Abstract

We address in this study the problem of modeling weighted networks through generalized stochastic block models. Stochastic block models, and their extensions through mixed-membership versions, are indeed popular methods for network analysis as they can account for the underlying classes/communities structuring real-world networks and can be used for different applications. Our goal is to develop such models to solve the weight prediction problem that consists in predicting weights on links in weighted networks. To do so, we introduce new mixed-membership stochastic block models that can efficiently be learned through a coupling of collapsed and stochastic variational inference. These models, that represent the first weighted mixed-membership stochastic block models to our knowledge, can be deployed on large networks comprising millions of edges. The experiments, conducted on diverse real-world networks, illustrate the good behavior of these new models.

1 Introduction

Link prediction in networks is a well-known problem that has been addressed by many studies. If knowing that a link relates two nodes in a network is important, the intensity of each link plays a major role in many situations. For example, in epidemiology, the number of contacts between two persons is an important factor to accurately estimate the probability of contagion between two persons. Similarly, to understand the population dynamics between two cities, it is not sufficient to know that there is a motorway or an airline relating them, it is also necessary to know the number of vehicles or passengers that transit between them. In the fields of economy and finance, to estimate whether a company will be controlled by another company which has recently acquired part of its shares, it is important to know the actual number of shares acquired. In all these examples, the relations between the entities involved (persons, cities, companies) can be modeled by weighted graphs, in which the weight on each link represents the intensity of the relationship between the nodes. Inferring the values of the weights between nodes in such graphs is known as the weight prediction problem. In this paper, we are interested in the exploration and analysis of weighted networks through the discovery of their latent structure and through the possibility to predict, from such structures, edge weights.

If weights in weighted graphs can sometimes take both positive and negative values, as in signed social networks for representing approval/disapproval, like/dislike or trust/distrust, most weighted networks rely on positive integers. The prevalence of this type of networks may be explained by the fact that many weighted networks are the result of the superposition, over time, of atomic, binary interactions. In communication networks for example, as in email networks, edges are weighted according to the number of exchanges between nodes, the atomic interaction being a single exchange. Similarly, in co-authorship networks, edges are weighted according to the number of collaborations between authors [New01], while in text mining and natural language processing applications word graphs are weighted on the basis of the number of times the words co-occur (in a sentence, a paragraph or a document).

Two main approaches have been proposed to solve the weight prediction problem in networks. In the first type of approaches, one finds methods that assume that the weight of a link is correlated with the similarity between the nodes of the link, this similarity being based on neighboring nodes [ZMY+15, ZXZ16]. If the as-
assumption "a node behaves like its neighbors" is used, to
different extent, in network studies, it is however not suf-
ficient to explain all the observed interactions between
nodes. Several researchers have thus adopted a second
type of approaches, based on generative models within
well-defined probabilistic frameworks. Such models aim
at making explicit how links, and their weights, are pro-
duced. Among such generative models, stochastic block
models and their extensions through mixed-membership
stochastic block models have received particular atten-
tion [KN11, ABFX09, KER08, FCX15] as they can ac-
count for the underlying classes that structure real-world
networks. Nevertheless, most models proposed so far
have been devoted to unweighted networks and, to our
knowledge, only two models in the stochastic block
model family have been proposed for weighted graphs:
The latent block structure model of [AJC14] and the
weighted stochastic block model of [Pei18]. These two
models however suffer from the same drawback as stan-
dard stochastic block models, namely the fact that a node
can belong to only one class, which is not realistic for
many networks and can be corrected through mixed-
membership block models.

We thus propose in this study:

1. New mixed-membership block models that can
   solve the weight prediction problem on networks in
   which weights are positive integers;

2. A scalable inference method, based on a combina-
tion of collapsed and stochastic variational infer-
ence, for deploying the above models on large net-
works;

3. An experimental illustration of the behavior of these
   models on several real-world networks.

The remainder of the paper is organized as follows: Sec-
tion 2 describes related work; Section 3 then presents
the weighted mixed-membership stochastic block mod-
els we retained while Section 4 details its inference. Sec-
tion 5 illustrates the behavior of the proposed models
on several real-world networks. Finally, Section 6 con-
cludes the study.

2 Related work

Weighted versions of the stochastic block model have
been introduced firstly in [MRV10] and then in [AJC14]
who proposed a model referred to as WSBM. WSBM can
be seen as a special case, in which nodes are constrained
to belong to only one latent class, of the weighted mixed-
membership stochastic block model we introduce in this
paper, as this latter model can assign nodes to several
classes. More recently, an extended version of WSBM
has been presented in which different kernels can be used
to model different types of weights [Pei18]. An effi-
cient Markov Chain Monte Carlo (MCMC) method is
used for inference. If this type of models is interest-
ing, it nevertheless relies again on the assumption that a
node belongs to only one class, which may be inappro-
priate for real world networks. Furthermore, unlike mixed-
membership stochastic block models, the lack of a hier-
archical prior structure does not allow one to rely on effi-
cient non-parametric extensions (hence the use of costly
model selection techniques for non-parametric versions).

Similar to the model we introduce here, count processes
with Poisson distributions and Gamma conjugate pri-
or have been previously studied, notably by Zhou et
al. [ZC12, ZC15]. The relation of such processes
with Negative Binomial processes is well-known and
has been highlighted by these authors who applied these
processes for topic modeling with the Beta-Gamma-
Gamma-Poisson model (EPM) ([ZHDC12]) that relies
on MCMC inference. They also applied them for
overlapping community detection and link prediction
[Zho15]. The main difference between this model and
the one we introduce in the next section is that in the
former weights are distributed as Poisson variables that
correspond to sums of class-dependent latent factors
(Eq. 1 of [ZHDC12]) while, in the latter, weights are
distributed as a sum of Poisson variables weighted by
class-membership factors (Eq. 1 below).

3 Mixed-membership stochastic block
models and (un)weighted graphs

As usual, we consider here that a network is represented
by a graph \( G = (V, E) \) where \( V \) is the set of nodes such
that \( N = |V| \) and \( E \) the set of edges. We furthermore
consider the matrix \( Y = (y_{ij})_{1 \leq i, j \leq N} \) such that \( y_{ij} = 0 \)
if \( (i, j) \notin E \) and \( y_{ij} \in \mathbb{Z}_{>0} \) otherwise. When the net-
work is unweighted, \( y_{ij} \in \{0, 1\} \) and \( Y \) is the adjacency
matrix.

