Recovering networks from distance data

Prabhakaran, S ; Boehm, A ; Metzner, K J ; Roth, V

JMLR: Workshop and Conference Proceedings 25:349–364, 2012 Asian Conference on Machine Learning

Recovering Networks from Distance Data

Sandhya Prabhakaran SANDHYA.PRABHAKARAN@UNIBAS.CH
Department of Mathematics and Computer Science, University of Basel, Bernoullistrasse 16, CH-4056 Basel, Switzerland

Karin J Metzner KARIN.METZNER@USZ.CH
University Hospital Zurich, Department of Medicine, Division of Infectious Diseases and Hospital Epidemiology, Rämistrasse 100, CH-8091 Zurich, Switzerland

Alexander Böhm ALEXANDER.BOEHM@SYNMIKRO.UNI-MARBURG.DE
LOEWE-Zentrum für Synthetische Mikrobiologie, Microbial Signaling, Hans-Meerwein-Strasse, 35043 Marburg, Germany

Volker Roth VOLKER.ROTH@UNIBAS.CH
Department of Mathematics and Computer Science, University of Basel, Bernoullistrasse 16, CH-4056 Basel, Switzerland

Editor: Steven C.H. Hoi and Wray Buntine

Abstract

A fully probabilistic approach to reconstructing Gaussian graphical models from distance data is presented. The main idea is to extend the usual central Wishart model in traditional methods to using a likelihood depending only on pairwise distances, thus being independent of geometric assumptions about the underlying Euclidean space. This extension has two advantages: the model becomes invariant against potential bias terms in the measurements, and can be used in situations which on input use a kernel- or distance matrix, without requiring direct access to the underlying vectors. The latter aspect opens up a huge new application field for Gaussian graphical models, as network reconstruction is now possible from any Mercer kernel, be it on graphs, strings, probabilities or more complex objects. We combine this likelihood with a suitable prior to enable Bayesian network inference. We present an efficient MCMC sampler for this model and discuss the estimation of module networks. Experiments depict the high quality and usefulness of the inferred networks.

Keywords: Network inference, Gaussian graphical models, pairwise Euclidean distances, MCMC

1. Introduction

Gaussian graphical models (GGMs) provide a rigid formalism to represent high-dimensional distributions of random variables (objects). Using GGMs one infers the network of dependencies amongst these objects through their pairwise partial correlations. The partial correlations are seen as a measure of conditional dependence between objects and are obtained from the inverse of the covariance matrix. Conditional independence is asserted between any two objects if the pairwise partial correlation is zero and this indicates the absence of an edge between these objects in the network. Identifying networks is a challenging problem when the unknown network structure has to be learnt from observed measurements that tend to be noisy and also when the number of objects are far more larger than the

measurements themselves. Further, traditional network inference models depend on geometric translations of the data which can pose problems as explained in Section 2. In many real-world scenarios one rarely has access to the objects' underlying vectorial representations but only to their pairwise distances implying that the geometric translations are entirely lost. In the current paper, we introduce a novel sparse network inference mechanism called the *Translation-invariant Wishart Network* (TiWnet) model that is designed solely to work on pairwise distances. To the best of our knowledge this is the first paper that deals with network structure discovery using pairwise distances. Additionally, to deal with noisy measurements and situations where the number of measurements are much smaller than the objects, we present the construction of *module networks* where networks are learnt on groups of variables called *modules*. To set the stage, we begin with a description of the classical framework for estimating sparse GGMs. One usually starts with a $n \times d$ observed data matrix X^o (the superscript ^{o} means "original" and is used here only for notational consistency), its d columns interpreted as the outcome of a measuring procedure in which some property of the n objects of interest is measured. In a biological setting, for instance, the objects could be n genes and one set of measurements (one column) could be gene expression values from one microarray. All d columns in X^o are assumed to be i.i.d. according to $\mathcal{N}(\mathbf{0}, \Sigma)$. Then, the inner product matrix $S^o = X^o(X^o)^t$ follows a central Wishart distribution $\mathcal{W}_d(\Sigma)$ in d degrees of freedom (if $d \geq n$, otherwise S^o is pseudo-Wishart [1]), and its likelihood as a function of the inverse covariance $\Psi := \Sigma^{-1}$ is

$$\mathcal{L}(\Psi) \propto |\Psi|^{\frac{d}{2}} \cdot \exp\left[-\tfrac{1}{2}\operatorname{tr}(\Psi S^o)\right]. \tag{1}$$

Every algorithm for network reconstruction relies on some potentially interesting sparsity structure garnered within the inverse covariance matrix $\Psi := \Sigma^{-1}$. Ψ contains the (scaled) partial correlations between the n random variables forming the nodes in the network: a zero entry in Ψ_{ij} concurs to no edge prevailing between the pair of random variables (i, j) in the network.

Related work. To infer the underlying network, it is straightforward (at least from a methodological viewpoint) to maximize the Wishart likelihood while ensuring that Ψ is sparse. This is exactly the approach followed in "graph lasso" (Friedman et al., 2007), where a ℓ_1 sparsity constraint on Ψ is used. A methodologically similar, but simplified approach that decouples this joint estimation problem into n independent neighborhood-selection problems is dealt in Meinhausen and Bühlmann (2006). The model presented in Kolar et al. (2010a) employs a logistic regression model with a ℓ_1/ℓ_2 penalty for the neighborhood-selection problem while additionally assuming a conditioning variable that holds information about the associations between nodes to steer sparsity. Another method to extract networks called "walk-summable graphs" is introduced in Johnson et al. (2006) where a neighborhood is constructed based on *walks* accumulated by every node in the graph and weighted as a function of the edgewise partial correlations present in Ψ.

1. The names of the Wishart distribution are inconsistent in the literature. We use the notation in Díaz-García et al. (1997).

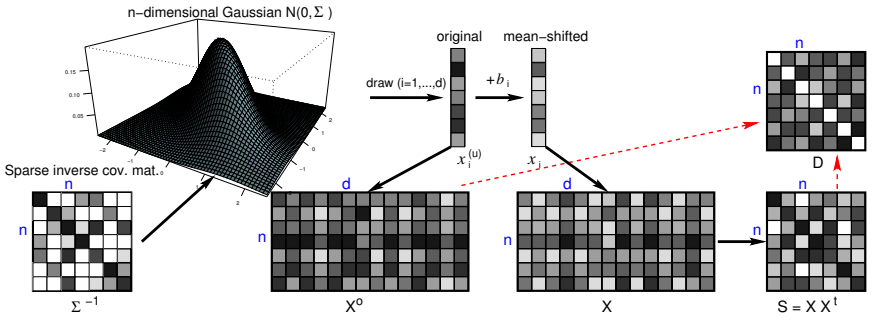

Figure 1: Assumed Underlying Generative Process. Black arrows indicate the workflow when drawing samples from this model; n, d: matrix dimensions. The dashed red arrows highlight the same distance matrix D produced from either the "original" data X^o or the "mean-shifted" data X.

