Supplementary Information

A new method for constructing networks from binary data

Claudia D. van Borkulo[1,2], Denny Borsboom[1], Sacha Epskamp[1], Tessa F. Blanken[1], Lynn Boschloo[2], Robert A. Schoevers[2] and Lourens J. Waldorp[1]

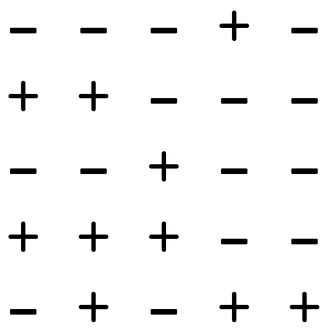
[1]Interdisciplinary Center Psychopathology and Emotion regulation, University Medical Center Groningen, University of Groningen

[2]Department of Psychology, Psychological Methods, University of Amsterdam

This Supplementary Information contains two sections. In the Supplementary Methods section, we provide a more detailed description of the Ising model and a worked example on how to compute probabilities of certain states with this model. Also, the theoretical background of the estimation procedure is described more elaborately. In the Supplementary Results section, we describe the results of an additional analysis on the validity of our proposed method eLasso.

Supplementary Methods

In this section, we provide an introduction to the Ising model on which eLasso is based. The original model was based on magnetic behaviour of metals[1]. In the easiest case, such models operate on the behaviour of small dipoles or spins of a ferromagnet, which are arranged as in Supplementary Figure S1. This two-dimensional representation is often called a grid or lattice. An individual dipole can be either in a "spin up" (+ 1) or a "spin down" position (- 1)[2]; in an alternative variant, which we will use here, these variables are scored 0 or 1. Objects, or variables, in the Ising model can interact with each other but interactions are subject to an important restriction: they can only influence direct *neighbours*. That is, taking neurons as an example, a firing neuron can only



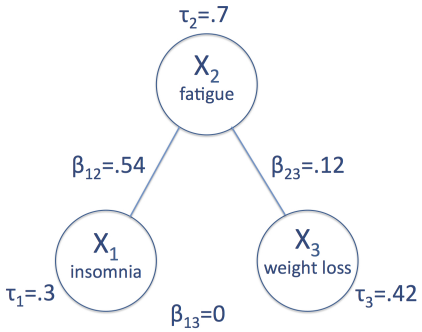
Supplementary Figure S1: Graphical representation of the Ising model.

transmit information to a connected neuron.

Although the Ising model was used to explain ferromagnetism, it generalises to all kinds of *objects* in a network that can be in two states: a voter who can be "pro" or "con, a neuron that can "fire" or "not fire", or a person that can be "infected" or "not infected" by some virus[2]. When applying this model to psychopathology, variables can be symptoms of a disorder, which can be either "present" or "absent". Now, suppose we have p variables collected in the set of nodes V. Then there are 2^p possible configurations of the system (in a basic application, these would for instance be all possible item response patterns). Suppose we have three symptoms of major depressive disorder (MDD) according to the DSM-IV-TR: "insomnia", "fatigue", and "significant unintentional weight loss or gain"[3]. Then there are $2^3 = 8$ possible configurations: (0, 0, 0), (1, 0, 0), (1, 1, 0), and so on. A system is inclined to move to or persist in the most favourable configuration possible. For example, if two nodes (say, anxiety and depressed mood) are positively connected, then a configuration in which one is present but the other is not (e.g., an individual who is anxious but not sad) is less consistent with the model structure than one in which both are present or both are absent. Thus, the configuration that is most consistent with the model has the highest probability of occurring. In the Ising model, consistency of a configuration depends on the Hamiltonian function $H(x)$:

$$H(x) = -\sum_{j \in V} \tau_j x_j - \sum_{(j,k) \in E} \beta_{jk} x_j x_k, \tag{1}$$

where V is the set of nodes and E is the set of edges, τ_j is the threshold of symptom x_j, β_{jk} is



Supplementary Figure S2: A hypothetical example network of three symptoms: x_1 (insomnia or sleeping too much), x_2 (fatigue and, or loss of energy), and x_3 (significant unintentional weight loss or gain). β_{12} and β_{23} are the connection strengths (interaction parameters). Since there is no connection between x_1 and x_3, $\beta_{13} = 0$.

