Symptom network models in depression research

van Borkulo, Claudia Debora

APPENDIX C

SUPPLEMENTARY INFORMATION TO CHAPTER 9

Adapted from:

Supplementary Information to: Van Borkulo, C. D., Wichers, M.C., Boschloo, L, Schoevers, R.A., Kamphuis, J. H., Borsboom, D. & Waldorp, L. J. (2015). The contact process as a model for predicting network dynamics of psychopathology. Manuscript submitted for publication.

This Supplementary Information contains seven sections. Section C.1 consists of derivations of transition probabilities. Section C.2 contains the validation study to assess performance of graphicalVAR. In Section C.3, we provide R code to simulate data according to the contact process model. Section C.4 contains results of a comparison of the Fisher information variance and the sample variance. Sections C.5 and C.6 show figures that are not displayed in Chapter 9. Finally, in Section C.7 we explain the contruction of the t-tests and the resulting quality of the test statistic.

C.1 Derivations

C.1.1 Transition probabilities

The rates of the independent Poisson processes in (2.3) of the main paper can be equivalently characterized by the transition matrix (Norris, 1997, Theorem 2.4.3). As the number of infected nodes increases by 1 at rate $\lambda k_s(V)$ and decreases by 1 at rate μ, the generator matrix $Q_s(x)$ of the two-state Markov process can be defined as (Brzezniak & Zastawniak, 2000; Grimmett, 2010; Singer, 1981)

$$Q_s(x) = \begin{pmatrix} -\lambda k_s(x) & \lambda k_s(x) \\ \mu & -\mu \end{pmatrix} \tag{C.1}$$

This defines a system of differential equations with Kolmogorov forward equations $\frac{d}{ds}P_s(x) = P_s(x)Q_s(x)$, in which $P_s(x) = \exp[sQ_s(x)]$ is the transition matrix, and $\exp[sQ_s(x)] = \sum_{j=0}^{\infty} Q_s^j / j$ (Norris, 1997). For our two-state process ($j = 0, 1$), we need to solve the forward equations with the elements $p_s^{jj}(x)$. Because $P_s(x)$ is a stochastic matrix, in which the sum of each row equals 1, we only need to solve the differential equations

$$\begin{aligned} \frac{d}{ds}p_s^{01}(x) &= -\lambda k_s(x)(1 - p_s^{01}(x)) + \mu p_s^{01}(x) \\ \frac{d}{ds}p_s^{10}(x) &= \lambda k_s(x) p_s^{10}(x) - \mu(1 - p_s^{10}(x)), \end{aligned}$$

since $p_s^{00}(x) = 1 - p_s^{01}(x)$ and $p_s^{11}(x) = 1 - p_s^{10}(x)$. The resulting solutions are

$$\begin{aligned} p_s^{01}(x) &= \frac{\lambda k_s(x)}{\lambda k_s(x) + \mu} + \left(p_0^{01}(x) - \frac{\lambda k_s(x)}{\lambda k_s(x) + \mu} \right) \exp[-(\lambda k_s(x) + \mu)s] \\ p_s^{10}(x) &= \frac{\mu}{\lambda k_s(x) + \mu} + \left(p_0^{10}(x) - \frac{\mu}{\lambda k_s(x) + \mu} \right) \exp[-(\lambda k_s(x) + \mu)s]. \end{aligned} \tag{C.2}$$

Here, the first part on the right hand side is the equilibrium part, while the second part is sometimes referred to as the deviation from equilibrium, which decreases exponentially with s. Therefore, we use the equilibrium part of the solution and obtain the transition probability matrix

$$P_s(x) = \begin{pmatrix} 1 - p_s(x) & p_s(x) \\ q_s(x) & 1 - q_s(x) \end{pmatrix}, \tag{C.3}$$

where $p_s(x) = p_s^{01}(x)$ and $q_s(x) = p_s^{10}(x)$ and

$$p_s(x) = \frac{\lambda k_s(x)}{\lambda k_s(x) + \mu}, \quad q_s(x) = \frac{\mu}{\lambda k_s(x) + \mu}. \tag{C.4}$$