2. Underlying Problems with Existing Methods

The above papers and related approaches, however, have been built on an assumption that the d columns in X^o are i.i.d. This particular assumption of considering columns to be *identically* distributed might be too restrictive: even if the underlying Gaussian generative process is a valid model, different column-wise bias terms are common in practice. In the above biological example, there might be global expression differences between the d microarrays. For valid network inference, it is therefore essential to model these unknown shifts and by doing so, one relaxes the column i.i.d. assumption [2]. To model this, such column-wise biases are included in the generative model by introducing a shifting operation in which scalar bias terms $b_{(i=1,\dots,d)}$ are added to the "original" column vectors $\boldsymbol{x}_i^o$, which results in a mean-shifted vector $\boldsymbol{x}_i$, forming the i-th column in X, cf. Figure 1. Hence the columns come from *different* distributions i.e. they cease to be *identically distributed*. In the classical case of not considering column biases, X^o is distributed as $\mathcal{N}(\mathbf{0}, \Sigma)$, but in TiWnet which now accommodates these column biases, the joint distribution of all matrix elements is expressed, that here is matrix normal $X \sim \mathcal{N}(M, \Omega)$ with mean matrix $M := \mathbf{1}_n \boldsymbol{b}_d^t$ and covariance tensor $\Omega := \Sigma_{n\times n} \otimes I_d$. This model implies that $S = XX^t$ follows a *non-central* Wishart distribution $S \sim \mathcal{W}(\Sigma, \Theta)$ with non-centrality matrix $\Theta := \Sigma^{-1} M M^t$. Practical use of the non-central Wishart for network inference, however, is severely hampered by its complicated form and more so, the problem of estimating the unknown non-centrality matrix Θ based on only one observation of S which is problematically analogous to identifying the mean of any distribution given only a single data point.

It is, thus, desirable to use a simpler distribution. One possible way of handling such column biases is to "center" the columns by subtracting the empirical column means $\hat{b}_i$, and using the matrix $S_C = (X - \mathbf{1}\hat{\boldsymbol{b}}^t)(X - \mathbf{1}\hat{\boldsymbol{b}}^t)^t$ in the standard central Wishart model. Since the entries in the i-th column, $\{x_{1i}, \cdots, x_{\mathrm{n}i}\}$, are not independent but coupled via the Σ-part in Ω, this centering, however, brings about undesired side effects; apart from removing

2. Network inference based on non-i.i.d. data has been studied previously, for example in Kolar et al. (2010b), Zhou et al. (2010) but these do not deal with data having different column-wise biases.

the additive shift, the original columns are modified with the resulting column-centered matrix S_C being rank deficient. As a consequence, $S_C \not\sim \mathcal{W}(\Sigma)$. Instead, S_C follows the more complicated *translation invariant* Wishart distribution, see Equation (2) below. Our experiments show that the presence of column-wise shifts is not only a theoretical problem of model mismatch but also a severe practical problem for inferring the underlying network.

Another problem-arising situation is where even observing $X_{n\times d}$ is not valid, instead one assumes access to a measuring procedure which directly returns pairwise relationships between n objects. Two variants are considered: either a positive definite similarity matrix identified with the matrix S is measured, or pairwise squared distances arranged in a matrix D is measured, defined component-wise as $D_{ij} = S_{ii} + S_{jj} - 2S_{ij}$. In the first case with S or in the second case with D, column-centering is still possible by the usual "centering" operation in kernel PCA: $S_C = QSQ^t = -(1/2)QDQ^t$, with $Q_{ij} = \delta_{ij} - \frac{1}{n}$. However, using this matrix in the standard Wishart model induces obviously the same problems related to model mismatch as in the vectorial case above.

3. Novel Solution to Network Inference

The proposed solution to overcome the above intertwined problems of having to work with column-wise shifts and the complicated non-central Wishart is to use a likelihood model in TiWnet that depends only on squared Euclidean distances D where these distances are not affected by any column-wise shifts, cf. the red dashed arrows in Figure 1. The likelihood model invariant to shifts has been studied before in the *Translational-invariant Wishart Dirichlet* (TiWD) cluster process (Vogt et al., 2010). This is a fully-probabilistic model for clustering and is specifically devised to work with pairwise Euclidean distances by suitably encoding the translational and rotational invariances. Although the TiWD clustering model and TiWnet use identical likelihoods, the priors in both models are different. We develop a new prior construction that enables network inference. This prior is similar to the spike and slab model introduced in Mitchell and Beauchamp (1988).

The TiWD clustering model uses a Dirichlet-Multinomial type prior over clusters with the priors being restricted to block-diagonal form. This kind of prior construction is incompetent for network inference since if such a prior is used, all networks would always decompose into separated clusters which are fully connected within themselves. Therefore, to enable network recovery an enhanced prior construction is necessary and to this end, TiWnet uses a prior that relaxes the block-diagonal form. The TiWnet prior is designed to ensure sparsity of network edges and the resulting Ψ is constructed to be a sparse diagonally-dominant matrix. The prior construction is further elaborated in Section 4.2.

We illustrate the difference between the TiWnet and TiWD prior constructions in Figure 2. The top panel of Figure 2 depicts the original network generated using Ψ (no longer block-diagonal) meant for network inference and the inferred network using TiWnet. The black/green edges depict the positive/negative partial correlations between the nodes. The bottom panel of Figure 2 shows different views of the network obtained using the TiWD clustering method: the left plot shows that the network is densely connected bearing no resemblance to the original network and the right plot highlights that the network gets decomposed into separate fully-connected clusters. Moreover, the network fails to capture the positive/negative partial correlations between the nodes since the inferred Σ in the case