the interaction strength between symptoms j and k, and x_j (and x_k) can assume one of two values $\{0, 1\}$[2,4]. The threshold of a symptom, τ_j, indicates the autonomous disposition of the symptom to take the value one, i.e., it describes the probability of that value in the absence of any influences of neighbouring symptoms. Consequently, when this threshold is positive, the symptom tends to be present. This state (with value 1) is, in this case, preferred over the absent state (with value 0), since it lowers the energy. On the other hand, a negative threshold indicates that the symptom, taken by itself, has a disposition to be absent. The interaction strength between two symptoms, β_{jk}, indicates how symptoms influence each other: when $\beta_{jk} > 0$, symptoms will prefer to be in the same state, whereas $\beta_{jk} < 0$ indicates that symptoms will prefer to be in different states. The sum of the interactions runs over all existing connections (edges) in the set E. We define Θ to be a matrix (p by p), containing τ_j on the diagonal and β_{jk} on the off-diagonal.

As soon as we know the structure and parameters of the network, it is easy to calculate the energy of a configuration. Suppose our three symptoms have a structure and parameters as depicted in Supplementary Figure S2. For each possible configuration, the Hamiltonian is given in Supplementary Table S1.

As stated before, the lower the value of the Hamiltonian function for a certain configuration,

Supplementary Table S1: Example of how to calculate the Hamiltonian of all possible configurations of a network with three nodes.

config.	insomnia	fatigue	weight loss	probability	Hamiltonian
1 1 1	present	present	present	P(1 1 1)	$-(\tau_1+\tau_2+\tau_3)-(\beta_{12}+\beta_{23}+0)$
1 1 0	present	present	absent	P(1 1 0)	$-(\tau_1+\tau_2+0)-(\beta_{12}+0+0)$
1 0 1	present	absent	present	P(1 0 1)	$-(\tau_1+0+\tau_3)-(0+0+0)$
1 0 0	present	absent	absent	P(1 0 0)	$-(\tau_1+0+0)-(0+0+0)$
0 1 1	absent	present	present	P(0 1 1)	$-(0+\tau_2+\tau_3)-(0+\beta_{23}+0)$
0 1 0	absent	present	absent	P(0 1 0)	$-(0+\tau_2+0)-(0+0+0)$
1 0 1	present	absent	present	P(1 0 1)	$-(\tau_1+0+\tau_3)-(0+0+0)$
0 0 0	absent	absent	absent	P(0 0 0)	$-(0+0+0)-(0+0+0)$

the higher the probability of that configuration. The probability of configuration x is given by[5,6]:

$$\begin{aligned}\mathbb{P}_\Theta(x) &= \frac{1}{Z(\Theta)}\exp[-H(x)] \\ &= \frac{1}{Z(\Theta)}\exp\left[\sum_{j\in V}\tau_j x_j + \sum_{(j,k)\in E}\beta_{jk}x_j x_k\right],\end{aligned} \tag{2}$$

where $Z(\Theta)$ is the normalising constant (or partition function) that guarantees that the distribution sums to one:

$$Z(\Theta) := \sum_{x\in\{0,1\}^p}\exp\left[\sum_{j\in V}\tau_j x_j + \sum_{(j,k)\in E}\beta_{jk}x_j x_k\right]. \tag{3}$$

This distribution sums to one when $Z(\Theta)$ sums over all possible configurations. For a small number of variables, as in our example, computing the normalising constant is feasible. When the number of variables increases, however, the state space (set of possible configurations) increases exponentially, which makes the normalising constant intractable.

