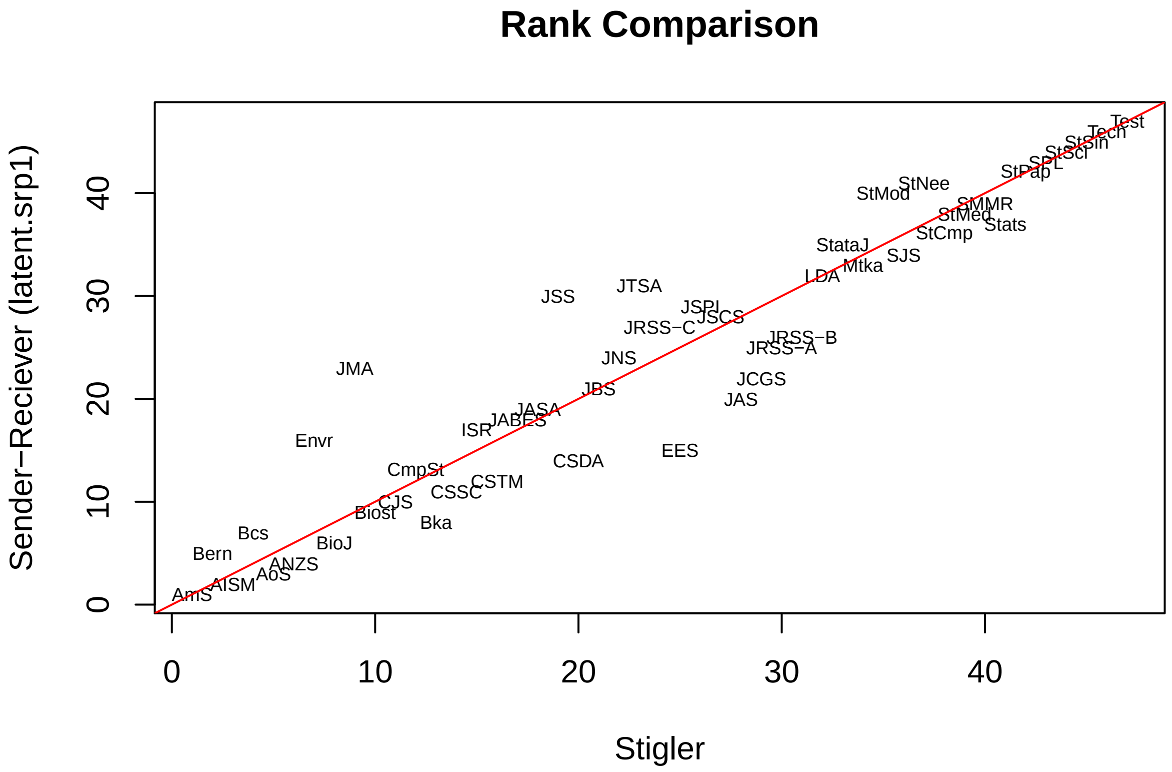


Figure 2: Comparison of rankings by the Stigler and sender-receiver (latent.srp1) model. There is more differentiation towards the middle due to scores being more tightly clustered.

4) Equivalence to GLM:

Varin et al. point out in Section 5.1 that export scores can be estimated with standard GLM software. Indeed, the estimates of export scores in Table 5 of Varin et al. are identical to estimates from a quasibinomial generalized linear model (GLM) with logit link on proportional citation counts (differing only by a constant depending on the constraint imposed). The response is a vector of $\{\frac{c_{ij}}{t_{ij}}\}$ with zeros for diagonal elements. As in the Stigler model we must have a reference journal or zero-sum constraint. The model is implemented in the `glm` code chunk in the markdown version of this document.

5) Network Model Extensions:

A benefit of the network model described above is extensibility, both theoretically and computationally.

We can make minor adjustments to the `latent.srp1` model above, such as including an intercept term for dyad-wise distributions, i.e., $\lambda_{ij} = \exp(\theta + \beta_i + \alpha_j)$. We can also make more significant changes, such as locating journals in latent space and adding cluster labels. These models and corresponding visualizations provide a deeper understanding of the landscape of journals.