3.1 Weighted version of MMSB (WMMSB)

Mixed-membership stochastic block models extend
stochastic block models [ABFX09] by allowing nodes to
"belong" to several blocks (or classes) through a given
(usually Dirichlet) probability distribution. Prior to gen-
erate a link between two nodes, a particular class is se-
lected for each node. The link is then generated ac-
cording to a probability distribution \( F \), sometimes re-
ferrred to as the kernel distribution, that depends on the
selected classes. The generative process behind such
models can be summarized as: (a) For each node \( i \), draw
\( \theta_i \sim \text{Dir}(\alpha) \), where \( \theta_i \) and \( \alpha \) are \( K \)-dimensional vectors, \( K \) denoting the number of classes considered; (b) Generate two sets of latent class memberships, \( Z_\rightarrow = \{ z_{i \rightarrow j} \sim \text{Cat}(\theta_i), 1 \leq i, j \leq N \} \) and \( Z_\leftarrow = \{ z_{j \leftarrow i} \sim \text{Cat}(\theta_j), 1 \leq i, j \leq N \} \), with categorical draws; (c) Generate or not a link between two nodes \((i, j)\) according to \( y_{ij} \sim F(\phi_{z_{i \rightarrow j} z_{i \leftarrow j}}) \), where \( F \) is a distribution in the exponential family and \( \phi_{z_{i \rightarrow j} z_{i \leftarrow j}} \), a variable, usually drawn from a conjugate distribution, that represents the relations between classes. A standard choice for \( F \) is the Bernoulli distribution, \( \phi \) being the conjugate Beta distribution: \( y_{ij} \sim \text{Bern}(\phi_{z_{i \rightarrow j} z_{i \leftarrow j}}) \), \( \phi_{kk'} \sim \text{Beta}(\lambda_0, \lambda_1) \). We refer to this model as the MMSB model.

A natural way to model weights that correspond to positive integers is to use Poisson distributions. Considering a conjugate Gamma distribution for \( \phi \), we define a first weighted extension of MMSB, which we will refer to as WMMSB, through:

\[
\begin{align*}
\theta_i & \sim \text{Dir}(\alpha), \quad z_{i \rightarrow j} \sim \text{Cat}(\theta_i), \quad z_{j \leftarrow i} \sim \text{Cat}(\theta_j), \\
y_{ij} & \sim \text{Poi}(\phi_{z_{i \rightarrow j} z_{i \leftarrow j}}), \quad \phi_{kk'} \sim \text{Gamma}(r, \frac{1-p}{p}),
\end{align*}
\]

where \( r \) and \( \frac{1-p}{p} \) are the shape and rate parameters of the Gamma distribution. Note that both directed and undirected graphs can be considered with the above formulation, the matrix \( \Phi = (\phi_{kk'})_{k,k'\in\{1,...,K\}} \) being symmetric for undirected graphs.

The Poisson-Gamma combination in WMMSB is interesting as it directly leads to: \( y_{ij} | Z \sim \text{NB}(r, \frac{1-p}{p}) \), where \( \text{NB} \) denotes the negative binomial distribution. This latter distribution allows one to represent overdispersed count data and has been used in different contexts. Lastly, by marginalizing over the variables \( z_{i \rightarrow j} \) and \( z_{i \leftarrow j} \), which take values in \( \{1, \cdots, K\} \), one obtains:

\[
y_{ij} | \theta, \Phi \sim \sum_{1 \leq k, k' \leq K} \theta_{ik} \theta_{jk'} \text{Poi}(\phi_{kk'}). \tag{1}
\]

### 3.2 A Beta-Gamma augmentation (WMMSB-bg)

The generative process for WMMSB defined above assumes that the parameters of the Poisson distributions used to generate links are drawn from the same Gamma distribution. Having a unique prior over these parameters however limits the ability of the model to capture the variance in the relations between the latent classes. Hierarchical extensions can be used here to have a better representation of the classes and the relations between them through class-dependent shape and rate parameters in the above Gamma distribution. We propose here to model the class-dependent shape parameter with another Gamma distribution and the class-dependent rate parameter with a Beta prior:

\[
r_{kk'} \sim \text{Gamma}(c_0 r_0, 1/c_0), \quad p_{kk'} \sim \text{Beta}(c, c(1-c)).
\]

The variable \( y_{ij} \) is again distributed according to a negative binomial distribution, of the form:

\[
y_{ij} | Z \sim \text{NB}(r_{z_{i \rightarrow j} z_{i \leftarrow j}}, p_{z_{i \rightarrow j} z_{i \leftarrow j}}). \tag{2}
\]

As one can note, and contrary to WMMSB, the parameters of the negative binomial distribution depend this time on the classes selected for each node, meaning that classes now play a prominent role in the model. We will refer to this model as WMMSB-bg.

The above extension exploits again the conjugacy of the distributions considered and is reminiscent of the Beta-Gamma-Gamma-Poisson model [ZHDC12] and the Gamma-Negative Binomial process [ZC15]. However, as for most hierarchical Bayesian models, exact inference is intractable and one must resort to approximate inference. The next section describes the variational inference scheme we have followed for that.

### 4 Inference

Variational Inference (VI) and Markov Chain Monte Carlo (MCMC) are two popular methods for learning the parameters of Bayesian models that (roughly speaking) realize a trade-off between time complexity (in particular wrt the size of the training set), one of the main advantages of VI, and accuracy in approximating the posterior distribution, one of the main advantages of MCMC. We have chosen the former as we are interested in a version of our model that scales to large datasets. Several studies have tried to scale MCMC to large datasets, using either, as far as we know, divide-and-conquer or subsampling approaches. But, as noted in [RB17], divide-and-conquer approaches have yet to solve a recombination method (on MCMC estimates computed on subsets of the data), while subsampling approaches only guarantee control of the approximation of the true posterior distribution in few situations that are rarely satisfied in practice, thus losing the main advantage of MCMC methods. Collapsed variational Bayes inference presents the advantage, over standard variational inference, to rely on weaker assumptions and has proven to be efficient on the latent Dirichlet allocation model [TNW07]. Recent advances in stochastic variational inference [HBWP13], notably based on well-designed sampling techniques [GB13, KGBS13], have furthermore shown that it is possible to speed-up (collapsed) variational inference with online updates based on minibatches. Coupling collapsed and stochastic variational inference thus leads here to an efficient inference method that can be used on large networks.
We first provide below the results obtained through collapsed variational inference for all the above models. A detailed derivation of these results is given in Appendix A. We then describe how stochastic variational inference is used on these models.

### 4.1 Collapsed variational inference

In the remainder, we use the notation \( n^{-ij} \) to indicate that the superscript \( ij \) is excluded from the underlying count variable, and \( n_i \) to indicate a sum over the dotted subscript index. Furthermore, \( \Pi \) will denote the model parameters: \( \Pi = (\Theta, \Phi, Z) \) for MMSB and WMMSB, where \( \Theta \) is a \( K \times N \) matrix, \( \Phi \) a \( K \times K \) matrix and \( Z \) an \( N \times N \) matrix, and \( \Pi = (\Theta, \Phi, Z, R, P) \) for WMMSB-bg, where \( R \) and \( P \) are \( K \times K \) matrices. Lastly, \( \Omega \) will denote the hyperparameters \( \Omega = (\alpha, \lambda_0, \lambda_1) \) for MMSB, \( \Omega = (\alpha, r, p) \) for WMMSB and \( \Omega = (\alpha, c_0, r_0, c, \epsilon) \) for WMMSB-bg.