Thus, computing the full likelihood function for the model is computationally prohibitive. An alternative is to use the so-called pseudo-likelihood, which only uses the (conditional) probability of X_j given all other nodes $X_{\setminus j}$. For the expression of this conditional probability, the normalising constant reduces to the sum over all possible configurations of one single node (X_j), which is just $\{0,1\}$. In this case, the normalising constant becomes

$$Z(\Theta) := 1 + \exp\left[\tau_j + \sum_{k\in V_{\setminus j}}\beta_{jk}x_k\right], \tag{4}$$

where V is the set of nodes $\{1, 2, .., p\}$ and $V_{\setminus j}$ is the set of nodes, excluding node j. Therefore, the normalising constant for the conditional probability of X_j given all other nodes $X_{\setminus j}$ is in fact tractable, even if the normalising constant for the full model is not. From combining equation 2 and 4, it follows that the conditional probability of X_j given all other nodes $X_{\setminus j}$ is given by

$$\mathbb{P}_\Theta(x_j \mid x_{\setminus j}) = \frac{\exp\left[\tau_j x_j + x_j \sum_{k \in V_{\setminus j}} \beta_{jk} x_k\right]}{1 + \exp\left[\tau_j + \sum_{k \in V_{\setminus j}} \beta_{jk} x_k\right]}. \tag{5}$$

With our network of three variables and the parameters in Supplementary Figure S2, we can calculate the probabilities of each configuration. As an example, we will compare two probabilities: the probability of $x_2 = 1$ given that (1) $x_1 = 1$ and $x_3 = 0$ and that (2) $x_1 = 0$ and $x_3 = 1$. In the case of $\mathbb{P}_\Theta(x_2 = 1 \mid x_1 = 1, x_3 = 0)$, the Hamiltonian can be computed by filling in the parameters as given in Supplementary Figure S2:

$$H(x) = -(\tau_1 + \tau_2) - (\beta_{12}) = -(.3 + .7) - .54 = -1.54 \tag{6}$$

Now, the probability can be computed using Formula 5:

$$\mathbb{P}_\Theta(x_2 = 1 \mid x_1 = 1, x_3 = 0) = \frac{e^{-1.54}}{1 + e^{-1.54}} = .82 \tag{7}$$

Doing the same for the second case results in probability $\mathbb{P}_\Theta(x_2 = 1 \mid x_1 = 0, x_3 = 1) = .78$. Apparently, in this model, the probability that a person will be fatigued is higher for a person who has insomnia but no weight loss, than for someone who suffers from weight loss, but does not have insomnia.

Since the Ising model assumes that only pairwise interactions exist, the conditional probability of x_j given all other variables is therefore reduced to the conditional probability of x_j given the neighbours of x_j:

$$\mathbb{P}_\Theta(x_j \mid x_{\setminus j}) = \mathbb{P}_\Theta(x_j \mid x_{ne(j)}) \tag{8}$$

where ne(j) is the set of neighbours of node x_j. We can now relate the Ising model to undirected graphs, since any set of variables for which conditional probabilities can be written as in (8) satisfy

the Markov property[2]. A set of random variables with the Markov property can be described by an undirected graph. Such graphs are also known as Markov random fields, Markov networks, or undirected graphical models[7]. In daily practice, the graph structure of psychological constructs is unknown, as opposed to the spins in a ferromagnet that are arranged as in Supplementary Figure S1. Therefore, the estimation of the unknown graph structure is of central importance.

By viewing X_j as the response variable and all other variables $X_{\setminus j}$ as the predictors, we may fit a logistic regression function to investigate which nodes are part of the neighbourhood of the response variable. The intercept τ_j of the regression equation is the threshold of the variable, while the slope β_{jk} of the regression equation is the connection strength between the relevant nodes. In order to perform the logistic regression, we need multiple independent observations of the variables.

To establish which of the variables in the data are neighbours of a given variable, and which are not, we used ℓ_1-regularised logistic regression[6,8]. This technique is commonly called the *lasso* (least absolute shrinkage and selection operator) and optimises neighbourhood selection in a computationally efficient way, by optimising the convex function

$$\hat{\Theta}_j^{\rho} = \arg\min_{\Theta_j}\{-x_{ij}\cdot(\tau_j + \sum_{k\in V_{\setminus j}} \beta_{jk}x_{ik}) + \log(1+\exp\{\tau_j + \sum_{k\in V_{\setminus j}} x_{ik}\beta_{jk}\}) + \rho\sum_{k\in V_{\setminus j}}|\beta_{jk}|\}, \quad (9)$$