We assume that in each time segment $[s, s+h)$, with $h > 0$, the underlying process is right-continuous, meaning that when a node is, e.g., in a healthy state at time s, it stays in that state until time $s+h$; then it switches to an infected state. The holding time is the time between events in which the state of the nodes is assumed to be invariant and exponentially distributed. As a result we can use the discrete time Markov chain $\xi_i(x)$ with transition probabilities $p_i(x)$ and $q_i(x)$ for $i = 1, 2, \ldots$

C.2 Validation study graphicalVAR

C.2.1 Design

We assessed the performance of graphicalVAR in a simulation study. Time series data were simulated by generating networks (i.e., *true* networks) with a similar number of nodes as our real data (i.e., 10). We followed the steps in Yin and Li (2011) to simulate temporal and contemporaneous networks, using a constant of 1.1 and making 50% of the edges negative. Number of simulated time points was 50, 100, 150, 200, and 500, density of the temporal network was set to .1, .3, and .5 and density of the contemporaneous network was set to .3. We investigated the temporal network. The quality of network estimation was assessed by inspecting correlations between the true and estimated network parameters, the sensitivity (i.e., true positive rate), and specificity (i.e., true negative rate).

C.2.2 Results

Figure C.1 shows that with only 50 time points, true and estimated networks differ somewhat. However, the average correlation remains high ($M = .91, SD = .07$).

With increasing sample size, the average correlation increases up to .98 ($SD = .02$) for the largest sample size. More detailed information about performance of graphicalVAR is provided by sensitivity and specificity. Sensitivity is overall high ($M = .88, SD = .15$) but varies across densities. With sample sizes of 200 and larger, sensitivity increases to .94 on average ($SD = .10$), and to .98 ($SD = .04$) when true networks were more dense (less sparse). Across all conditions, specificity is moderate to high ($M = .79, SD = .13$), indicating an acceptable false positive rate (i.e., most edges that are estimated are present in the true network). To conclude,

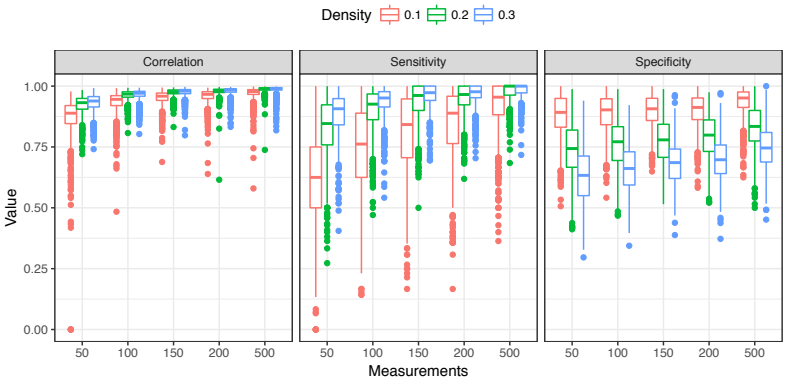

FIGURE C.1. Performance of graphicalVAR. Correlation (left) between the true and the estimated temporal network , sensitivity (middle), and specificity (right) are displayed of simulated temporal networks with densities of .1 (red), .3 (green) and .5 (blue).

graphicalVAR demonstrates to be an acceptable method to estimate graphical models from continuous data. The simulation study indicates that with sample sizes of 100 and more, the estimated and true network show high concordance. Correlations and sensitivity are high and sensitivity is moderate and increasing with sample size. Specificity is moderate tot high across all conditions, indicating acceptable levels of false positives among the estimated edges. Thus, graphicalVAR exhibits high sensitivity. However, sensitivity can be moderate for the less densely connected temporal networks, for the benefit of a high specificity.