From Jensen’s inequality, for any distribution \( q \), one has:

\[
\log p(Y|\Omega) \geq \mathbb{E}_q[\log p(Y, \pi, [\Omega]) + \log H(q(\pi))],
\]

where \( H(q) \) denotes the entropy. The goal of variational inference is then to find \( q(\pi) \) that maximizes the right-hand side of the above inequality, usually referred to as the Evidence Lower Bound (ELBO). In its collapsed version, following [TNW07], one weakens the mean-field assumption made over the variational distribution, leading to, for MMSB and WMMSB: 

\[
q(\Pi) = q(\theta, \Phi|Z)q(Z).
\]

For all \((i, j), q(z_{i \rightarrow j}, z_{i \leftarrow j}|\gamma_{ij})\) follows a multinomial distribution with parameters \( \gamma_{ijkk'}, (k, k') \in \{1, \cdots, K\}^2 \). The evidence is then lower bounded by:

\[
\log p(Y|\Omega) \geq \mathbb{E}_q[\log p(Y, Z) + \log H(Z)].
\]

The derivation of the collapsed variational updates is obtained by maximizing the ELBO w.r.t \( \gamma_{ijkk'} \):

\[
\frac{\partial \mathcal{L}_Z}{\partial \gamma_{ijkk'}} = \frac{\partial}{\partial \gamma_{ijkk'}} \left( \sum_{Z} \sum_{k_1} \sum_{k_2} q(Z^{-ij}) [\gamma_{ijkk'}] \right)
\]

\[
\left( \log p(Y, Z^{-ij}, z_{i \rightarrow j} = k_1, z_{i \leftarrow j} = k_2) + \log q(Z^{-ij}, z_{i \rightarrow j} = k_1, z_{i \leftarrow j} = k_2) \right)
\]

\[
= E_{q(Z^{-ij})}[p(Y, Z^{-ij}, z_{i \rightarrow j} = k, z_{i \leftarrow j} = k')],
\]

\[
+ H[Z^{-ij}] - \log(\gamma_{ijkk'}) + 1.
\]

By equating this derivative to zero, one obtains the following update:

\[
\gamma_{ijkk'} \propto \exp E_{q(Y^{-ij})}[\log P(z_{i \rightarrow j} = k, z_{i \leftarrow j} = k'|Y^{-ij}, Y^{-ij}, \Omega)],
\]

with \( P(z_{i \rightarrow j} = k, z_{i \leftarrow j} = k'|Y^{-ij}, Z^{-ij}, \Omega) \) being the collapsed Gibbs update of WMMSB. They take the form:

\[
P(z_{i \rightarrow j} = k, z_{i \leftarrow j} = k'|Y^{-ij}, Z^{-ij}, \Omega) \\
\propto (n^{-ij}_{ik} + \alpha_k)(n^{-ij}_{i \leftarrow jk'} + \alpha_{k'})
\]

\[
\text{NB} \left( y_{ij}; n_{kk'}^{ij} + r, p(n_{kk'}^{ij} + r) \right),
\]

with count statistics given by:

\[
n_{ik}^{-} = \sum_j \delta(z_{i \rightarrow j} = k),
\]

\[
n_{kk'}^{-} = \sum_{ij} y_{ij} \delta(z_{i \rightarrow j} = k, z_{i \leftarrow j} = k'),
\]

\[
n_{ik'}^{+} = \sum_{ij} \delta(z_{i \rightarrow j} = k, z_{i \leftarrow j} = k').
\]

By applying a first order Taylor expansion on Eq. (3), following [TNW07], one obtains:

\[
\gamma_{ijkk'} \propto (E_{q(Z^{-ij})}[n^{-ij}_{ik}] + \alpha_k)(E_{q(Z^{-ij})}[n^{-ij}_{i \leftarrow jk'}] + \alpha_{k'})
\]

\[
\text{NB} \left( y_{ij}; E_{q(Z^{-ij})}[n_{kk'}^{ij}] + r, p \right),
\]

with:

\[
p' = \frac{p}{E_{q(Z^{-ij})}[n_{kk'}^{ij}] + 1}.
\]

Finally, using a Gaussian approximation (as in e.g. [AWST09]), one obtains, setting \( E_{q(Z^{-ij})}[n^{-ij}_{ik}] = N_{ik}^{ij}, E_{q(Z^{-ij})}[n^{-ij}_{i \leftarrow jk'}] = N_{ik'}^{ij} \), and \( q(Z^{-ij})[n_{kk'}^{ij}] = N_{kk'}^{ij} \):

\[
N_{ik}^{ij} = \sum_{j; x_{ij}=x} \gamma_{ijkk'}, \quad N_{ik'}^{ij} = \sum_{i; x_{ij}=x} \gamma_{ijkk'},
\]

\[
N_{kk'}^{ij} = \sum_{ij; x_{ij}=x} \gamma_{ijkk'}, \quad E = \sum_{ij} y_{ij} \gamma_{ijkk'}. \quad (4)
\]

In this inference scheme, the parameters \( \gamma_{ij} \) are the local parameters while the count statistics \( N_{ij} \) represent the sufficient statistics and global counts. Finally, the model parameters can be recovered from their estimates as follows:

\[
\hat{\theta}_{ik} = \frac{N_{ik}^{ij} + N_{ik'}^{ij} + \alpha_k}{2N + \alpha},
\]

\[
\hat{\phi}_{kk'} = \begin{cases} 
\frac{N_{kk'}^{ij} + \lambda_1}{N_{kk'}^{ij} + \lambda_1 + p(N_{kk'}^{ij} + r)} & \text{for MMSB,} \\
\frac{N_{kk'}^{ij} + \lambda_1}{N_{kk'}^{ij} + \lambda_1 + p(N_{kk'}^{ij} + r)} & \text{for WMMSB.}
\end{cases}
\]

#### 4.1.1 Beta-Gamma augmentation

For WMMSB-bg, the derivation is slightly more complex and is fully detailed in Appendix A. We just provide here the main steps of this derivation.
We consider the following collapsed variational distribution for WMMSB-b: 
\[ q(\Pi) = q(\theta, \Phi, Z, R, P)q(Z)q(R)q(P), \]
with \( R = (r_{kk'}) \), \( P = (p_{kk'}) \), \( 1 \leq k, k' \leq K \). As before, \( q(z_{i \rightarrow j}, z_{i \leftarrow j} | \gamma_{ij}) \) is multinomial with parameter \( \gamma_{ij} \).