in which i represents the independent observations $\{1,2,..,n\}$, $\hat{\Theta}_j^{\rho}$ contains all β_{jk} and τ_j parameters, and ρ is the penalty parameter. The final term with ρ ensures shrinkage of the regression coefficients[6,9]. This optimisation procedure is applied to each variable in turn with all other variables as predictors. To this end, the R package `glmnet` can be used[10]. The `glmnet` package uses a range of maximal 100 penalty parameter values. The result is a list of 100 possible neighbourhood sets, some of which may be the same. To choose the best set of neighbours, we used the EBIC[11] (extended Bayesian Information Criterion). The EBIC has a good trade-off between positive selection rates (proportions of true selected edges) and false discovery rates (proportions of false positives among the selected edges) in selecting edges in the Ising model[12]. The EBIC is the ordinary BIC with an additional term that penalises more complexity (more connections) and more variables. The EBIC is preferable in this situation, because the ordinary BIC is too liberal for high

dimensional data[11]. The EBIC is represented as

$$\mathrm{BIC}_\gamma(j) = -2\ell(\hat{\Theta}_J) + |J| \cdot \log(n) + 2\gamma|J| \cdot \log(p-1), \tag{10}$$

in which $\ell(\hat{\Theta}_j)$ is the log likelihood (see below), $|J|$ is the number of neighbours selected by logistic regression at a certain penalty parameter ρ, n is the number of observations, $p-1$ is the number of covariates (predictors), and γ is a hyperparameter, determining the strength of prior information on the size of the model space[13]. From equation (5), it follows that the log likelihood of the conditional probability of X_j given its neighbours $X_{ne(j)}$ over all observations is

$$\ell(\hat{\Theta}_j) = \sum_{i=1}^{n} \left(\tau_j x_{ij} + \sum_{k \in V_{\setminus j}} \beta_{jk} x_{ij} x_{ik} - \log(1 + \exp\{\tau_j + \sum_{k \in V_{\setminus j}} x_{ik}\beta_{jk}\}) \right). \tag{11}$$

The EBIC has been shown to be consistent for model selection and to performs best with hyper-parameter $\gamma = 0.25$ for the Ising model[13]. The model with the set of neighbours J that has the lowest EBIC is selected.

At this stage, we have the regression coefficients of the best set of neighbours for every variable; i.e., we have both β_{jk} and β_{kj} and have to decide whether there is an edge between nodes j and k or not. Two rules can be applied to make the decision: the AND rule, where an edge is present if both estimates are nonzero, and the OR rule, where an edge is present if at least one of the estimates is nonzero[6,8].

Although we do have the final edge set by applying one of the rules, note that for any two variables j and k, we get two results: the result of the regression of j on k (β_{jk}), and the result of the regression of k on j (β_{kj}). To obtain an undirected graph, the weight of the edge between nodes j and k, ω_{jk}, is defined as the mean of both regression coefficients β_{jk} and β_{kj}. This methodology is incorporated in the `R` package `IsingFit` (`http://cran.r-project.org/web/packages/IsingFit/IsingFit.pdf`).

Supplementary Results

Another way to assess effectivity of the method is to inspect the F1 score, which takes both *precision* and *recall* into account [14]. Precision expresses the proportion of correctly estimated connections with respect to all estimated connections and is defined as PRE = TP/(TP + FP). Recall corresponds to the proportion of correctly estimated connections with respect to all connections that should have been estimated and is defined as REC = TP/(TP + FN), which is in fact the same as sensitivity. The F1 score is then defined as $F_1 = 2 \cdot \text{PRE} \cdot \text{REC}/(\text{PRE} + \text{REC})$.

F1 scores increase with more observations but to a different extent for different conditions (Supplementary Table S2). For almost all conditions with more than 100 observations, F1 scores are moderate to high ($M = .713$, $sd = .143$)). The only exception results when the largest random and scale-free networks (100 nodes) are coupled with the highest level of connectivity, as we also have seen in the results of sensitivity and specificity in the main text. In these cases, the F1 score is low (.271) and moderate (.516), respectively.

More detailed information about *eLasso*'s performance is given by the two components of the F1 score: precision and recall. Since recall is the same as sensitivity, we only discuss precision here. Overall, precision is high across all conditions ($M = .920$, $sd = .122$) with lower precision scores for the largest and most dense random networks (see Supplementary Table S3).