5.1) Latent Space Model:

As a preliminary extension, consider the example of a two-dimensional latent space model using the `latentnet` package (Krivitsky and Handcock 2015). This model posits distances between journals as latent variables that affect edge weights (i.e., citation counts). Below, `latent.srp2` adds to `latent.srp1` that journals reside in two-dimensional euclidean space, i.e., $\lambda_{i,j} = \exp(\|Z_i - Z_j\| + \beta_i + \alpha_j)$. (See formulation in Equation 4 of Krivitsky et al. (2009). For background see Hoff, Raftery, and Handcock (2002); Krivitsky and Handcock (2008); Krivitsky and Handcock (2015).)

```
latent.srp2 = ergmm(Cnet~euclidean(d=2) + sender(base=0) + receiver(base=0) - 1,
      response = "citations", family="Poisson.log", seed=123,
      control=ergmm.control(interval=200, sample.size=10000, burnin=100000))
```

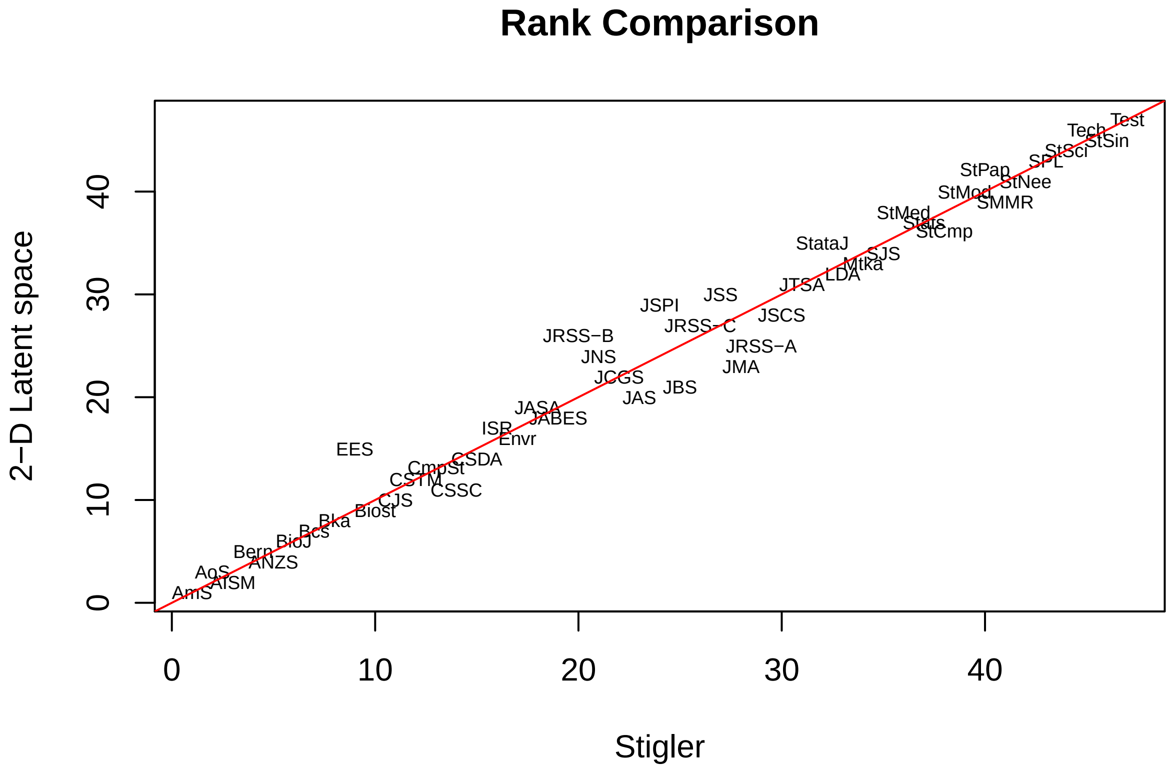


Figure 3: Comparison of rankings by the Stigler and two-dimensional latent space (latent.srp2) model.