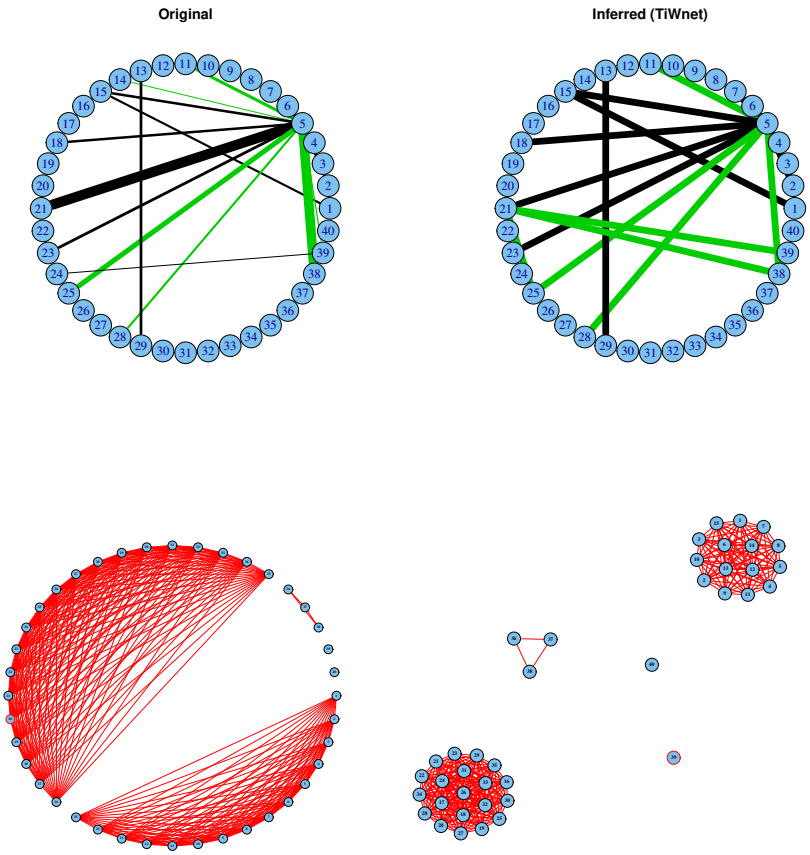

Figure 2: Illustration of the difference between TiWnet and TiWD clustering (Vogt et al., 2010) using data generated from Ψ (no longer block-diagonal) designed for network inference. **Top**: Left: Original network. Right: Inferred sparse network using TiWnet. The black/green edges denote positive/negative partial correlations between nodes. **Bottom**: Left: Densely-connected network obtained using the block-diagonal Σ inferred from TiWD clustering. The edges do not differentiate between positive/negative partial correlations. Right: The same network as on the left now showing that the network decomposes into separate fully-connected clusters. Here the network decomposes into 5 clusters viz. 3 fully-connected and 2 singletons.

of TiWD clustering only contains information regarding the cluster structure but without signs. From the above discussion, it is obvious that clustering is a specialized case of network inference and that general networks cannot be recovered using the TiWD clustering model of Vogt et al. (2010).

Combining this enhanced prior suitable for network reconstruction with the likelihood, we are able to perform Bayesian network inference in TiWnet. For inference we devise a Metropolis-within-Gibbs sampler where the Metropolis-Hastings step proposes an appropriate Ψ matrix and the Gibbs iteration involves repeating the Metropolis-Hastings step

for every node. We refer the reader to Section 4.3 for complete details of our inference mechanism.

Below we present the major contributions of TiWnet as against other related network inference approaches. "Graph lasso" was devised for estimating a truly sparse network from the data. Since TiWnet is fully probabilistic, on output we not only obtain a single network but a distribution of networks explaining the data. For many cases in reality, this is more meaningful since one has access to possible structural variations of the extracted networks. Further, if required, our method has the flexibility to yield a single MAP-estimate network by simulated annealing. Such a sparse annealed network has desirable properties which seem to be difficult to achieve by "graph lasso". It is necessary to point out that the central Wishart model is only justified for *zero* column-shifts. All of the earlier methods, to our knowledge, have solely relied on this and not catered to the inherent column-shifts, and might generate biased networks. Instead, TiWnet based on D is shift-invariant and we show that in practical applications this shift invariance is essential for recovering correct networks. Due to this, network reconstruction is possible using any D induced by a Mercer kernel [3].

4. The TiWnet Model

4.1. Likelihood model

One starts with a given matrix D containing pairwise squared distances between row vectors of an unobserved matrix $X \sim \mathcal{N}(M, \Omega)$. This means that in addition to the classical framework for GGMs, arbitrary column biases $b_{(i=1,\ldots,d)}$ are now allowed which "shift" the columns in X but leave the pairwise distances unaffected.

Since by assumption D contains squared Euclidean distances, there is a set of inner product matrices S that fulfill $D_{ij} = S_{ii} + S_{jj} - 2S_{ij}$ (McCullagh, 2009). If S_* is one (any) such matrix, the equivalence class of these matrices mapping to a single D is formally described as set $\mathbb{S}(D) = \{S | S = S_* + \mathbf{1}\boldsymbol{v}^t + \boldsymbol{v}\mathbf{1}^t, S \succeq 0, \boldsymbol{v} \in \mathbb{R}^n\}$. This $\mathbb{S}$ is exactly the set of inner product matrices that can be constructed by arbitrarily biasing the column vectors in X. Shifting the viewpoint from column to row vectors, this invariance means that the density does not depend on the origin of the coordinate system in which the n objects are represented as vectors containing d different measurements. Column-wise biases referred to before reduce in this view to simple shifts of the origin of an underlying coordinate system. And the idea of column centering reduces to selecting one specific representative S_C from the set of all possible $S \in \mathbb{S}(D)$, namely the one whose origin is at the sample mean. Since such column centering, however, destroys the central Wishart property of S (assuming it was a Wishart matrix before), the strategy is therefore to avoid the selection of a representative $S \in \mathbb{S}$ altogether. Instead, a probabilistic model for D is used. It turns out that the distribution of an arbitrary $S \in \mathbb{S}$ can be derived analytically as a singular Wishart distribution with a rank-deficient covariance matrix (McCullagh, 2009; Vogt et al.,

3. This statement does not necessarily imply that it is *meaningful* to use any Mercer kernel for reconstructing a Gaussian graphical model. The main focus here is not on kernels as a means for mapping input vectors to high-dimensional feature spaces in order to exploit nonlinearity in the input space. Rather, kernels are viewed as similarity measures between structured objects having no direct vectorial representation.

2010), and its likelihood in the rank-deficient inverse covariance $\widetilde{\Psi}$ is

$$\begin{aligned}\mathcal{L}(\widetilde{\Psi}) &\propto \det(\widetilde{\Psi})^{\frac{d}{2}} \exp\big(-\tfrac{1}{2}\mathrm{tr}(\widetilde{\Psi}S)\big)\\ &= \det(\widetilde{\Psi})^{\frac{d}{2}} \exp\big(\tfrac{1}{4}\mathrm{tr}(\widetilde{\Psi}D)\big),\end{aligned} \tag{2}$$

with $\widetilde{\Psi} = \widetilde{\Psi}(\Psi) = \Psi - (\mathbf{1}^t\Psi\mathbf{1})^{-1}\Psi\mathbf{1}\mathbf{1}^t\Psi$. Although the matrix S appears in the first term in Equation (2), the likelihood is constant on all $S \in \mathbb{S}(D)$, hence it depends only on D. Further $\widetilde{\Psi}$ has rank $r = n-1$, and $\mathbb{S}$ also contains rank-deficient matrices (like the column-centered S_C of rank $q \leq n-1$). In fact, Equation (2) represents the marginal likelihood based on the statistic $(X - \mathbf{1}\hat{\boldsymbol{b}}^t)$. On further exclusion of scalar multiples by basing the marginal likelihood on the standardized statistics $(X - \mathbf{1}\hat{\boldsymbol{b}}^t)/\|(X - \mathbf{1}\hat{\boldsymbol{b}}^t)\|$, one arrives at the shift- and scale-invariant likelihood, cf. Tunnicliffe-Wilson (1989); McCullagh (2009):

$$\begin{aligned}\mathcal{L}(\widetilde{\Psi}) &\propto \det(\widetilde{\Psi})^{\frac{d}{2}}\mathrm{tr}(\widetilde{\Psi}S)^{-(n-1)d/2}\\ &= \det(\widetilde{\Psi})^{\frac{d}{2}}\mathrm{tr}(-(1/2)\widetilde{\Psi}D)^{-(n-1)d/2}.\end{aligned} \tag{3}$$

Thus, there is a valid probabilistic model underlying Equation (3), and with a suitable prior Bayesian inference for Ψ is well-defined.