In some cases it might be desirable to have a higher recall at the expense of precision. In *eLasso*, recall can generally be increased in two ways. First, *eLasso* identifies the set of neighbours for each node by computing the EBIC. EBIC penalises solutions that involve more variables and more neighbours. This means that if the number of variables is high, EBIC tends to favour solutions that assign fewer neighbours to any given node. In this procedure, a hyperparameter called γ determines the strength of the extra penalty on the number of neighbours [13]. In our main simulation study, we used $\gamma = .25$. When $\gamma = 0$, no extra penalty is given for the number of neighbours, which results in a greater number of estimated connections. Second, we applied the so-called AND-rule to determine the final edge set. The AND-rule requires both regression coefficients β_{jk} and β_{kj} (from the ℓ_1-regularised logistic regression of X_j on X_k and of X_k on X_j) to be nonzero. Alternatively, the OR-rule can be applied. The OR-rule requires only one of β_{jk} and β_{kj} to be nonzero, which also results in more estimated connections.

Applying the OR-rule and $\gamma = 0$, indeed results in a loss of precision, but is still reasonable across all conditions ($M = .735$, $sd = .131$; Supplementary Table S3). This indicates that, in a liberal setting, the estimated network contains more connections that are not present in the true network than in the more conservative setting.

To conclude, inspecting the F1 scores (and its component precision), confirm the results for specificity. *eLasso* adequately recovers the true network structure in almost all simulated conditions. Exceptions to this rule are larger and/or more dense networks.

1. Ising, E. Beitrag zur theorie des ferromagnetismus. *Z. Phys. A-Hadrons. Nucl.* **31**, 253–258 (1925).

2. Kindermann, R. & Snell, J. L. *Markov random fields and their applications*, vol. 1 (American Mathematical Society Providence, RI, 1980).

3. Association, A. P. *Diagnostic and statistical manual of mental disorders (4th ed., text rev.).* (American Psychiatric Publishing, Inc., 2000).

4. Cipra, B. A. An introduction to the ising model. *Am. Math. Mon.* **94**, 937–959 (1987).

5. Loh, P. L. & Wainwright, M. J. Structure estimation for discrete graphical models: Generalized covariance matrices and their inverses. *Ann. Stat.* **41**, 3022–3049 (2013).

6. Ravikumar, P., Wainwright, M. J. & Lafferty, J. D. High-dimensional ising model selection using l1-regularized logistic regression. *Ann. Stat.* **38**, 1287–1319 (2010).

7. Wainwright, M. J. & Jordan, M. I. Graphical models, exponential families, and variational inference. *Found. Trends Mach. Learn.* **1**, 1–305 (2008).

8. Meinshausen, N. & Bühlmann, P. High-dimensional graphs and variable selection with the lasso. *Ann. Stat.* **34**, 1436–1462 (2006).

9. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B* **58**, 267–288 (1996).

10. Friedman, J., Hastie, T. & Tibshirani, R. Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.* **33**, 1 (2010).

11. Chen, J. & Chen, Z. Extended bayesian information criteria for model selection with large model spaces. *Biometrika* **95**, 759–771 (2008).

12. Foygel, R. & Drton, M. High-dimensional ising model selection with bayesian information criteria. *arXiv preprint arXiv:1403.3374* (2014).

13. Foygel, R. & Drton, M. Bayesian model choice and information criteria in sparse generalized linear models. *arXiv preprint arXiv:1112.5635* (2011).

14. Jardine, N. & van Rijsbergen, C. J. The use of hierarchic clustering in information retrieval. *Inform. Storage Ret.* **7**, 217–240 (1971).

Supplementary Table S2: F1-scores (based on precision and recall) as a measure of performance of eLasso. Data is simulated under various conditions (s_{size}, n_{nodes}, connectedness (p (probability of a connection), pa (preferential attachment), pr (probability of rewiring)) when the AND-rule and $\gamma = .25$ is applied. For networks with 100 nodes, deviating levels of connectedness are displayed between brackets. Results of applying eLasso with the OR-rule and $\gamma = 0$ are displayed between brackets.