Estimates of export scores from the Stigler model are very highly correlated (.99) with the corresponding estimates from this model (receiver minus sender coefficient). The rankings differ only slightly, as shown in Figure ?. (Code for a three-dimensional latent space model is in the `latent_sr3` code chunk and plotted against the Stigler model in the `latent_sr23_analysis` chunk. The correlation is greater than .99)

Below (Figure ?, left) is a plot of estimated journal positions from `latent.srp2`. Node sizes are scaled to receiver minus sender coefficient. The coloring corresponds to the clustering model of Varin et al. (see their Section 3), as do the cluster labels on the right. Although there is no clustering term in our latent space model, the clustering of the authors is fairly well captured. The plot shows how the clusters fit together, and which journals are neighbors. However, we should be careful not to put too much stock in the exact positions. The right-hand plot displays the uncertainty in the positions using a sample of draws from the model.

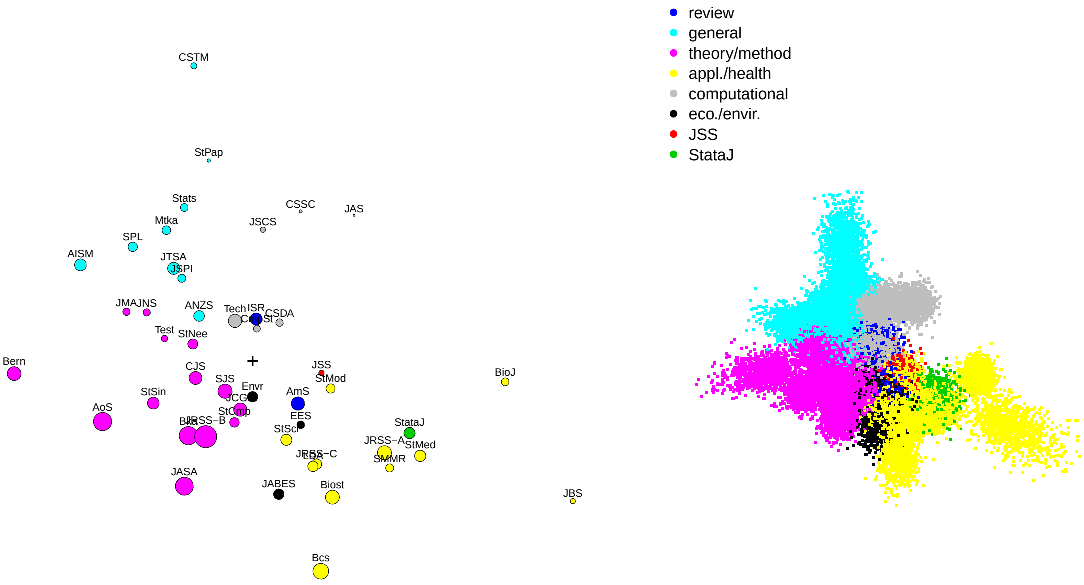


Figure 4: Estimated journal positions from the two-dimensional latent space model. Left: Point estimates with node size scaled to receiver minus sender coefficient. Right: Sample of positions from the model. Colouring is due to the hierarchical clustering of Varin et al.