The same development as above applies for the parameters \( \gamma_{ijkk'} \), given here also by Eq. 3. Furthermore, the predictive link probability and \( \hat{\phi}_{kk'} \) now take the form:
\[
p(y_{ij}|Y^{\rightarrow ij}, Z^{\leftarrow ij}, z_{i \rightarrow j} = k, z_{i \leftarrow j} = k', \Omega) \sim \text{NB} \left( y_{ij}; N_{kk'}^{\rightarrow ij} + \mathbb{E}_q[p_{kk'}], \frac{\mathbb{E}_q[p_{kk'}]}{N_{kk'}^{\rightarrow ij} + 1} \right),
\]
and:
\[
\hat{\phi}_{kk'} = \frac{\mathbb{E}_q[p_{kk'}](N_{kk'}^{\rightarrow ij} + \mathbb{E}_q[r_{kk'}])}{N_{kk'} - \mathbb{E}_q[p_{kk'}] + 1}.
\]

Setting \( q(P) = p(P|Y, Z, \Omega) \) where \( p \) is the true distribution and exploiting the conjugacy of the Beta and the negative binomial distributions leads to a Beta distribution for \( p_{kk'} \): \( p_{kk'} \sim \text{Beta}(\epsilon c + N_{kk'}^+, c(1-\epsilon) + N_{kk'}^\phi) \), where:
\[
\mathbb{E}_q[p_{kk'}] = \frac{\epsilon c + N_{kk'}^{\phi}}{\epsilon c + N_{kk'} + c(1-\epsilon) + N_{kk'}^\phi} \mathbb{E}_q[r_{kk'}].
\]

Lastly, as for its true distribution, the variational distribution for \( r_{kk'} \) is taken in the Gamma family: \( q(r_{kk'}) \sim \text{Gamma}(a_{kk'}, b_{kk'}) \).

4.2 Stochastic variational inference with stratified sampling

Stochastic variational inference can then be used to optimize the collapsed ELBO through unbiased, yet noisy, estimates of its natural gradient computed on sampled data points. Different sampling strategies [GB13, KGBS13] have been proposed for that purpose. Following the study in [GB13], we rely here on stratified sampling that allows one to control the number of links and non-links used in the inference process. For each node \( i \), \( 1 \leq i \leq N \), one first constructs a set, denoted \( s_1 \), containing all the nodes to which \( i \) is connected to as well as \( M \) sets of equal size, denoted \( s_{1,m} \), \( 1 \leq m \leq M \), each containing a sample of the nodes to which \( i \) is not connected to\(^1\).

\(^1\)The sampling is here uniform over the nodes not connected to \( i \) with replacement; sampling without replacement led to poorer results in our experiments.

1. Sample a node \( i \) uniformly from all nodes in the graph; with probability \( \frac{1}{s_1} \), either select \( s_1 \) or any set from \( S_0^i \) (in the latter case, the selection is uniform over the sets in \( S_0^i \)). We will denote by \( s_i \) the set selected and by \( |s_i| \) its cardinality.

2. For each node \( j \in s_i \), compute \( \gamma_{ijkk'} \) through Eq. 3 and intermediate global counts according to:
\[
\hat{N}_{\omega \rightarrow ik}^\theta + = \frac{1}{|s_i|} \frac{1}{Cg(s_i)} \sum_{k' \neq ik'} \gamma_{ijkk'} \quad \hat{N}_{\omega \rightarrow jk'}^\theta = \frac{1}{Cg(s_i)} \sum_k \gamma_{ijkk'},
\]
\[
\hat{N}_{\omega \rightarrow ik}^\phi + = \frac{1}{|s_i|} \frac{1}{Cg(s_i)} \gamma_{ijkk'} \quad \hat{N}_{\omega \rightarrow jk'}^\phi = \frac{1}{|s_i|} \frac{1}{Cg(s_i)} \gamma_{ijkk'} u_{ij},
\]
where \( C \) is a constant that is 2 for undirected graphs and 1 for directed graphs and \( g(s_i) = \frac{1}{Nm} \) if \( s_i \in S_0^i \) and \( \frac{1}{N} \) otherwise.

3. Update of the global counts (online version of Eq. 4):
\[
\begin{align*}
N_{\omega \rightarrow ik}^\theta + &\leftarrow (1 - \rho_i^\theta) N_{\omega \rightarrow ik}^\theta + \rho_i^\theta \hat{N}_{\omega \rightarrow ik}^\theta, \\
N_{\omega \rightarrow jk'}^\theta + &\leftarrow (1 - \rho_i^\theta) N_{\omega \rightarrow jk'}^\theta + \rho_i^\theta \hat{N}_{\omega \rightarrow jk'}^\theta, \\
N_{\omega \rightarrow ik}^\phi + &\leftarrow (1 - \rho_i^\phi) N_{\omega \rightarrow ik}^\phi + \rho_i^\phi \hat{N}_{\omega \rightarrow ik}^\phi, \\
N_{\omega \rightarrow jk'}^\phi + &\leftarrow (1 - \rho_i^\phi) N_{\omega \rightarrow jk'}^\phi + \rho_i^\phi \hat{N}_{\omega \rightarrow jk'}^\phi.
\end{align*}
\]

4. \( \rho_i^\phi = \frac{1}{(\tau + \frac{1}{c})^k} \) with \( k \in (0.5, 1) \).

5. Go back to step 1 till convergence.

As one can note, the intermediate global counts correspond to a restriction, on minibatches, of the complete computation given in Eq. 4. The value of \( C \) is due to the fact that in undirected networks, each edge can be seen twice. The terms \( \frac{1}{|s_i|} \) and \( \frac{1}{Cg(s_i)} \) serve as a normalization in the gradient-like updates of the global counts (as there are more non-links than links, each non-link minibatch, representing a smaller fraction of the non-links, leads to more conservative updates). The “gradient steps” \( \rho_i^\phi \) are discussed below (Robbins-Monro condition).

Lastly, to be able to efficiently compute such quantities as \( N_{\omega \rightarrow ij}^\phi \) used for the computation of the link probability, one needs to store in memory, for each pair of nodes \((i, j)\), a \( K \times K \) matrix, which is not feasible for large networks. Thus, following [FBD13], we replace here \( N_{\omega \rightarrow ij}^\phi \) by \( \tilde{N}_{\omega \rightarrow ij}^\phi \), which amounts to assume that the contribution of each individual pair of nodes is negligible compared to all other pairs, a reasonable assumption when the network is large.
Table 1: Network datasets used in the experiments. Type A is for co-authorship, type C is for communication (e.g. email exchange), type H is for hyperlinks, type L is for lexical network and I for interaction network (e.g. money loan).