C.3 R code for the simulation process

This section contains the annotated R code of function SimFunction() to simulate data according to the contact process ().

```
SimFunction=function(l,m,nobs,adj)
{
  # l=lambda
  # m=mu
  # nobs= number of observations
  # adj= unweighted adjacency matrix
  # note: number of events in time interval (between observations) is poisson
      distributed

  x=ncol(adj)
  y=sum(rowSums(adj)>0) # number of nodes in the network
  z=rowSums(adj) # number of edges per node
  w=rep(0,x)

  # Generate starting point (there has to be at least one infected variable).
      The probability of being infected from the start is determined on
      empirical data of patients.
  for(i in 1:(x)){
    if(z[i]!=0)w[i]=sample(0:1,1,prob=c(5/16,11/16))
  }

  r=c(1,rpois(nobs*3,(l+m))) # nobs*3 in order to simulate enough events
  t=sum(r)
  pois <- rpois(nobs, l)
  if (pois[1]==0) pois[1]=pois[1]+1
  obs <- cumsum(pois)
  if (obs[length(obs)]<t) r=r else r=c(1,rpois(nobs*10,(l+m))) # to prevent that
      there are too few events to simulate the number of observations needed.
  t=sum(r) # if r changed, than t is updated here

  data=matrix(c(w,rep(0,x*(t-1))),length(w),x,byrow=T) # Matrix with the data.
      Contains the starting point in row 1 and is empty (zeros) for all other
      rows.
```

```
# let the "fully observed" process run over time
for(j in 2:t)
{
  transitionprob <- rep(NA,x)
  for (i in 1:x)
  {
    w=which(adj[i,]==1)  # neighbors of node i
    k=sum(data[j-1,w]) # number of infected neighbors at t-1
    transitionprob[i] <- l*k/(l*k+m) # This is P and Q=1-P. Transition
        probabilities according to Brzezniak (2000), taking the network
        topology into account.
  }
  data[j,]=data[j-1,] # Copying the previous time point, to change hereafter
      the node that changes

  # Randomly draw the node that will change. Depending on whether that node is
      on or off, it is decided whether it will change or not (with a to be
      calculated probability).
  # When nothing changes, a new node is drawn. This is repeated until a change
      occured.

  # If there are infected symptoms, proceed. Else, the process has died out.
  if(any(data[j-1,]==1)){
    count=0
    while (count<1)
    {
      s=sample(x,1) # Draw a random node. If this node infected, perform the
          recovery procedure. If the node is recovered, perform the infection
          procedure
      if (data[j-1,s]==0) # Procedure for infection
      {
        if(runif(1)<transitionprob[s]) # This refers to the probability to be
            infected, given the node is recovered
        {
          data[j,s]=1
          count=count+1
        } else data[j,s]=0 # the node stays recovered and a new node must be
            randomly drawn
      } else { # Procedure for recovery
        if(runif(1) < (1-transitionprob[s])){ # This refers to the probability
            to recover, given the node is infected (1-P)
          data[j,s]=0
```

```
            count=count+1
          } else data[j,s]=1
        }
      }

    } else (j=t)
  }

  ## We now have the process as if it was fully observed. Next, we do the "
      observations"
  dataobs=matrix(NA,nobs,x,byrow=T)
  for (i in 1:nobs) dataobs[i,] <- data[obs[i], ]

  results=list(data = data, dataobs = dataobs)
  return(results)
}
}
```

C.4 Variance

To assess the quality of the estimate $\hat{\rho}$, we need the variance of $\hat{\rho}$. Since the variance is unknown, we will show how the variance can be estimated. First, we consider the most common estimate of the variance based on the Fisher information.

C.4.1 Fisher information variance

It is derived from the second-order derivatives of the loglikelihood (equation 3.10 in the main article) with the delta method. These second-order derivatives are represented in the Hessian $H_t(\lambda,\mu)$ as

$$H_t(\lambda,\mu) = \begin{bmatrix} -\frac{U_t}{\lambda^2} & 0 \\ 0 & -\frac{D_t}{\mu^2} \end{bmatrix}, \tag{C.5}$$

where U_t is the number of upward jumps and D_t is the number of downward jumps (see equation 3.7 of the main article). Taking the negative of the inverted Hessian (the observed Fisher information matrix) results in the covariance matrix. The Fisher information variance of $\hat{\lambda}$ and $\hat{\mu}$ is

$$\hat{\sigma}^2_{\hat{\lambda}_F} = \frac{\hat{\lambda}^2}{U_t}, \quad \hat{\sigma}^2_{\hat{\mu}_F} = \frac{\hat{\mu}^2}{D_t}. \tag{C.6}$$

Although the Fisher information variance is the usual way to calculate the variance, Fiocco and van Zwet (2003) stated that the sample variance, which is based on the variation over nodes of the observed process, is a better estimate. Therefore, we consider the sample variance as a second estimator.