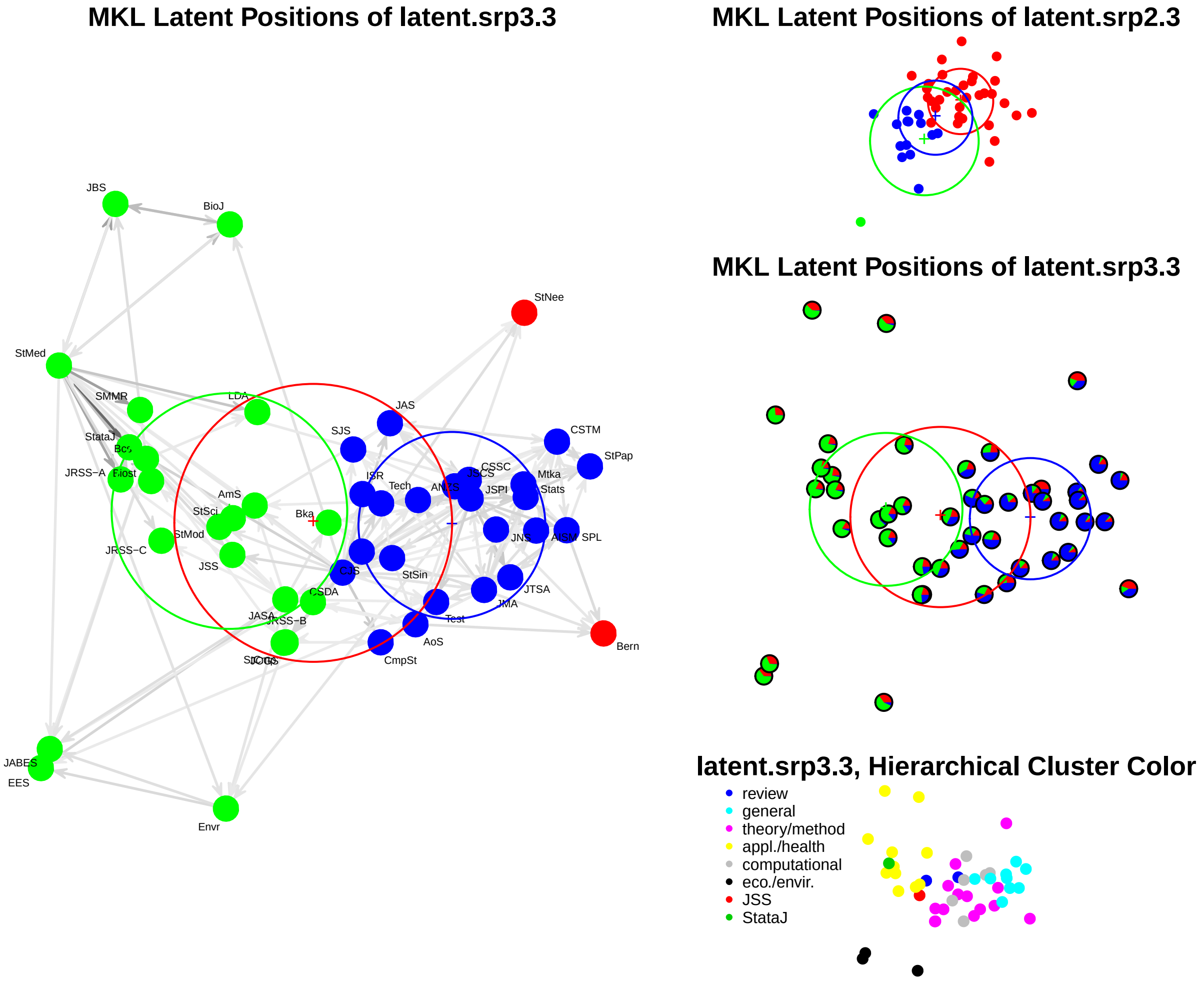
Figure ? gives a visual aid to the observation (see Section 7.2 of Varin et al.) that many journals are not significantly different in rank, and therefore 'grouped' rankings are more appropriate than traditional ordering. We see a periphery of mostly low-ranked journals on the right and a small cluster of leading journals around JRSS-B. Roughly, journals decrease in rank from left to right, but middle-ranked journals are dispersed widely from top to bottom. Centrality does not equate to rank or prestige, as we can see with *Biometrics (BSC)* and *Bernoulli (Bern)* on the edge of the plot.