The reader should notice that Equation (3) can be computed either from the distances D, or from *any* inner product matrix $S \in \mathbb{S}(D)$. The practical advantage of this property is as follows: in the literature one finds a large "zoo" of Mercer kernels for many objects ranging from graphs to probability distributions to strings etc. These kernels represent elements in $\mathbb{S}(D)$ where D is induced by $D_{ij} = S_{ii} + S_{jj} - 2S_{ij}$. Most of the methods used for constructing kernels have no information about the origin of the kernel's underlying space, meaning that the exact form of the kernel matrix is irrelevant as long as it belongs to set $\mathbb{S}(D)$. Most supervised kernel methods like SVMs are invariant against choosing different representatives in $\mathbb{S}$, and in common unsupervised kernel methods like kernel PCA the *rows* of X are considered i.i.d. implying that subtracting the empirical column means (leading to S_C) is the desired centering procedure for selecting a candidate in $\mathbb{S}(D)$. However, the sampling model for GGMs is not invariant against choosing $S \in \mathbb{S}$, and even the centered S_C does not work properly in a central Wishart model. With TiWnet based on D, we can now use these Mercer kernels for reconstructing GGMs without being dependent on the choice of $S \in \mathbb{S}$.

4.2. Prior construction

For network inference in a Bayesian framework, we complement the likelihood (3) with a prior over Ψ. In principle, any distribution over symmetric positive definite matrices can be used. The likelihood has a simple functional form in $\widetilde{\Psi}$, but our main interest is in Ψ, since zeros in Ψ determine the topology. Unfortunately, the likelihood in Ψ is not in standard form making it plausible to use a MCMC sampler. For any two Σ matrices, Σ_1 and Σ_2 that are related by $\Sigma_2 = \Sigma_1 + \mathbf{1}\boldsymbol{v}^t + \boldsymbol{v}\mathbf{1}^t$, the likelihood is the same for Σ_1 and Σ_2 (McCullagh, 2009). This means that Ψ is non-identifiable and a sampler will have problems with such constant likelihood regions by continuing to visit them unless a prior is used that breaks this symmetry.

To deal with this problem, we quantize the space of possible Ψ-matrices such that any two candidates have different likelihood. This is achieved with a two-component prior: $P_1(\Psi)$ is uniform over the discrete set $\mathcal{A}$ of symmetric diagonally-dominant matrices with off-diagonal entries in $\{-1, +1, 0\}$, and diagonal entries are deterministic, conditioned on the off-diagonal elements i.e. $\Psi_{ii} = \sum_{j \neq i} |\Psi_{ij}| + \epsilon$ where ϵ is a positive constant added to ensure full rank of Ψ. Thus $\mathcal{A} = \{\Psi | \Psi_{ij} \in \{-1, +1, 0\}, \Psi_{ji} = \Psi_{ij}, \Psi_{ii} = \sum_{j \neq i} |\Psi_{ij}| + \epsilon\}$. Note that we treat only the off-diagonal entries as random variables. Note also that in this simple model we differentiate only between positive, negative and zero partial correlations, but it is straightforward to use more levels. Enforcing such a diagonally-dominant matrix construction ensures that the matrix will be positive definite. The usage of diagonally-dominant matrices for network reconstruction is further justified since these matrices form a strict subclass of GGMs that are walk summable (Johnson et al., 2005) and in Anandkumar et al. (2011) theoretical guarantees are provided establishing that walk-summable graphs make consistent sparse structure estimation possible. The second component of the prior is a sparsity-inducing prior $P_2(\Psi)$. This corresponds to a Laplacian prior over the number of edges for each node and is given by $P_2(\Psi|\lambda) \propto \exp(-\lambda \sum_{i=1}^{n} (\Psi_{ii} - \epsilon))$ where $(\Psi_{ii} - \epsilon)$ denotes the number of edges for the i^{th} node and λ is equivalent to the regularization parameter controlling the sparsity of the connecting edges.

4.3. Inference in TiWnet

To enable Bayesian inference in our model, we make use of the likelihood given in Equation (3) and the two-component prior described in Section 4.2. For analyzing the posterior of Ψ we iteratively sample one row/column in the upper triangle part of Ψ, conditioning on the rest, using a Metropolis-within-Gibbs sampler.

The proposal distribution defines a symmetric random walk on the row/column vector taking values in $\{-1, +1, 0\}$ by randomly selecting one value and resampling it with identical probability to the two other possible values. After updating the i-th row/column in the upper triangle matrix and copying the values to the lower triangle, the corresponding diagonal element is imputed deterministically as $\Psi_{ii} = \sum_{j \neq i} |\Psi_{ij}| + \epsilon$. This creates $\widetilde{\Psi}_{\text{proposed}}$ which is then accepted according to the usual Metropolis-Hastings equations based on the posterior ratio $P(\widetilde{\Psi}_{\text{proposed}}|\bullet)/P(\widetilde{\Psi}_{\text{old}}|\bullet)$. The acceptance threshold is given by just the posterior ratio since we implement a symmetric random walk Metropolis sampling. The entire Metropolis-within-Gibbs sampler is described in Algorithm 1.

Algorithm 1 Metropolis-within-Gibbs sampler

in i^{th} row/column vector in upper triangle of Ψ
1: Uniformly select index k, $k \in \{1, \cdots, i-1, i+1, \cdots, n\}$
2: Resample value at Ψ_{ik} by drawing with equal probability from $\{-1, +1, 0\}$
3: Set $\Psi_{ki} = \Psi_{ik}$ and update Ψ_{ii} and Ψ_{kk} (to ensure diagonal dominance). This is $\widetilde{\Psi}_{\text{proposed}}$
4: Compute $P(\tilde{\Psi}|\bullet) \propto \mathcal{L}(\widetilde{\Psi}) P_1(\Psi) P_2(\Psi)$
5: Calculate the acceptance threshold $\mathbf{a} = \min\left(1, \frac{P(\widetilde{\Psi}_{\text{proposed}}|\bullet)}{P(\widetilde{\Psi}_{\text{old}}|\bullet)}\right)$
6. Sample $\mathbf{u} \sim \text{Unif}(0, 1)$
7: **if** $(\mathbf{u} < \mathbf{a})$ accept $\widetilde{\Psi}_{\text{proposed}}$, **else** reject.