C.4.2 Sample variance

The sample variance depends on the estimates of the parameters for each node individually, as opposed to one single estimate for the whole network. The sample variance can, therefore, be estimated as

$$\begin{aligned}\hat{\sigma}^2_{\hat{\lambda}_S} &= \frac{1}{|V|}\sum_{x\in V}(\hat{\lambda}(x)-\bar{\lambda})^2,\\ \hat{\sigma}^2_{\hat{\mu}_S} &= \frac{1}{|V|}\sum_{x\in V}(\hat{\mu}(x)-\bar{\mu})^2,\\ \hat{\sigma}^2_{\hat{\rho}_S} &= \frac{1}{|V|}\sum_{x\in V}(\hat{\rho}(x)-\bar{\rho})^2,\end{aligned} \tag{C.7}$$

in which $|V|$ is the number of variables and $\bar{\lambda}$, $\bar{\mu}$ and $\bar{\rho}$ are the means of the estimates per node $\hat{\lambda}(x)$, $\hat{\mu}(x)$ and $\hat{\rho}(x)$.[1] The node-specific estimates are defined as

$$\hat{\lambda}(x) = \frac{U_t(x)}{A_t(x)}, \qquad \hat{\mu}(x) = \frac{D_t(x)}{B_t(x)} \tag{C.8}$$

and, consequently,

$$\hat{\rho}(x) = \frac{\hat{\lambda}(x)}{\hat{\mu}(x)} = \frac{U_t(x)B_t(x)}{A_t(x)D_t(x)}. \tag{C.9}$$

C.4.3 Comparing variance estimates

Fiocco and van Zwet (2004) stated that the sample variance is a better estimate than the Fisher information variance. We investigated both Fisher information and sample variance, as in equation C.6 and C.7, by comparing them to the Monte Carlo variance. The Monte Carlo variance is the variance of $\hat{\lambda}$ and $\hat{\mu}$ of simulated

[1] A model with separate estimates per node is an extension of the model used in this study. Model comparison of the original and alternative model using real data could reveal which model better fits the data. The goodness-of-fit measure to use for model comparison could be, for example, AIC_c or BIC. Simulating data under the null model (i.e., with only one estimate for the whole network) and the alternative model (i.e., estimates for each node) could reveal which fit measure is preferred.

data. The ratio between the estimated variance and the Monte Carlo variance should be approximately 1.

We computed the Fisher information variance σ_F^2 and the Monte Carlo variance σ_{MC}^2 for each of the 8 data sets and computed their ratio $\sigma_F^2 / t\sigma_{MC}^2$, where t is the number of observations of the simulation. It follows from Figure C.2, that the Fisher information variance is not a good estimate or the variance across all conditions (results for networks with 50% and 100% replacement have similar results, not shown here); the Fisher information variance clearly underestimates the variance.

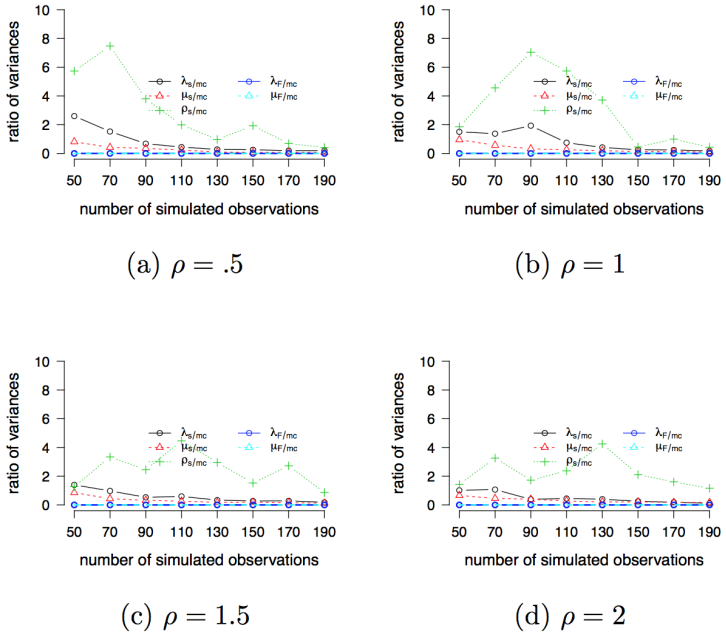

FIGURE C.2. Ratio of the Fisher information variance and the Monte Carlo variance (F/mc) of $\hat{\lambda}$ and $\hat{\mu}$ and the ratio of the sample variance and the Monte Carlo variance (s/mc) of $\hat{\lambda}$, $\hat{\mu}$, and $\hat{\rho}$ as a function of the number of observations. For different values of ρ (a through d) with pure lattice networks.