Further, the two-dimensional latent space model (`latent.srp2`) accounts better for topical connections between journals than the sender-receiver only model (`latent.srp1`). The residuals of `latent.srp2` have range $(-26, 27)$ with standard deviation 3.7, while those of `latent.srp1` have range $(-128, 79)$ with standard deviation 8.7. The largest residual of `latent.srp1` corresponds to citations from Biometrics (Bcs) to Statistics in Medicine (StMed). There is a clear topical connection between those journals that is not captured by their general tendency to import/export citations. In the `latent.srp2` model they are positioned near eachother and that residual drops to 3.

5.2) Latent Space Cluster Model:

We can further extend the model to include a clustering term for a fixed number of clusters. In addition to estimating cluster assignments this provides probabilities of cluster membership for each journal. (See Krivitsky and Handcock (2015), Krivitsky and Handcock (2008)). The resulting model reveals that divisions between clusters are soft and many journals should be thought to straddle two or more clusters.

Figure ? (left) shows the output of a three-cluster latent space model in three-dimensional space (`latent.srp3.3`). For visual clarity, an edge $i \to j$ is only shown if it accounts for at least seven percent of j's total citations received. In comparison, the three-cluster model in two-dimensional space (`latent.srp2.3`, Figure ? top right) does not separate clusters well, as shown by the overlap in the variance circles (radii equals the square root of intracluster variance (Krivitsky and Handcock 2015)). The middle-right plot of Figure ? shows the probabilistic group membership underlying the left-hand plot. Finally, the bottom-right plot uses the `latent.srp3.3` estimated positions, but colors nodes by the hiearchical clustering of Varin et al. The `latent.srp3.3` model agglomerates the hierarchical clusters into two major groups, one that is mostly applications and one that is theoretical, general and computational. The hierarchical 'review' cluster straddles the latent space clusters, reflecting its broad subject matter. (Code for a few additional models and plots is in the `latent_sr??` code chunks in the the markdown version of this document. I was unable to reliably fit a latent space model with six or more clusters for direct comparison to the hierarchical clustering.)



MKL Latent Positions of latent.srp3.3

MKL Latent Positions of latent.srp2.3

MKL Latent Positions of latent.srp3.3

latent.srp3.3, Hierarchical Cluster Color

Figure 5: Comparison of two- and three-dimensional latent space cluster models. (Left) Edges only pictured if sender accounts for at least seven percent of receiver's total citations received.

The scores and ranks calculated from the cluster models are very similar to earlier models. (Code to compare models is in the `compare_clustered` code chunk in the markdown version of this document.) Only a few journal ranks change by more than a few places. We reiterate that ranking differences may not be significant, but reference them for ease of comparison.

6) Network Structure:

The Stigler model offers a method to test for significant difference in the export scores of two journals, but does not give an overall picture of network hierarchy or structure. Along with visualization, descriptive network statistics help to analyze network structure.

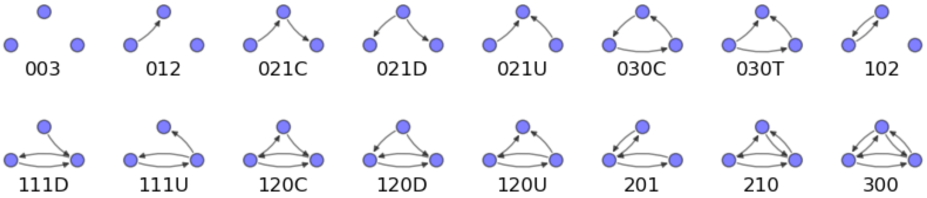


Figure 6: Triad types

The dispersed hierarchy of statistics journals is reflected in the triad census. In the thinned network of Figure 1, 021U triads make up the majority of triads other than the single-edge type, 012, which suggests hierarchy and overall sparsity. However, there are also significant counts of 102 and 111D type triads, indicating more lateral connections. (See Figure ? for triad labels.) There are no cyclic triads of type O30C and 120C, which would indicate a lack of hieracrchy, and very few of the full (300) and near-full (210) types which also contain cycles.