Since the proposal distribution, $\widetilde{\Psi}_{\text{proposed}}$, defines a symmetric random walk on set $\mathcal{A}$ consisting of diagonally-dominant matrices, one can reach any other element in $\mathcal{A}$ with non-zero probability after a sufficient number of $\frac{n(n-1)}{2}$ steps that account for number of elements in the upper triangle of Ψ. This construction ensures ergodicity in the Markov chain.

Note that the (usually unknown) degrees of freedom d in the shift- and scale-invariant likelihood (Equation (3)) appears only in the exponents and, thus, has the formal role of an annealing parameter. We use this property during the burn-in period, where d is slowly increased to "anneal" the system until the acceptance probability reaches below a certain threshold, and then the sampled Ψ-matrices are averaged to approximate the posterior expectation. If a truly sparse solution is desired, the annealing is continued until a network is "frozen".

Implementation & complexity analysis. Presumably the most efficient way to recompute $P(\widetilde{\Psi}|\bullet)$ after a row/column update of Ψ is through the identity: $\det(\widetilde{\Psi}) = (\det(\Psi)/\mathbf{1}^t\Psi\mathbf{1})\cdot n$ (McCullagh, 2009). Assume now we have a QR factorization of Ψ_{old} before the update. Then the new $\Psi = \Psi_{\text{old}} + \boldsymbol{v}_i\boldsymbol{v}_i^t + \boldsymbol{v}_j\boldsymbol{v}_j^t$ where i,j are the row/column indices of Ψ_{old} to be updated along with the corresponding diagonal elements and this accounts for two rank-one updates. Thus the QR factorization of the new $\widetilde{\Psi}$ can also be computed in $O(n^2)$ time and $\det(\widetilde{\Psi})$ is then derived as $\prod_i R_{ii}$. The trace $\text{tr}(\widetilde{\Psi}D)$ is also computed in $O(n^2)$ time, as it is the sum of the *element-wise* products of the entries in $\widetilde{\Psi}$ and D. An entire sweep of the Gibbs sampler involves n such updates adding up to a total time complexity of $O(n^3)$ per sweep. It is clear that this scaling behavior is prohibitive for very large matrices, but matrices of size in the hundreds can be easily handled, and for larger matrices with a "complex" inverse covariance structure the statistical significance of the reconstructed networks is questionable anyway, unless a really huge number of measurements is available. Moreover there is an elegant way of avoiding such large matrices by reconstructing *module networks* as outlined in the next section.

5. Inferring Module Networks

A particularly interesting property of TiWnet is its applicability to learning module networks. We define a module as a completely connected sub-graph, forming nodes in a module network. As a motivating example we refer to our gene-expression example of $X_{n\times d}$ where the measurements consist of d microarrays for n genes. In usual situations having far more objects than measurements, one should not be too optimistic to reconstruct a meaningful network, in particular if the measurements are noisy and if the underlying network has "hubs"– nodes with high degrees. Generally when the node neighborhoods are small, networks can be learnt well whereas when the neighborhoods tend to grow larger as in the case with hubs, learning networks gets difficult due to the higher-order dependencies existing between nodes. Unfortunately, both high noise and existence of hubs are common in such data. To address these issues, we present the computationally-attractive method of initially creating clusters of objects, that we connote as modules, over which networks are learnt. Considering the gene-expression example, there are usually groups of genes which have highly correlated expressions and can often be jointly represented by one cluster without losing too much relevant information, due to high noise. To create clusters, we begin with

the d-dimensional expression profile vectors, $\boldsymbol{x} \in \mathbb{R}^d$, of the n genes and use a mixture model to cluster these expression vectors into "modules", reducing n to the effective number of modules. The mixture model density is given by $p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k p_k(\boldsymbol{x})$ where π_k is the mixing coefficient and $p_k(\boldsymbol{x})$ is the component distribution of the k^{th} module. The link to learn networks on top of these modules goes via kernels defined on probability distributions. We can use kernels like *Bhattacharyya kernel*: $K_B(k,j) = \int (\sqrt{p_k(\boldsymbol{x})}\sqrt{p_j(\boldsymbol{x})})\, d\boldsymbol{x}$ (Jebara et al., 2004) over the component distributions of the modules to compute an inner-product matrix of the modules. Network inference is then performed using this resulting inner-product matrix.

Usually, there is no information available about the origin of the underlying space, and by reconstructing networks from such kernels we heavily rely on the geometric invariance encoded in the TiWnet model. This elegant solution for inferring module networks overcomes statistical problems, and is also a principled way of applying the TiWnet to large problem instances. An example of this strategy is presented in Section 6.

6. Experiments

Toy examples. The TiWnet is compared with two lasso-based methods: "graph lasso" (Friedman et al., 2007) and "logistic lasso" (Kolar et al., 2010b) on artificial data. For this we implemented a data generator that mimics the assumed generative model as shown in Figure 1. First, a sparse inverse covariance matrix $\Psi \in \mathbb{R}^{n\times n}$ with $n = 25$ is sampled. Networks with uniformly sampled node degrees are relatively easy to reconstruct for most methods, while networks with "hubs" are better suited for showing differences. Hubs are nodes with high degrees that appear naturally in many real networks since they often are scale-free i.e. their node degrees follow a power law. We simulate such networks by drawing node degrees from a Pareto distribution and use these values as parameters in a binomial model for sampling 0/1 entries in the rows/columns of Ψ. The sign of these entries is randomly flipped, and scaled with samples from a Gamma distribution. The diagonal elements are imputed as the row-sums of absolute values plus some small constant to ensure full rank. We draw d vectors $\boldsymbol{x}_i^o \in \mathbb{R}^n$ from $\mathcal{N}(\mathbf{0}_n, \Psi)$, and arrange them as columns in X^o. $S^o = X^o(X^o)^t$ is then a central Wishart matrix. To study the effect of biased measurements, we randomly generate biases $b_{(i=1,\ldots,d)}$, resulting in the mean-shifted vectors $\boldsymbol{x}_i$ in Fig. 1. The resulting matrix S is non-central Wishart with non-centrality matrix $\Theta = \Sigma^{-1}MM^t$, and $M = \mathbf{1}\boldsymbol{b}^t$.