<table>
<thead>
<tr>
<th>Datasets</th>
<th>Nodes</th>
<th>Edges</th>
<th>Density</th>
<th>Directed</th>
<th>Diameter</th>
<th>Weights</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>astro-ph</td>
<td>16,706</td>
<td>121,251</td>
<td>0.87</td>
<td>False</td>
<td>14</td>
<td>1.8</td>
<td>3.3</td>
</tr>
<tr>
<td>hep-th</td>
<td>8,361</td>
<td>15,751</td>
<td>0.45</td>
<td>False</td>
<td>1</td>
<td>5.2</td>
<td>16</td>
</tr>
<tr>
<td>moreno_names</td>
<td>1,773</td>
<td>9,131</td>
<td>5.81</td>
<td>False</td>
<td>8</td>
<td>1.8</td>
<td>3.0</td>
</tr>
<tr>
<td>fb_uc</td>
<td>1,899</td>
<td>20,296</td>
<td>5.63</td>
<td>True</td>
<td>4</td>
<td>2.8</td>
<td>4.7</td>
</tr>
<tr>
<td>digg_reply</td>
<td>30,398</td>
<td>85,247</td>
<td>0.09</td>
<td>True</td>
<td>11</td>
<td>2.0</td>
<td>0.2</td>
</tr>
<tr>
<td>slashdot</td>
<td>51,083</td>
<td>130,370</td>
<td>0.05</td>
<td>True</td>
<td>11</td>
<td>2.1</td>
<td>0.3</td>
</tr>
<tr>
<td>enron</td>
<td>87,273</td>
<td>320,154</td>
<td>0.04</td>
<td>True</td>
<td>15</td>
<td>3.4</td>
<td>12.4</td>
</tr>
<tr>
<td>wiki-link</td>
<td>100,312</td>
<td>887,426</td>
<td>0.09</td>
<td>True</td>
<td>14</td>
<td>1.7</td>
<td>3.0</td>
</tr>
<tr>
<td>prosper-loans</td>
<td>89,269</td>
<td>3,330,225</td>
<td>0.42</td>
<td>True</td>
<td>2</td>
<td>2.0</td>
<td>0.2</td>
</tr>
</tbody>
</table>

4.2.1 Robbins-Monro condition

The convergence of stochastic variational inference is guaranteed under the Robbins-Monro condition [RM51] that imposes constraints on the gradient step, \( \sum \rho_t = \infty \) and \( \sum \rho_t^2 < \infty \) which can be obtained with \( \rho_t = \frac{1}{(\tau t + 1)^\kappa} \) with \( \kappa \in (0, 1] \). Thus, we maintain a gradient step for each of the global counts \( N^g \) and \( N^v \) accounting respectively for \( N^g \) and \( N^v \). For \( N^g \), we maintain individual gradient steps \( \rho_t^g \) for \( 1 \leq i \leq N \), following [MJG09]: this improved both convergence and prediction performance. Furthermore, to increase the speed of the inference, we update the global count \( N^g \) and \( N^v \) only after a minibatch round. For the global count \( N^g \), we update it after a burn-in period \( T_{\text{burnin}} \) such that \( T_{\text{burnin}} \leq |S| \). This heuristic provides a trade-off between updating the global statistics after each observation, which slows down the inference and may result in bad local optima, and updating them only after minibatches that are potentially large (proportional to the number of nodes).

5 Experimental validation

We evaluated the performance of the above models on several real world weighted networks, both directed and undirected. Their characteristics and properties are summarized in Table 1 and detailed descriptions are available in the online Koblenz network collection\(^2\). For both astro-ph and hep-ph datasets, we used the cleaned versions available in the graph-tool framework. In most large scale, real-life networks, the data is zero-inflated. However, predicting links (and weights) on mostly inactive regions of a network is usually not interesting, and researchers and practitioners have focused on predicting what’s happening in active regions of the network.

\(^2\)http://konect.uni-koblenz.de/networks/
Predicting links and predicting weights on links are two different tasks, and there is no guarantee that a model performing well on one task will perform well on the other. Even though we focus in this study on the weight prediction problem, we still want, for completeness reasons, to illustrate the behavior of the different models on the link prediction task.

## 5.1 Link prediction

In addition to the proposed mixed-membership models, we consider here two standard link prediction models, the stochastic block model, referred to as SBM, and its weighted extension, referred to as WSBM. We use the most recent version of these two models, namely the microcanonical stochastic block model implementation of [Pei18], which relies on an efficient MCMC inference method. In all models, the number of classes is set to $K = 10$ (as illustrated below, the choice of the number of latent classes, in between 10 and 50, does not have an important impact on these models).

As usual, the missing link prediction task is evaluated with the AUC-ROC score. For weighted models, we simply predict here a link through the probability that an edge exists between two unobserved nodes $(i, j)$ belonging to the test set, namely:

$$p(y_{ij} \geq 1|\hat{\Theta}, \hat{\Phi}) = 1 - \sum_{1 \leq k,k' \leq K} \hat{\theta}_{ik} \hat{\theta}_{jk'} e^{-\hat{\phi}_{kk'}}$$

Table 2 summarizes the results obtained with the aforementioned models when using 10% and 100% of the training data to fit the model. As one can note, when the complete training set is used (100%), SBM outperforms WSBM on 5 datasets and is overall the best performing model. This can be attributed to the fact that SBM directly aims at predicting links, unlike the weighted models, and does so via MCMC inference, which is known to yield accurate estimate when there is sufficient data. The mixed-membership family does not yield good results in this setting (100%) and is only interesting, in particular via WMMMSB-bg, on four datasets (astro-ph, digg_reply, enron, and prosper_loans). Within this family, WMMMSB-bg outperforms the other models on seven datasets (hep-th, moreno_names, fb_uc, digg_reply, slashdot, wiki-link and prosper_loans). For this reason, we will not use its simpler version, WMMSB, in the remainder of this study. Lastly, there is however an important degradation for SBM models when only 10% of the training set is used. Mixed-membership stochastic block models are more stable in this case (except on enron and wiki-link), indicating that the stochastic variational inference used in these models is appropriate with relatively few data.

## 5.2 Weight prediction

For this task, in addition to the previous models, we consider three other stochastic block models from [AJC14], among which two are weighted. These models are based on a generic variational inference scheme with several kernels: a Bernoulli kernel for the model referred to as SBM-ai, a Normal kernel for the model referred to as WSBM-ai-n, and a Poisson kernel for the model referred to as WSBM-ai-p. Lastly, we also consider the Edge Partition Model (EPM) proposed in [Zho15] (see Section 2), the inference of which relies on MCMC.