In contrast, in the full network the most popular triads are by far the full (300) or almost-full (210) types, indicative of the network density. The R output below shows all network triad statistics for the thinned down (`Cnet.strong`) and full network (`Cnet`).

```r
summary.statistics(Cnet.strong~triadcensus())
```

```
##  triadcensus.012  triadcensus.102 triadcensus.021D triadcensus.021U
##             3626              262               53             1107
## triadcensus.021C triadcensus.111D triadcensus.111U triadcensus.030T
##               92              189                5               77
## triadcensus.030C  triadcensus.201 triadcensus.120D triadcensus.120U
##                0                3               88                7
## triadcensus.120C  triadcensus.210  triadcensus.300
##                0                5                6
```

```r
summary.statistics(Cnet~triadcensus())
```

```
##  triadcensus.012  triadcensus.102 triadcensus.021D triadcensus.021U
##              998              794              301              471
## triadcensus.021C triadcensus.111D triadcensus.111U triadcensus.030T
##              376             1365              788              441
## triadcensus.030C  triadcensus.201 triadcensus.120D triadcensus.120U
##               47             1557              860              491
## triadcensus.120C  triadcensus.210  triadcensus.300
##              663             3596             3131
```

7) Conclusions:

```r
#looking into how to test for significant difference in "export scores" in the network model

# first, how imporant are the covariances dropped by the quasi-variance method? ####

# GLM Model
# g1 = glm(y~x1-1,family = quasibinomial(link="logit"), weights = as.vector(Tmatrix))
  g1v = vcov(g1) #46 x 46

# var(export score i - export j) is approx the sum of their quasi var
# should technically be var(esi) + var (esj) - 2covar(ij)
# why drop covar when it would be pretty easy to include?

glm.quasi = data.frame()
for (i in 1:45) {
  for (j in (i+1):46) {
    if (i!=j) {
      v = g1v[i,i] + g1v[j,j] - 2*g1v[i,j] #var(score i - score j)
      q = fit.table2$qse[i]^2 + fit.table2$qse[j]^2
      glm.quasi = rbind(glm.quasi, c(v,q))
      }
  }
}
plot(glm.quasi, xlab = "Variance", ylab = "quasi") #very close -> not much lost
cor(glm.quasi[,1],glm.quasi[,2]) #.9993644

# quasi-stigler estimates affirmed "asymptotically normally distributed" p.17

# IF network models coef diffs ASSUMED about normal how does it compare? #####

l1v <- summary(latent.srp1)$pmean$cov #cov matrix from pmean estimate?
  #don't usually use pmean so have to check what's where
  (summary(latent.srp1)$pmean$beta) #reciever, sender
cor((summary(latent.srp1)$pmean$beta)[1:47] - (summary(latent.srp1)$pmean$beta)[48:94], fit.table2$quas

c1 = (summary(latent.srp1test)$pmean$beta)

# find Z scores and component pars (diffs and variances of diffs)
vars = data.frame()
z = data.frame()
diffs = data.frame()
for (i in 1:45) { #45 for comparisons. should be 46.
  for (j in (i+1):46) {
      v = l1v[i,i] + l1v[i+47,i+47] + l1v[j,j] + l1v[j+47,j+47] - 2*l1v[i,i+47] - 2*l1v[j,j+47] - 2* (l
      q = fit.table2$qse[i]^2 + fit.table2$qse[j]^2
      diff.net = (c1[i] - c1[i+47] - c1[j] + c1[j+47])
      diff.quasi = fit.table2$quasi[i] - fit.table2$quasi[j]
      diffs = rbind(diffs, c(diff.net, diff.quasi))
      z1 = diff.net / sqrt(v)
      z2 = diff.quasi / sqrt(q)
      vars = rbind(vars, c(v,q))
      z = rbind(z,c(z1, z2))
  }
}
```

```
# look at results
plot(z, xlab = "network", ylab = "stigler",
     xlim = c(-5,5), ylim = c(-5,5))
abline(v = c(-2,0,2), h = c(-2,0,2), col = c("red","green","red"))
cor(z[,1],z[,2]) #.965. strongly correlated but still  many instances where stigler z would be close to
 # remember this model is POISSON while quasi-stigler is quasiBINOMIAL

plot(vars, xlab = "Variance from network model", ylab = "quasi")
#still high cor, .96, but fans
abline(a=0.007863, b=1.734556, col = "red")
plot(diffs, xlab = "diff.net", ylab ="diff.stigler") #diffs are also pretty similar

plot(diffs[,1]/sqrt(vars[,1]), diffs[,2]/sqrt(vars[,2]))
lm(diffs)

#example on p. 20. #8 , 19
i = 8; j = 19
bka = (summary(latent.srp1)$pmean$beta)[i] - (summary(latent.srp1)$pmean$beta)[i+47]
jasa = (summary(latent.srp1)$pmean$beta)[j] - (summary(latent.srp1)$pmean$beta)[j+47]
jb = l1v[i,i] + l1v[i+47,i+47] + l1v[j,j] + l1v[j+47,j+47] - 2*l1v[i,i+47] - 2*l1v[j,j+47] - 2* (l1v[i,
#set i = 8, j = 19 and find v from above: 0.003514869
(bka - jasa)/sqrt(jb) #why not agreeing with exact z stat of .31

# an alternate bootstrap approach? ####

# sample n models fits and compare ri - si - (rj - sj)? <- computationally heavy
# if we run the model again the recievers - senders might all shift by some amount. need to put in a co
    # that doesn't affect predictions bc prediction is by reciever + sender (constants cancels out)
#look at rank instead? still a slow way to do it though
#variances change a bit, but don't shift
 #  [plot(diag((summary(latent.srp1))$pmean$cov),diag((summary(latent.srp1test))$pmean$cov)) ]

# END ####
```

```
http://arxiv.org/pdf/physics/0505169v4.pdf
http://deepblue.lib.umich.edu/bitstream/handle/2027.42/60774/eleicht_1.pdf?sequence=1&isAllowed=y
http://phys.org/news/2009-12-algorithm-sports-teams-google-pagerank.html
http://www.phys.utk.edu/sorensen/ranking/Documentation/Sorensen_documentation_v1.pdf
```