In a **first experiment** we tune all parameters (the ℓ_1 regularization parameter in the lasso-based methods and the corresponding λ-parameter in the prior $P_2(\Psi)$ of TiWnet) to maximize the Wishart-likelihood on an independent test matrix S^o_{test} sampled from the same underlying normal distribution used for S^o. This enables the application of predictive likelihood for model selection on which our synthetic experimental results are based (Fig. 3). The optimal parameters are found by averaging over 20 such training/test instances. The quality of the reconstructed networks is measured as follows: A binary vector $\boldsymbol{l}$ of size $n(n-1)/2$ encoding the presence of an edge in the upper triangle matrix of Ψ is treated as "true" edge labels, and this vector is compared with a vector $\hat{\boldsymbol{l}}$ containing the absolute values of elements in the reconstructed $\hat{\Psi}$ after zeroing those elements in $\hat{\boldsymbol{l}}$ which are not sign-consistent with those in Ψ. The agreement of $\boldsymbol{l}$ and $\hat{\boldsymbol{l}}$ is measured with the F-score,

i.e. the highest harmonic mean of precision and recall under thresholding the elements in $\hat{\boldsymbol{l}}$. The top left panel in Fig. 3 shows boxplots of F-scores obtained in 50 experiments with randomly generated Ψ-matrices for "graph lasso", "logistic lasso" and TiWnet, and the top right panel shows the outcome of a statistical test for significance of the differences, see figure caption for further details. From the results we conclude that (i) column bias indeed severely reduces the performance of methods relying on a central Wishart distribution (ii) column centering does not overcome this problem and (iii) TiWnet performs as well as "graph lasso" on the original (not shifted) data. For (iii) "graph lasso" should be highly appropriate since the model assumptions are exactly met.

Analyzing the reconstructed networks in the central row of Figure 3, it is obvious that the original "graph lasso" network is very dense, and that thresholding the edge weights is essential for a high F-score. The TiWnet representing averaged $\hat{\Psi}$-matrices from the MCMC sampler is also dense, but it seems that most of the "true" edges are clearly accentuated. Further studying this effect leads us to a **second experiment**, where we directly compare the lasso-type networks reconstructed using a sequence of ℓ_1 regularization parameters with the "frozen" TiWnet after annealing the Markov chain. In this comparison, however we do *not* allow for further thresholding the edge weights when computing the F-score (i.e. we replace the entries in $\hat{\boldsymbol{l}}$ by their sign). The bottom left panel in Figure 3 shows that TiWnet clearly outperforms the lasso methods, even though the best F-score among all tested ℓ_1 regularization parameters is reported. We conclude that the lasso methods have problems to reconstruct such networks with hubs, and that the annealing mechanism in our MCMC sampler produces sparse networks of very high quality. To test the dependency of these results on the validity of the model assumptions, in a **third experiment** we substitute the Gaussian to produce X^o with a Student-t. The resulting plot of F-scores (Figure 4) has the same structure as in Figure 3 (top row) with globally smaller median values. Thus, these results obtained above do not qualitatively vary under such model mismatches.

A Module network of *Escherichia coli* genes. For inferring module networks in a biological context, we applied the TiWnet to a published dataset of promoter activity data from ≈ 1100 *Escherichia coli* operons (Zaslaver et al., 2006). The promoter activities were recorded with high temporal resolution as the bacteria progressed through a classical growth curve experiment experiencing a "diauxic shift". Certain groups of genes are induced or repressed during specific stages of this growth curve. Cluster analysis of the promoter activity data was performed using a spherical Gaussian mixture model with shared variance σ: $p(x) = \sum_k \pi_k \mathcal{N}(x|\mu_k, \sigma)$ along with a Dirichlet-process prior to automatically select the number of clusters. This revealed the presence of 14 distinct gene clusters (see expression profiles of nodes in Figure 5). Network inference with TiWnet was carried out on a Bhattacharyya kernel K_B computed over the Gaussian clusters where $K_B(k,j) = \exp^{-||\mu_k - \mu_j||^2/\,8\sigma^2}$ (see Jebara et al. (2004)). When the clusters were analyzed, genes known to be co-regulated were predominantly found in the same or nearby clusters with positive partial correlations. For example, during the diauxic shift experiment, the transcriptional activator *CRP* induces a certain set of genes in a specific growth phase (Keseler et al., 2011). Strikingly, of the 72 known *CRP* regulated operons in the dataset, 43 genes are found in cluster 6 or the four neighboring clusters (3,9,11,13). Likewise, genes involved in specific molecular functions (those coding for proteins involved in amino acid biosynthesis pathways) were found in close proximity in the network, for example in nodes 1

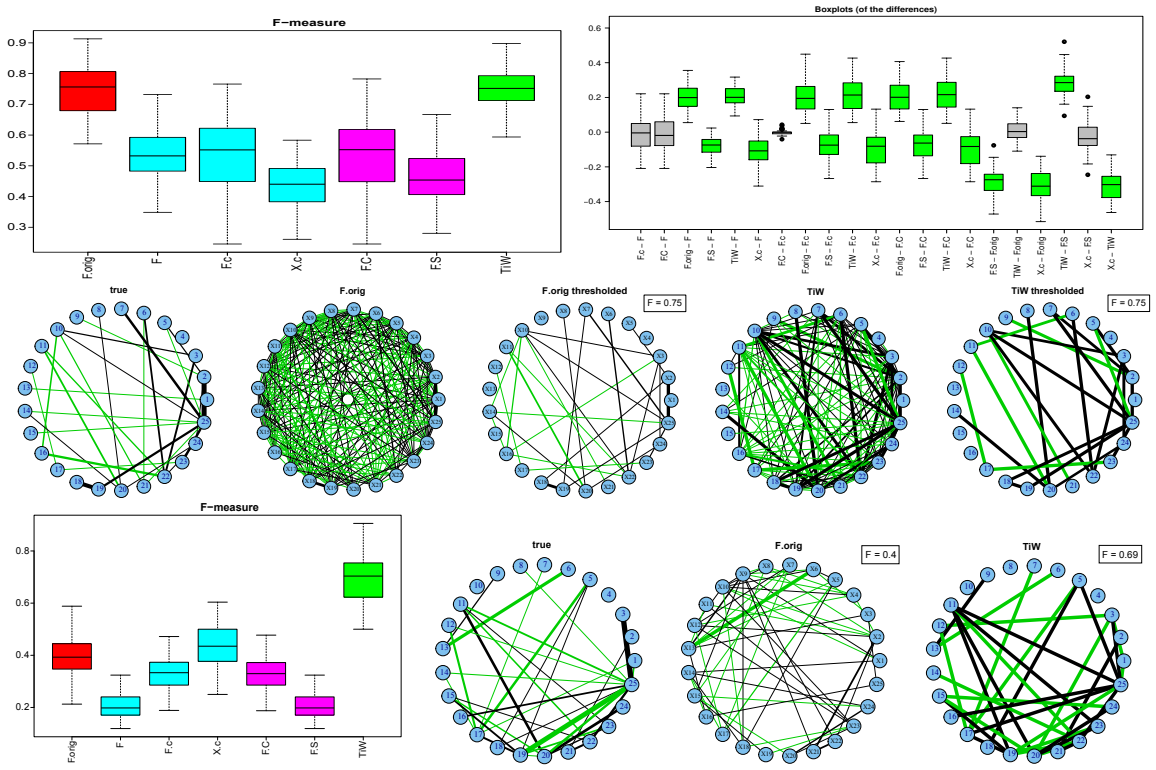

Figure 3: **Top.** Left: structure recovery measured by the maximum F-score under thresholding the edge weights. The methods are "graph lasso" (Friedman et al., 2007, 2009) (prefix "F"), "logistic lasso" (Kolar et al., 2010b) ("X") and TiWnet. Box colors encode the data used: red = original X^o, cyan = X, magenta = S, green = D. Suffixes denote centering used: no suffix means directly using the empirical covariance computed from X^o; ".c": column-centered mean-shifted X; ".C": column-centered S; ".S": uncentered S. Right: boxplot of pairwise differences. A green box means that a statistical test (Friedman with post-hoc) assigns a p-value < 0.05 to this comparison. **Middle.** Example networks. Black/green edges = positive/negative partial correlation. Left to right: "true" graph, "graph lasso" network with parameters tuned to maximize test likelihood, optimally thresholded "graph lasso", TiWnet from averaged sampled matrices, optimally thresholded TiWnet. **Bottom.** Structure recovery as in top left, but *without* thresholding the estimated edge weights. The networks show one example of a "true" graph, the best graph found with "graph lasso" and the annealed (or MAP-estimate) TiWnet respectively.