		Random			Scale-free			Small world		
s_{size}	n_{nodes}	$p = .1(.05)$	$p = .2(.10)$	$p = .3(.15)$	$pa = 1$	$pa = 2(1.25)$	$pa = 3(1.5)$	$pr = .1$	$pr = .5$	$pr = 1$
100	10	0.399 (0.439)	0.384 (0.512)	0.369 (0.529)	0.360 (0.481)	0.306 (0.504)	0.291 (0.529)	0.401 (0.581)	0.405 (0.546)	0.409 (0.566)
	20	0.306 (0.432)	0.282 (0.461)	0.291 (0.460)	0.284 (0.423)	0.186 (0.402)	0.135 (0.406)	0.314 (0.478)	0.314 (0.471)	0.286 (0.460)
	30	0.254 (0.406)	0.225 (0.420)	0.207 (0.347)	0.252 (0.370)	0.118 (0.275)	0.083 (0.187)	0.275 (0.446)	0.255 (0.406)	0.251 (0.422)
	100	0.148 (0.290)	0.100 (0.187)	0.054 (0.160)	0.149 (0.275)	0.125 (0.216)	0.076 (0.141)	0.214 (0.351)	0.160 (0.310)	0.168 (0.299)
500	10	0.702 (0.711)	0.701 (0.728)	0.746 (0.757)	0.713 (0.747)	0.663 (0.769)	0.659 (0.789)	0.763 (0.779)	0.754 (0.798)	0.750 (0.788)
	20	0.693 (0.690)	0.682 (0.718)	0.665 (0.694)	0.653 (0.692)	0.529 (0.709)	0.462 (0.675)	0.713 (0.732)	0.683 (0.714)	0.683 (0.723)
	30	0.663 (0.693)	0.642 (0.670)	0.442 (0.554)	0.626 (0.667)	0.409 (0.584)	0.390 (0.508)	0.687 (0.714)	0.660 (0.691)	0.658 (0.682)
	100	0.580 (0.567)	0.292 (0.281)	0.140 (0.147)	0.539 (0.597)	0.472 (0.533)	0.296 (0.413)	0.644 (0.673)	0.600 (0.640)	0.594 (0.639)
1000	10	0.838 (0.796)	0.792 (0.781)	0.811 (0.804)	0.786 (0.787)	0.754 (0.823)	0.760 (0.858)	0.824 (0.824)	0.835 (0.826)	0.838 (0.830)
	20	0.791 (0.764)	0.776 (0.776)	0.745 (0.760)	0.741 (0.764)	0.691 (0.764)	0.598 (0.784)	0.794 (0.786)	0.781 (0.770)	0.790 (0.774)
	30	0.779 (0.748)	0.725 (0.720)	0.563 (0.634)	0.726 (0.734)	0.556 (0.674)	0.505 (0.657)	0.786 (0.767)	0.779 (0.751)	0.778 (0.749)
	100	0.716 (0.705)	0.400 (0.430)	0.183 (0.231)	0.678 (0.686)	0.578 (0.638)	0.441 (0.483)	0.769 (0.744)	0.725 (0.717)	0.734 (0.716)
2000	10	0.821 (0.842)	0.864 (0.830)	0.879 (0.835)	0.847 (0.845)	0.834 (0.877)	0.822 (0.887)	0.877 (0.865)	0.884 (0.857)	0.873 (0.845)
	20	0.838 (0.804)	0.847 (0.811)	0.821 (0.794)	0.807 (0.804)	0.762 (0.814)	0.719 (0.813)	0.863 (0.817)	0.847 (0.809)	0.840 (0.807)
	30	0.846 (0.798)	0.807 (0.762)	0.650 (0.678)	0.812 (0.800)	0.646 (0.777)	0.598 (0.718)	0.854 (0.810)	0.844 (0.794)	0.838 (0.794)
	100	0.800 (0.754)	0.496 (0.494)	0.226 (0.271)	0.780 (0.756)	0.704 (0.689)	0.516 (0.531)	0.838 (0.788)	0.817 (0.769)	0.814 (0.770)

Supplementary Table S3: Precision and recall on which the F1-score is based. Data is simulated under various conditions (s_{size}, n_{nodes}, connectedness (p (probability of a connection), pa (preferential attachment), pr (probability of rewiring)) when the AND-rule and $\gamma = .25$ is applied. For networks with 100 nodes, deviating levels of connectedness are displayed between brackets. Results of applying eLasso with the OR-rule and $\gamma = 0$ are displayed between brackets.