C.5 Violin plot of estimates of ρ not shown in Chapter 9

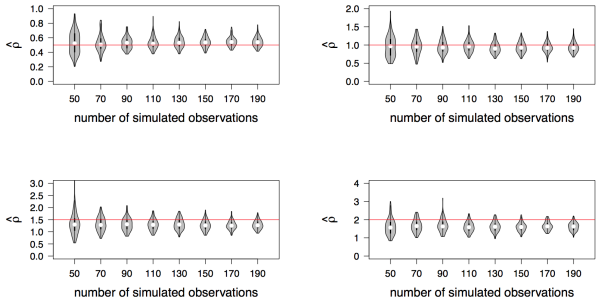

FIGURE C.3. Violin plots of the estimates of ρ. Distributions of estimates of 100 simulated data sets with pure lattice structure, increasing amount of observations (50, 70,..., 190) and with $\rho = .5$ (a), $\rho = 1$ (b), $\rho = 1.5$ (c), and $\rho = 2$ (d). The red lines indicate the true value of ρ, with which the data was simulated.

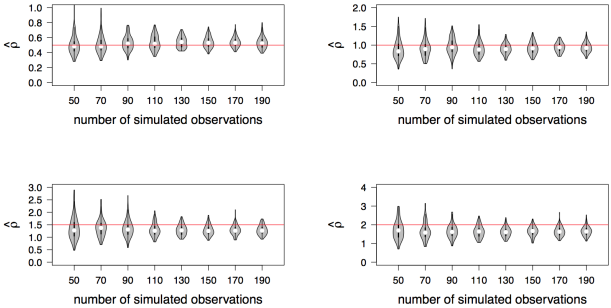

FIGURE C.4. Violin plots of the estimates of ρ. Distributions of estimates of 100 simulated data sets with 50% replacement networks, increasing amount of observations (50, 70,..., 190) and with $\rho = .5$ (a), $\rho = 1$ (b), $\rho = 1.5$ (c), and $\rho = 2$ (d). The red lines indicate the true value of ρ, with which the data was simulated.

C.6 Plots of sample variances not shown in Chapter 9

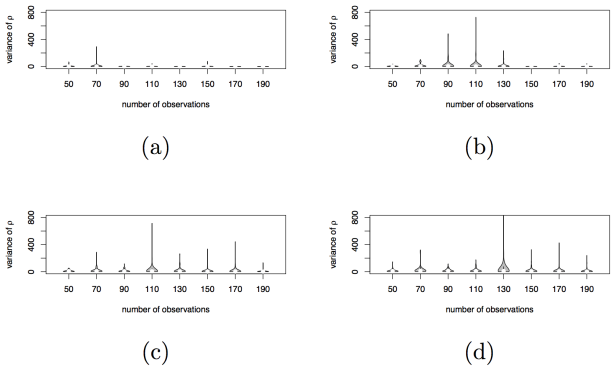

FIGURE C.5. Sample variances of simulated data based on a network with pure lattice structure and (a) $\rho = .5$, (b) $\rho = 1$, (c) $\rho = 1.5$, and (d) $\rho = 2$.

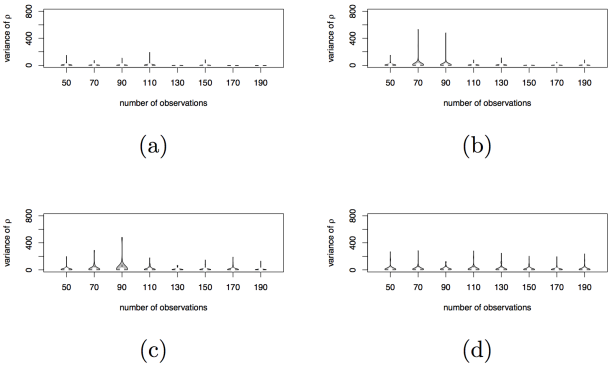

FIGURE C.6. Sample variances of simulated data based on a network with 50% replacement structure and (a) $\rho = .5$, (b) $\rho = 1$, (c) $\rho = 1.5$, and (d) $\rho = 2$.