References

Arbel, Julyan. 2015. "Statistics Journals Network." https://statisfaction.wordpress.com/2015/04/16/statistics-journals-network/.

Hoff, Peter D., Adriean E. Raftery, and Mark S. Handcock. 2002. "Latent Space Approaches to Social Network Analysis." *Journal of the American Statistical Association* 97 (460): 1090–1098.

Krivitsky, Pavel N. 2012. "Exponential-Family Random Graph Models for Valued Networks." *Electronic Journal of Statistics* 6: 1100–1128.

———. 2015. *ergm.Count: Fit, Simulate and Diagnose Exponential-Family Models for Networks with Count Edges*. http://CRAN.R-project.org/package=ergm.count: The Statnet Project.

Krivitsky, Pavel N., and Carter T. Butts. 2015. *Modeling Valued Networks with Statnet.* http://statnet.csde.washington.edu/wor
The Statnet Development Team.

Krivitsky, Pavel N., and Mark S. Handcock. 2008. "Fitting Position Latent Cluster Models for Social Networks with Latentnet." *Journal of Statistical Software* 24 (5).

———. 2015. *latentnet: Latent Position and Cluster Models for Statistical Networks.* 2.7.1. http://CRAN.R-project.org/package=latentnet: The Statnet Project.

Krivitsky, Pavel N., Mark S. Handcock, Adrian E. Raftery, and Peter D. Hoff. 2009. "Representing Degree Distributions, Clustering, and Homophily in Social Networks with Latent Cluster Random Effects Models." *Social Networks* 27 (5): 417–428.

Varin, Cristiano, Manuela Cattelan, and David Firth. 2016. "Statistical Modelling of Citation Exchange Between Statistics Journals." *Journal of the Royal Statistical Society A* 179 (1): 1–33.