			Random			Scale-free			Small world		
s_{size}	n_{nodes}		$p = .1(.05)$	$p = .2(.10)$	$p = .3(.15)$	$pa = 1$	$pa = 2(1.25)$	$pa = 3(1.5)$	$pr = .1$	$pr = .5$	$pr = 1$
100	10	REC	0.256 (0.348)	0.241 (0.395)	0.229 (0.409)	0.221 (0.363)	0.184 (0.380)	0.172 (0.397)	0.253 (0.458)	0.257 (0.412)	0.260 (0.434)
		PRE	0.896 (0.595)	0.947 (0.729)	0.949 (0.750)	0.962 (0.712)	0.903 (0.747)	0.941 (0.793)	0.962 (0.794)	0.958 (0.810)	0.946 (0.814)
	20	REC	0.183 (0.324)	0.166 (0.339)	0.173 (0.339)	0.168 (0.315)	0.104 (0.288)	0.074 (0.305)	0.188 (0.359)	0.189 (0.349)	0.168 (0.342)
		PRE	0.936 (0.645)	0.946 (0.722)	0.924 (0.718)	0.906 (0.645)	0.886 (0.663)	0.822 (0.609)	0.952 (0.718)	0.931 (0.727)	0.952 (0.706)
	30	REC	0.146 (0.295)	0.128 (0.307)	0.118 (0.242)	0.146 (0.269)	0.064 (0.186)	0.044 (0.126)	0.160 (0.328)	0.147 (0.287)	0.144 (0.305)
		PRE	0.953 (0.653)	0.910 (0.660)	0.818 (0.610)	0.932 (0.593)	0.801 (0.524)	0.765 (0.37)	0.960(0.695)	0.967 (0.689)	0.951 (0.686)
	100	REC	0.080 (0.186)	0.056 (0.132)	0.031 (0.134)	0.081 (0.185)	0.067 (0.139)	0.040 (0.085)	0.121 (0.238)	0.087 (0.205)	0.092 (0.195)
		PRE	0.957 (0.657)	0.485 (0.325)	0.249 (0.199)	0.936 (0.537)	0.930 (0.484)	0.818 (0.423)	0.961 (0.670)	0.963 (0.632)	0.970 (0.644)
500	10	REC	0.550 (0.649)	0.551 (0.672)	0.617 (0.704)	0.561 (0.687)	0.501 (0.713)	0.499 (0.734)	0.650 (0.726)	0.628 (0.765)	0.623 (0.757)
		PRE	0.972 (0.787)	0.963 (0.795)	0.945 (0.819)	0.977 (0.820)	0.982 (0.834)	0.971 (0.852)	0.924 (0.840)	0.943 (0.833)	0.940 (0.822)
	20	REC	0.539 (0.633)	0.537 (0.678)	0.527 (0.643)	0.492 (0.613)	0.364 (0.619)	0.302 (0.557)	0.569 (0.691)	0.538 (0.665)	0.538 (0.676)
		PRE	0.970 (0.758)	0.935 (0.763)	0.903 (0.754)	0.970 (0.795)	0.970 (0.830)	0.978 (0.855)	0.954 (0.778)	0.935 (0.772)	0.936 (0.777)
	30	REC	0.508 (0.637)	0.498 (0.620)	0.298 (0.470)	0.999 (0.985)	0.999 (0.991)	0.999 (0.994)	0.996 (0.970)	0.995 (0.965)	0.995 (0.965)
		PRE	0.955 (0.759)	0.901 (0.728)	0.851 (0.674)	0.971 (0.755)	0.951 (0.786)	0.928 (0.722)	0.960 (0.777)	0.953 (0.753)	0.946 (0.746)
	100	REC	0.416 (0.537)	0.189 (0.336)	0.091 (0.164)	0.372 (0.498)	0.311 (0.420)	0.174 (0.289)	0.481 (0.600)	0.433 (0.554)	0.428 (0.558)