C.7 Statistical testing

We constructed two t-tests. One that tests $\hat{\rho}$ against the percolation threshold of 1 (a one-sample t-test) and one that compares two values of $\hat{\rho}$ of two different systems (an independent two-sample t-test).

With a one-sample t-test, $\hat{\rho}$ can be tested against the percolation threshold. When $\hat{\rho}$ is larger than 1, the symptoms in the network will remain active indefinitely. The statistic for this one-sample t-test is defined as

$$t = \frac{\hat{\rho} - 1}{\sqrt{\hat{\sigma}^2_{\rho_s} / n}}, \tag{C.10}$$

in which n is the number of nodes. In this case, since one person is compared to a fixed value (1), $\hat{\sigma}^2_{\rho_s}$ is the sample variance of the person under consideration estimated as in equation (C.7). Since the variance of $\hat{\rho}$ is based on the estimates per node, for the t statistic, it has to be divided by n. The number of degrees of freedom is $n-1$.

With a two-sample t-test, $\hat{\rho}$ of two persons can be compared. The statistic for the independent two-sample t-test is defined as

$$t = \frac{\hat{\rho}_1 - \hat{\rho}_2}{\sqrt{\hat{\psi}^2_{\rho_s} / n}}, \tag{C.11}$$

where $\hat{\rho_1}$ and $\hat{\rho_2}$ are the estimates of the percolation indicators of person 1 and 2 respectively, and n is the pooled number of observations the estimates of ρ is based on (the number of nodes in both networks). $\hat{\psi}^2_{\rho_s}$ is the sample variance estimated as in equation (C.7). The number of degrees of freedom is based on the number of variables of both samples ($n1 + n2 - 2$).

C.7.1 Quality of test statistic

With our simulated data we can check whether the distribution of the test statistics is normal. For the one-sample t-test, we used the data set that is simulated with $\rho = 1$, meaning that the null-hypothesis ($\rho = 1$) is true. This data set contains simulations with 50 through 190 observations, each with 100 replications. For all 8 $\times$ 100 simulations, we tested whether $\hat{\rho} = 1$ ($df = 21$). To investigate the distribution of the two-sample t-test, we tested half of the data set that is simulated with $\rho = 1$ against the other half. Since we know that $\rho = 1$ for all simulations contained

in this data set, we know that the null hypothesis is true: $\hat{\rho}_1 = \hat{\rho}_2$ ($df = 42$). To compare the empirical distribution of both t-statistics, we drew 800 samples from a t distribution with $df = 21$ and 400 samples from a t distribution with $df = 42$, respectively.

The density plots in Figure C.7 show that both empirical distributions are normal. The distribution of the one-sample t-test is only slightly skewed to the left. Both empirical distributions have wider tails than the theoretical distributions, indicating that the type I error will be larger.

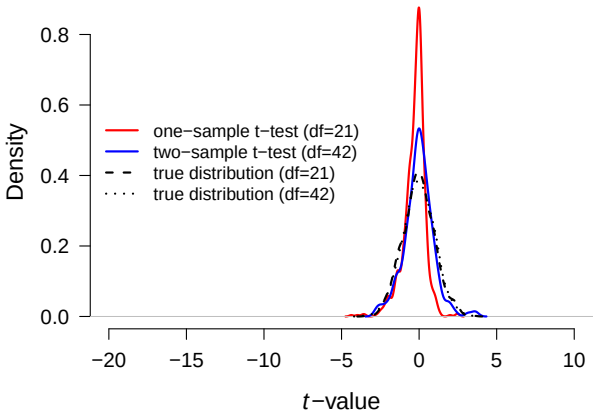

FIGURE C.7. Density plots of t-values of the one-sample t-test against the value 1 (red line), of the two-sample t-test (blue line), the true distribution of t-values with 21 degrees of freedom as in the one-sample t-test (black dashed line), and the true distribution of t-values with 42 degrees of freedom as in the two-sample t-test (black dotted line). Data were simulated with $\rho = 1$ and networks with 100% replacement. Data simulated with pure lattice structure and 50% replacement show similar results.