For both WMMMSB-bg and EPM, we used the inferred posterior distribution to estimate the missing weights by:

$$\hat{y}_{ij} | \hat{\Theta}, \hat{\Phi} = \sum_{1 \leq k,k' \leq K} \hat{\theta}_{ik} \hat{\theta}_{jk'} e^{-\hat{\phi}_{kk'}}$$

Since the stochastic block models have been primarily designed for solving the link prediction task, we do not have a posterior distribution adapted for weight prediction. Therefore, we used an estimation of the average weight value in each interaction based on the observed data. More precisely, let $N_k$ denote the number of nodes assigned to class $k$ by the model in the training set. The prediction of the weight on the link between two nodes
Table 3: Comparison of models in terms of MSE on different networks for $K = 10$. Results are averaged over 10 runs ± standard deviation. Best results are in bold.

<table>
<thead>
<tr>
<th>Model</th>
<th>MMSB</th>
<th>WMMSB-bg</th>
<th>EPM</th>
<th>SBM-ai</th>
<th>WSBM-ai-n</th>
<th>WSBM-ai-p</th>
<th>WSBM-SBM</th>
<th>WSBM-WSBM</th>
</tr>
</thead>
<tbody>
<tr>
<td>astro-ph</td>
<td>16.884 ± 0.81</td>
<td>20.182 ± 3.77</td>
<td>18.475 ± 0.71</td>
<td>18.448 ± 0.67</td>
<td>18.543 ± 0.55</td>
<td>18.127 ± 1.0</td>
<td>19.133 ± 0.89</td>
<td>18.298 ± 0.92</td>
</tr>
<tr>
<td>hep-th</td>
<td>111.218 ± 9.3</td>
<td>94.274 ± 8.23</td>
<td>121.334 ± 8.01</td>
<td>122.217 ± 14.85</td>
<td>122.316 ± 7.38</td>
<td>120.046 ± 12.53</td>
<td>121.755 ± 10.1</td>
<td>108.884 ± 14.77</td>
</tr>
<tr>
<td>moreno_names</td>
<td>5.334 ± 1.11</td>
<td>4.521 ± 0.89</td>
<td>5.077 ± 0.5</td>
<td>6.314 ± 1.77</td>
<td>6.355 ± 1.44</td>
<td>5.552 ± 0.9</td>
<td>4.616 ± 1.07</td>
<td>4.809 ± 1.48</td>
</tr>
<tr>
<td>fb_uc</td>
<td>15.240 ± 1.65</td>
<td>13.927 ± 1.06</td>
<td>12.966 ± 1.08</td>
<td>15.746 ± 1.56</td>
<td>18.298 ± 1.51</td>
<td>16.812 ± 1.2</td>
<td>15.456 ± 2.37</td>
<td>15.199 ± 2.39</td>
</tr>
<tr>
<td>digg-reply</td>
<td>1.528 ± 0.3</td>
<td>0.956 ± 0.04</td>
<td>2.022 ± 0.01</td>
<td>2.026 ± 0.0</td>
<td>2.028 ± 0.0</td>
<td>2.026 ± 0.0</td>
<td>2.022 ± 0.01</td>
<td>2.025 ± 0.01</td>
</tr>
<tr>
<td>slashdot</td>
<td>1.459 ± 0.19</td>
<td>0.897 ± 0.04</td>
<td>2.124 ± 0.01</td>
<td>2.164 ± 0.01</td>
<td>2.167 ± 0.0</td>
<td>2.173 ± 0.01</td>
<td>2.138 ± 0.0</td>
<td>2.144 ± 0.01</td>
</tr>
<tr>
<td>enron</td>
<td>58.099 ± 4.33</td>
<td>54.038 ± 3.6</td>
<td>61.337 ± 4.4</td>
<td>63.929 ± 4.17</td>
<td>66.878 ± 6.59</td>
<td>66.459 ± 6.6</td>
<td>58.384 ± 4.14</td>
<td>64.306 ± 5.28</td>
</tr>
<tr>
<td>wiki-link</td>
<td>5.846 ± 0.38</td>
<td>5.206 ± 0.16</td>
<td>5.633 ± 0.07</td>
<td>5.942 ± 0.09</td>
<td>5.874 ± 0.25</td>
<td>6.014 ± 0.14</td>
<td>5.993 ± 0.1</td>
<td>5.936 ± 0.2</td>
</tr>
<tr>
<td>prosper-loans</td>
<td>1.590 ± 0.43</td>
<td>1.217 ± 0.12</td>
<td>1.944 ± 0.0</td>
<td>2.043 ± 0.0</td>
<td>2.048 ± 0.0</td>
<td>2.043 ± 0.0</td>
<td>1.981 ± 0.0</td>
<td>1.979 ± 0.0</td>
</tr>
</tbody>
</table>

Figure 1: Impact of the number of classes on the performance of the different models, from $K = 10$ to $K = 50$, on astro.ph, enron and fb.uc.

$i$ and $j$ of the test set, respectively of class $k$ and $k'$, is given by:

$$\hat{y}_{ij} | i \in k, j \in k' = \sum_{(i', j') \in \mathcal{T}_k, i' \in k, j' \in k'} \frac{y_{i'j'}}{N_k N_{k'}},$$

where $i \in k$ means that node $i$ was assigned to class $k$ by the model and $\mathcal{T}_k$ denotes the training set.

The same method is used for the weighted stochastic block models WSBM, WSBM-ai, WSBM-ai-n and WSBM-ai-p as this method yielded even better results than the one directly making use of the expectation of $y_{ij}$ as in Eq. 5.

5.2.1 Inference time

Table 3 provides the mean squared error (MSE) scores for the different models and for all networks. For all models, the number of latent classes $K$ is set to 10. As one can note, the overall best performing model is WMMSB-bg, which yields the best results on 6 out of 9 datasets. MMSB, the other representative of the mixed-membership family, yields the best result on astro.ph while EPM is the best performing model on fb.uc and moreno_names, even though the difference with respect to WMMSB-bg on fb.uc is not large. Overall, the stochastic block models and their weighted versions do not perform well compared to WMMSB-bg (and to a lesser extent to EPM), which shows the importance of the soft class assignment mechanism present in mixed-membership models.

5.2.2 Impact of the number of classes

Figure 1 displays, for three datasets, the evolution of the MSE scores when the number of latent classes, $K$, varies from 10 to 50 (additional results are provided in Appendix B). As one can see, the results are relatively stable for all models. This is particularly true for the WMMSB-bg model. Note that the EPM model, due to its reliance on MCMC for inference, was not able to handle as many as 50 latent classes in a reasonable time (see below). This model also displays high variance when $K$ increases on the enron network, reflecting the fact that the variance of the weights in this network is by far the most important (see Table 1).