and 2 (Figure 5). Physiologically, this co-regulation makes sense since protein biosynthesis (carried out by the ribosome) depends on a constant supply of synthesized amino acids. Thus TiWnet can successfully identify connections between genes co-regulated by the same molecular factor, or are involved in interlinked molecular processes.

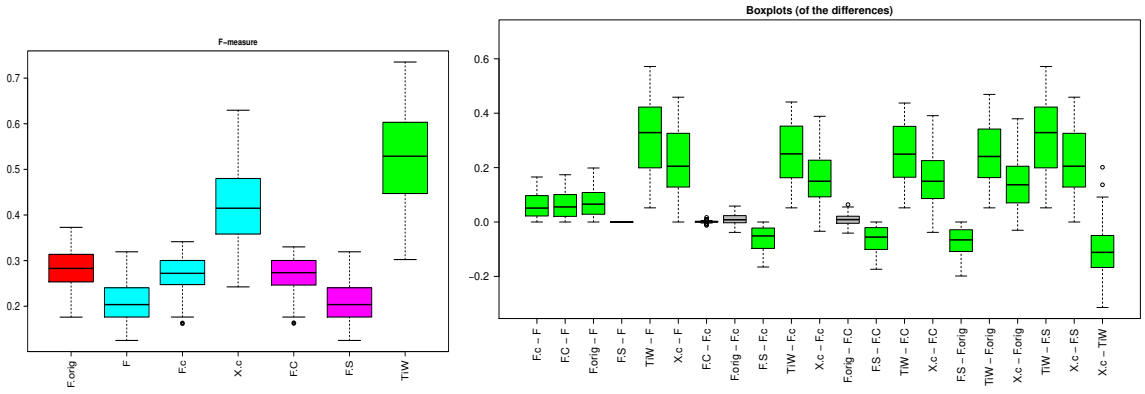

Figure 4: Results of artificial data using a multivariate Student-t distribution in three degrees of freedom instead of a normal distribution to generate the columns in X^o. **Left:** structure recovery measured by the F-score on binarized edge weights. The methods are "graph lasso" (Friedman et al., 2007, 2009) (prefix "F"), "logistic lasso" (Kolar et al., 2010b) ("X") and TiWnet ("TiW"). Box colors encode the data used: red = original X^o, cyan = "mean-shifted" X, magenta = similarities S, green = distances D. Suffixes denote centering used: no suffix means directly using the empirical covariance computed from X^o; ".c": column-centered mean-shifted X; ".C": column-centered S; ".S": uncentered S. **Right:** boxplot of pairwise differences. A green box means that a statistical test (Friedman with post-hoc) assigns a multiple testing corrected p-value < 0.05 to this comparison.

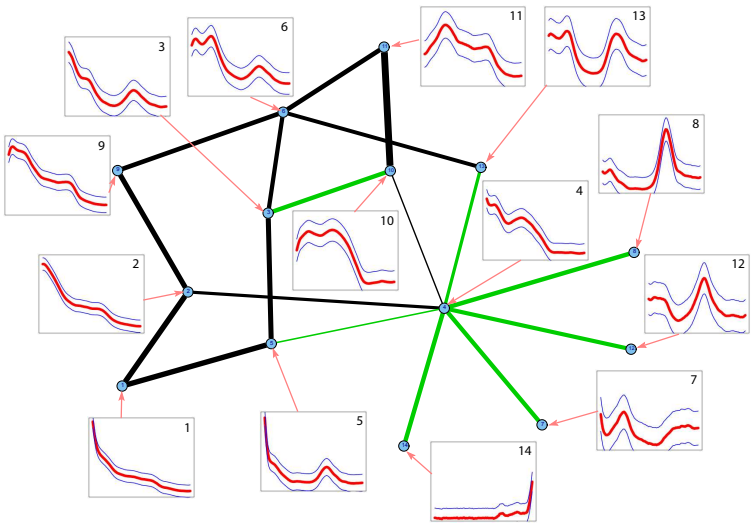

Figure 5: Module Network of *Escherichia coli* Genes.

"Landscape" of chemical compounds with *in vitro* activity against HIV-1. As a second real-world example TiWnet is used to reconstruct a network of chemical compounds. We enriched a small list of compounds identified in an AIDS antiviral screen by

NCI/NIH available at http://dtp.nci.nih.gov/docs/aids/searches/list.html#NPorA with all currently available anti-HIV drugs, yielding a set of 86 compounds. *Chemical hashed fingerprints* were computed from the chemical structure of the compounds that was encoded in SMILES strings (Weininger, 1988). The *Tanimoto* kernel, a similarity matrix S of inner-product type, is constructed by the pairwise Tanimoto association scores (Rogers and Tanimoto, 1960) between the compounds. Since the geometric position of the underlying Euclidean space is unclear, we again relied heavily on the geometric invariance inherent in TiWnet. The resulting network (Figure 6) shows several disconnected components which nicely correspond to chemical classes (the node colors). Currently available anti-HIV drugs are indicated by their chemical and commercial names alongside their 2D-structures depicting the chemical similarity underlying this network. These drugs belong to the functional groups "Nucleoside reverse transcriptase inhibitors (NRTI)", "Non-nucleoside reverse transcriptase inhibitors (NNRTI)", "Protease inhibitors", "Integrase inhibitors", or "Entry inhibitors", and most compounds of a certain functional type cluster together in the graph. Medically, this network can be very useful to predict "cross resistance" between resistant HIV-1 variants and drugs and is especially distinctive for NRTIs. The pairs *lamivudine-emtricitabine*, *tenofovir-abacavir*, and *d4T-zidovudine(ZDV)* show almost the same resistance profiles (Johnson et al., 2010). This similarity is very well reflected by our network where these pairs are in close proximity.