		PRE	0.961 (0.733)	0.642 (0.399)	0.311 (0.253)	0.979 (0.746)	0.979 (0.731)	0.973 (0.720)	0.974 (0.765)	0.973 (0.758)	0.971 (0.748)
1000	10	REC	0.726 (0.794)	0.671 (0.752)	0.710 (0.783)	0.662 (0.756)	0.620 (0.781)	0.622 (0.818)	0.738 (0.814)	0.751 (0.814)	0.758 (0.820)
		PRE	0.991 (0.799)	0.966 (0.812)	0.945 (0.826)	0.967 (0.821)	0.963 (0.870)	0.975 (0.902)	0.934 (0.835)	0.941 (0.840)	0.938 (0.840)
	20	REC	0.666 (0.752)	0.665 (0.784)	0.630 (0.770)	0.599 (0.736)	0.533 (0.699)	0.431 (0.709)	0.680 (0.776)	0.664 (0.764)	0.681 (0.772)
		PRE	0.974 (0.776)	0.933 (0.769)	0.910 (0.751)	0.970 (0.793)	0.983 (0.842)	0.980 (0.875)	0.953 (0.797)	0.947 (0.776)	0.941 (0.776)
	30	REC	0.658 (0.736)	0.603 (0.710)	0.420 (0.583)	0.578 (0.699)	0.389 (0.566)	0.340 (0.545)	0.669 (0.752)	0.663 (0.738)	0.661 (0.740)
		PRE	0.955 (0.706)	0.907 (0.730)	0.855 (0.694)	0.974 (0.773)	0.972 (0.833)	0.979 (0.828)	0.955 (0.782)	0.944 (0.765)	0.945 (0.759)
	100	REC	0.572 (0.671)	0.286 (0.427)	0.125 (0.199)	0.519 (0.624)	0.409 (0.544)	0.284 (0.351)	0.636 (0.713)	0.579 (0.679)	0.593 (0.680)
		PRE	0.956 (0.743)	0.666 (0.433)	0.339 (0.275)	0.980 (0.760)	0.983 (0.772)	0.981 (0.772)	0.973 (0.779)	0.969 (0.758)	0.965 (0.756)
2000	10	REC	0.711 (0.808)	0.775 (0.830)	0.810 (0.842)	0.746 (0.842)	0.728 (0.870)	0.712 (0.891)	0.821 (0.880)	0.829 (0.864)	0.822 (0.866)
		PRE	0.972 (0.879)	0.976 (0.829)	0.962 (0.828)	0.981 (0.848)	0.977 (0.883)	0.971 (0.882)	0.941 (0.852)	0.947 (0.850)	0.930 (0.824)
	20	REC	0.741 (0.804)	0.770 (0.838)	0.754 (0.840)	0.691 (0.805)	0.624 (0.769)	0.566 (0.754)	0.793 (0.837)	0.769 (0.836)	0.762 (0.844)
		PRE	0.964 (0.805)	0.942 (0.785)	0.901 (0.753)	0.971 (0.803)	0.978 (0.865)	0.985 (0.883)	0.947 (0.799)	0.942 (0.783)	0.936 (0.773)
	30	REC	0.756 (0.808)	0.740 (0.808)	0.529 (0.656)	0.698 (0.807)	0.483 (0.712)	0.430 (0.612)	0.772 (0.837)	0.762 (0.818)	0.754 (0.825)
		PRE	0.960 (0.789)	0.887 (0.721)	0.842 (0.702)	0.973 (0.792)	0.975 (0.854)	0.979 (0.869)	0.956 (0.786)	0.944 (0.772)	0.944 (0.766)
	100	REC	0.688 (0.767)	0.385 (0.539)	0.160 (0.250)	0.648 (0.736)	0.548 (0.607)	0.349 (0.398)	0.738 (0.793)	0.708 (0.776)	0.703 (0.777)
		PRE	0.955 (0.742)	0.698 (0.456)	0.385 (0.295)	0.980 (0.777)	0.984 (0.799)	0.988 (0.794)	0.969 (0.783)	0.965 (0.761)	0.965 (0.763)