5.2.3 Convergence analysis

Lastly, Figure 2 shows the evolution of the log-likelihood for the WMMSB and WMMSB-bg models on the test set for the enron, slashdot and proper-loans datasets. The number of visited edges corresponds to the number of edges used to infer the model (remember that the inference is stopped, for all models, when the likelihood on the validation set no longer increases). We used three different sets of values for the hyperparameters $r$ and $p$ of WMMSB: \{(10, 0.1); (1, 0.5); (0.5, 0.66)\}. Regard-
Table 4: Comparison of models in terms of inference time, in hour, on different datasets for $K = 10$. Results are averaged over $10$ runs ± standard deviation.

<table>
<thead>
<tr>
<th>Model</th>
<th>MMSB</th>
<th>WMMSB-bg</th>
<th>EPM</th>
<th>SBM-ai</th>
<th>WSBM-ai-n</th>
<th>WSBM-ai-p</th>
<th>SBM</th>
<th>WSBM</th>
</tr>
</thead>
<tbody>
<tr>
<td>astro-ph</td>
<td>0.09 ± 0.02</td>
<td>0.07 ± 0.02</td>
<td>1.08 ± 0.01</td>
<td>0.08 ± 0.01</td>
<td>0.07 ± 0.01</td>
<td>0.08 ± 0.02</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
</tr>
<tr>
<td>hep-th</td>
<td>0.05 ± 0.01</td>
<td>0.08 ± 0.04</td>
<td>0.26 ± 0.01</td>
<td>0.02 ± 0.01</td>
<td>0.02 ± 0.01</td>
<td>0.02 ± 0.01</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
</tr>
<tr>
<td>moreno.names</td>
<td>0.02 ± 0.02</td>
<td>0.01 ± 0.01</td>
<td>0.02 ± 0.01</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
</tr>
<tr>
<td>fb-uc</td>
<td>0.01 ± 0.01</td>
<td>0.02 ± 0.01</td>
<td>0.02 ± 0.01</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
<td>0.01 ± 0.01</td>
</tr>
<tr>
<td>digg-reply</td>
<td>0.51 ± 0.30</td>
<td>0.71 ± 0.27</td>
<td>4.69 ± 0.04</td>
<td>0.15 ± 0.01</td>
<td>0.14 ± 0.01</td>
<td>0.15 ± 0.01</td>
<td>0.02 ± 0.01</td>
<td>0.06 ± 0.01</td>
</tr>
<tr>
<td>slashdot</td>
<td>1.27 ± 0.24</td>
<td>1.51 ± 0.48</td>
<td>12.98 ± 1.26</td>
<td>0.39 ± 0.03</td>
<td>0.36 ± 0.03</td>
<td>0.39 ± 0.04</td>
<td>0.04 ± 0.01</td>
<td>0.08 ± 0.01</td>
</tr>
<tr>
<td>enron</td>
<td>1.52 ± 0.62</td>
<td>1.29 ± 0.61</td>
<td>25.03 ± 0.03</td>
<td>1.13 ± 0.08</td>
<td>1.04 ± 0.05</td>
<td>1.00 ± 0.01</td>
<td>0.10 ± 0.01</td>
<td>0.22 ± 0.01</td>
</tr>
<tr>
<td>wiki-link</td>
<td>2.15 ± 0.50</td>
<td>1.67 ± 0.34</td>
<td>25.10 ± 0.03</td>
<td>1.38 ± 0.16</td>
<td>1.48 ± 0.08</td>
<td>1.58 ± 0.33</td>
<td>0.29 ± 0.01</td>
<td>0.63 ± 0.05</td>
</tr>
<tr>
<td>prosper-loans</td>
<td>0.87 ± 0.24</td>
<td>1.71 ± 0.78</td>
<td>25.13 ± 0.03</td>
<td>1.64 ± 0.09</td>
<td>1.62 ± 0.10</td>
<td>1.64 ± 0.20</td>
<td>1.38 ± 0.09</td>
<td>3.04 ± 0.22</td>
</tr>
</tbody>
</table>

Figure 2: Log-likelihood convergence for WMMSB and WMMSB-bg models on a test set containing 20% of the edges of the networks. Three different sets of hyperparameters are used for WMMSB.

less of the values of these hyperparameters, one can observe that the augmented model WMMSB-bg converges to a better solution, in terms of likelihood, than the other models on enron and slashdot. The curve on slashdot further shows that the WMMSB models were stopped earlier than WMMSB-bg, illustrating the fact that this latter model, due to its additional prior assumptions, seems to be less prone to overfitting.

6 Conclusion

We studied in this paper the problem of modeling weighted networks through generalized stochastic block models. The stochastic block models proposed so far for weighted networks suffer from the same drawback as standard stochastic block models, namely the fact that a node can belong to only one class, which is not realistic for many networks and can be corrected by using mixed-membership block models. We have thus developed new mixed-membership stochastic block models to model (directed or undirected) weighted networks and have proposed a scalable inference method, based on a combination of collapsed and stochastic variational inference. This allowed us to deploy the new models on large networks comprising millions of edges. Experiments conducted on nine real-world networks of different types and sizes showed that the new models outperform previously proposed models on the weight prediction task, with reasonable inference time.

In the future, we want to develop versions of these models with different kernels so as to model signed networks and be able to generate different types of weights, thus extending the set of tools available for network analysis.

Acknowledgements

This work was partially supported by the Région Rhône-Alpes and the ANR project ANR-19-PIA-0003 (MIAI@Grenoble Alpes).

References


A Derivation of the collapsed variational updates for WMMSB-bg

In the WMMSB-bg model, the collapsed variational distribution takes the form:

\[ q(\Pi) = q(\Theta, \Phi | Z, R, P) q(Z) q(R) q(P) \]

The variational distribution for \( r_{kk'} \) is taken in the Gamma family: \( q(r_{kk'}) = \text{Gamma}(a_{kk'}, b_{kk'}) \) for \( 1 \leq k, k' \leq K \). The collapsed ELBO can thus be rewritten as:

\[ \log p(Y) \geq \mathcal{L}_{Z, R, P}, \]

with:

\[ \mathcal{L}_{Z, R, P} = \mathbb{E}_q[\log p(Y, Z, R, P|\Omega)] + H[q(Z)] + H[q(R)] + H[q(P)] \]

\[ = \mathbb{E}_q[\log p(Y, Z)] + H[q(Z)] + \mathbb{E}_q[\log p(R|Y, Z, P)] + H[q(R)] + \mathbb{E}_q[\log p(P|Y, Z)] + H[q(P)]. \]

Optimizing \( \gamma_{ijkk'} \) In the Beta-Gamma augmentation, the parameters \( p \) and \( r \) are marginalized in the update given by:

\[ \gamma_{ijkk'} \propto e^{E_q[1 - \gamma_{ij}] \log \mathbb{E}_q(r_{kk'}) | q(p_{kk'} | P(z_{i-j} = k, z_{i-j} = k', Y^{-ij}, Z^{-ij}, \Omega)]} \]