It is worth noting that "graph lasso" has similar difficulties on this dataset as in the toy examples. When following the solution path by varying the penalty parameter, it is difficult to find a good compromise between sparsity and connectivity: either the obtained graphs are very dense being difficult to plot and harder to interpret, or are increasingly sparse in which, however, several interesting structural connections are lost since many singleton nodes are created (for a graphical depiction, refer Figures 1-3 in the Supplement available at http://bmda.cs.unibas.ch/TiWnet).

7. Conclusion

The TiWnet model is a fully probabilistic approach to inferring GGMs from pairwise Euclidean distances obtained from inner-product similarity matrices (i.e. kernels) of n objects. Traditional models for reconstructing GGMs, for example lasso-type methods, are based on the central Wishart likelihood parametrized by the inverse covariance, and sparsity of the latter is usually enforced by some penalty term. Assuming a central Wishart, however, is equivalent to assuming that the origin of the coordinate system is known. If these methods use on input only kernel matrices, then usually only the kernels' pairwise distance information is truly relevant. Since traditional methods solely rely on the origin implicitly encoded in any such kernel, they might generate biased networks. Our TiWnet method is specifically designed to work with pairwise distances since the likelihood used in inference depends only on these distances. Combining this likelihood with a prior suited for sparse network recovery, we are able to extract sparse networks using only pairwise distances. This property opens up a huge new application field for GGMs, because network inference can now be carried out on any such distance matrix induced by a Mercer kernel on graphs, probability distributions or more complex structures. We also present an efficient MCMC sampler for TiWnet making it applicable to medium-size instances, and the possibly re-

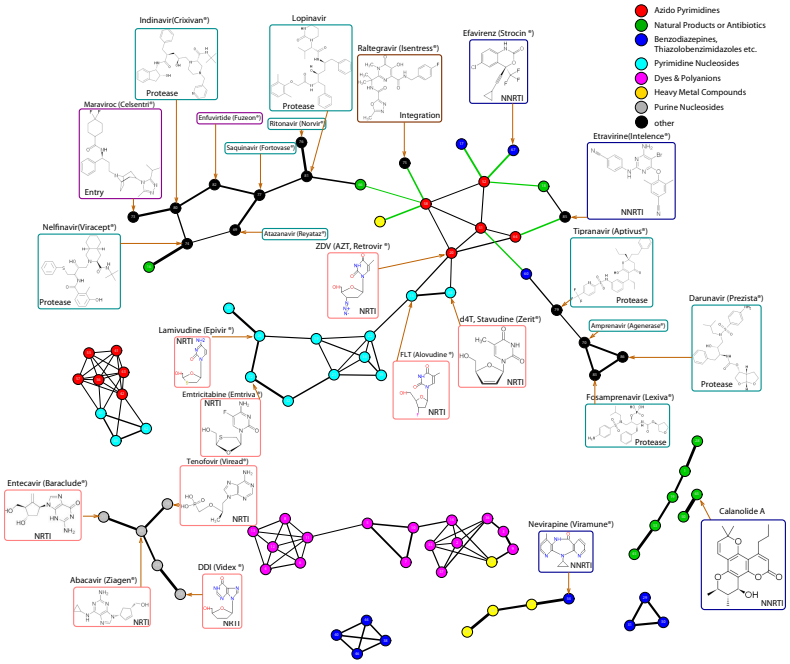

Figure 6: "Landscape" of Chemical Compounds with *In Vitro* Activity against HIV-1.

maining scaling issues may be overcome by inferring module networks using kernels defined on probability distributions over groups of nodes. Comparisons with competing methods demonstrate the high quality of networks obtained from TiWnet, evoking the effectiveness of working with pairwise distances. TiWnet is also robust to model mismatches unlike existing methods. The two real-world examples provide an insight into the huge variety of possible applications.

References

Animashree Anandkumar, Vincent Tan, and Alan S. Willsky. High-dimensional graphical model selection: Tractable graph families and necessary conditions. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 1863–1871. 2011.

José A. Díaz-García, Ramón Gutierrez Jáimez, and Kanti V Mardia. Wishart and Pseudo-Wishart distributions and some applications to shape theory. *J. Multivariate Anal.*, 63: 73–87, 1997.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9:432–441, 2007.

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. glasso: Graphical lasso - estimation of gaussian graphical models. R package version 1.4, 2009.

Tony Jebara, Risi Kondor, and Andrew Howard. Probability product kernels. *Journal of Machine Learning Research*, 5:819–844, 2004.

Jason K. Johnson, Dmitry M. Malioutov, and Alan S. Willsky. Walk-summable gaussian networks and walk-sum interpretation of gaussian belief propagation. *Technical Report - 2650, LIDS, MIT*, 2005.

Jason K. Johnson, Dmitry M. Malioutov, and Alan S. Willsky. Walk-sum interpretation and analysis of gaussian belief propagation. In *Advances in Neural Information Processing Systems 18*, pages 579–586. MIT Press, 2006.

Victoria A. Johnson, Francoise Brun-Vezinet, and Bonaventura Clotet et al. Update of the drug resistance mutations in HIV-1: Dec 2010. *Topics in HIV medicine*, 18(5):156–163, 2010.

Ingrid M. Keseler, Julio Collado-Vides, and Alberto Santos-Zavaleta et al. Ecocyc: a comprehensive database of Escherichia coli biology. *Nucleic Acids Research*, 39:D583–D590, 2011.

Mladen Kolar, Ankur P. Parikh, and Eric P. Xing. On sparse nonparametric conditional covariance selection. *The 27th International Conference on Machine Learning*, pages 559–566, 2010a.

Mladen Kolar, Le Song, Amr Ahmed, and Eric P. Xing. Estimating time-varying networks. *Annals of Applied Statistics*, 4(1):94 – 123, 2010b.

Peter McCullagh. Marginal likelihood for distance matrices. *Statistica Sinica*, 19:631–649, 2009.

Nicolai Meinhausen and Peter Bühlmann. High dimensional graphs and variable selection with the Lasso. *Annals of Statistics*, 38:1436–1462, 2006.

T. J. Mitchell and J. J. Beauchamp. Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, 83(404):pp. 1023–1032, 1988.

David J. Rogers and Taffee T. Tanimoto. A computer program for classifying plants. *Science*, 132:1115–1118, 1960.

Granville Tunnicliffe-Wilson. On the use of marginal likelihood in time series model estimation. *Journal of the Royal Statistical Society, Series B*, 51:15–27, 1989.

Julia E. Vogt, Sandhya Prabhakaran, Thomas J. Fuchs, and Volker Roth. The Translation-invariant Wishart-Dirichlet Process for Clustering Distance Data. In *Proc. of the 27th International Conference on Machine Learning*, 2010.

David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences*, 28(1):31–36, 1988.

Alon Zaslaver, Anat Bren, and Michal Ronen et al. A comprehensive library of fluorescent transcriptional reporters for Escherichia coli. *Nat Meth*, 3(8):623–8, August 2006.

Shuheng Zhou, John Lafferty, and Larry Wasserman. Time varying undirected graphs. *Machine Learning*, 83:295–319, 2010.