By using a first order Taylor expansion, one obtains:

\[ \gamma_{ijkk'} \propto (N_{kk'}^{\Theta - i} + \alpha_k)(N_{kk'}^{\Theta - j} + \alpha_k) \]

\[ \mathbb{N} \left( y_{ij}; \mathbb{N}_{kk'}^{Y - ij} + \mathbb{E}_q[r_{kk'}], p' \right), \]

with:

\[ p' = \frac{\mathbb{E}_q[p_{kk'}]}{\mathbb{E}_q[p_{kk'}] N_{kk'}^{\Phi - ij} + 1}. \]

Optimizing \( p_{kk'} \) In order to maximize the collapsed ELBO w.r.t. \( p_{kk'} \), one can let \( q(p_{kk'}) = p(p_{kk'} | Y, Z) = E_q(r_{kk'}) | p[p_{kk'} | Y^{(kk')}, Z^{(kk')}, r_{kk'}] \). As the negative binomial and Beta distributions are conjugate, a closed-form expression can be obtained:

\[ p(p_{kk'} | Y^{(kk')}, Z^{(kk')}, r_{kk'}) \propto p(Y^{(kk')} | Z^{(kk')}, r_{kk'}) p(r_{kk'}) \]

\[ \propto (1 - p_{kk'})^{r_{kk'}} N_{kk'}^{Y_{kk'}} p_{kk'}^{N_{kk'}^{Y_{kk'}}} (1 - p_{kk'})^{c(1 - c) - 1} \]

\[ \propto p_{kk'}^{c + N_{kk'}^{Y_{kk'}} - 1} (1 - p_{kk'})^{c + N_{kk'}^{Y_{kk'}} - 1} \]

\[ = \text{Beta}(c + N_{kk'}^{Y_{kk'}}, c(1 - c) + N_{kk'}^{Y_{kk'}} r_{kk'}). \]

Finally, by resorting again to a first order Taylor expansion, one obtains:

\[ p_{kk'} \sim \text{Beta}(c + N_{kk'}^{Y_{kk'}}, c(1 - c) + N_{kk'}^{Y_{kk'}} E_q[r_{kk'}]), \]

so that:

\[ \mathbb{E}_q[p_{kk'}] = \frac{c + N_{kk'}^{Y_{kk'}}}{c + N_{kk'}^{Y_{kk'}} + c(1 - c) + N_{kk'}^{Y_{kk'}} E_q[r_{kk'}]}. \]

Optimizing \( r_{kk'} \) To isolate the contribution, in the collapsed ELBO, that depends on \( r_{kk'} \) (through \( a_{kk'} \) and \( b_{kk'} \)), we only consider the links that have been generated within the classes \( k, k' \), denoted by \( Y^{(kk')} \). As \( y_{ij} \sim \mathbb{N}(r_{kk'}, p_{kk'}) \) if \( i \) is in class \( k \) and \( j \) in class \( k' \), one has:

\[ \mathcal{L}_{r_{kk'}} = \mathbb{E}_q(r_{kk'}) \left[ \log p(r_{kk'} | Y^{(kk')}, Z^{(kk')}, p_{kk'}) \right] + H[q(r_{kk'})]. \]

By applying Bayes rules and dropping the normalizing term that does not depend on \( r_{kk'} \), one gets:

\[ \mathcal{L}_{r_{kk'}} = \mathbb{E}_q(r_{kk'}) \left[ \log \left( \prod_{i,j \in Y^{(kk')}} \left( r_{kk'} + y_{ij} - 1 \right) p_{kk'} \right) \right] + H[q(r_{kk'})]. \]

If \( y_{ij} = 0 \), then \( \Gamma(r_{kk'} + y_{ij}) = \Gamma(r_{kk'} + y_{ij} + 1) \), whereas if \( y_{ij} \neq 0 \), then \( \Gamma(r_{kk'} + y_{ij}) = \frac{1}{y_{ij} r_{kk'} y_{ij}} \). Furthermore, in this latter case:

\[ B(r_{kk'}, y_{ij}) = \int_0^1 t^{r_{kk'} - 1} (1 - t)^{y_{ij} - 1} dt \leq \frac{1}{r_{kk'}}, \]

so that:

\[ \log \prod_{i,j \in Y^{(kk')}} \Gamma(r_{kk'} + y_{ij}) \left( r_{kk'} + y_{ij} + 1 \right) \geq N_{kk'} Y_{kk'} \log(r_{kk'}) + \text{cst}, \]

with \( N_{kk'} Y_{kk'} = \sum_{i,j \in Y^{(kk')}} y_{ij} \).

Furthermore, from the model definitions, one has:

\[ \log p(r_{kk'}) = (r_{kk'} - 1) \log(r_{kk'}) - r_{kk'} c_0 + \text{cst} \]
\[ H[q(r_{kk'})] = a_{kk'} + \log(b_{kk'}) + \log(\Gamma(a_{kk'})) + (1 - a_{kk'})\Psi(a_{kk'}). \]

Hence:

\[
\mathcal{L}_{[r_{kk'}]} \geq N_{kk'}^{\Phi} a_{kk'} b_{kk'} \log(1 - p_{kk'})
+ (r_0 c_0 - 1)(\Psi(a_{kk'}) + \log(b_{kk'}))
- c_0 a_{kk'} b_{kk'} + N_{kk'}^{Y} (\Psi(a_{kk'})
+ \log(b_{kk'})) a_{kk'} + \log(b_{kk'})
+ \log(\Gamma(a_{kk'})) + (1 - a_{kk'})\Psi(a_{kk'}). \]

Maximizing the right-hand term of the above inequality with respect to \(b_{kk'}\) yields:

\[ b_{kk'} = \frac{r_0 c_0 + N_{kk'}^{Y}}{a_{kk'}(c_0 - N_{kk'}^{\Phi} \log(1 - p_{kk'}))}. \]

As \(r_{kk'} \sim \text{Gamma}(a_{kk'}, b_{kk'})\), one finally obtains:

\[ \mathbb{E}_q[r_{kk'}] = a_{kk'} b_{kk'} = \frac{r_0 c_0 + N_{kk'}^{Y}}{c_0 - N_{kk'}^{\Phi} \log(1 - p_{kk'})}. \]

### B Impact of the number of classes

Figure 3 displays, for nine datasets, the evolution of the MSE scores when the number of latent classes, \(K\), varies from 10 to 50. These results confirm the ones reported in Figure 1, Section 5, on the stability of the different models, for weight prediction, with respect to the number of latent classes.

Figure 3: Performance sensibility when the number of latent classes vary from \(K = 10\) to \(K = 50\).