1 Bayesian computational algorithms for social network analysis
Alberto Caimo¹, Isabella Gollini²

1.1 Introduction

Interest in statistical network analysis has grown massively in recent decades and its perspective and methods are now widely used in many scientific areas which involve the study of various types of networks for representing structure in many complex relational systems such as social relationships, information flows, protein interactions, etc.

Social network analysis is based on the study of social relations between actors so as to understand the formation of social structures by the analysis of basic local relations. Statistical models have started to play an increasingly important role because they give the possibility to explain the complexity of social behaviour and to investigate issues on how the global features of an observed network may be related to local network structures. The observed network is assumed to be generated by local social processes which depend on the self-organising dyadic relations between actors. The crucial challenge for statistical models in social network theory is to capture and describe the dependency giving rise to network global topology allowing inference about whether certain local structures are more common than expected.

Unfortunately the computational burden required to estimate these models is the main barrier to estimation. Recent theoretical developments and advances in approximate procedures have given the possibility to make important progress to overcome statistical inference problems.

¹) Social Network Analysis Research Center,
Università della Svizzera italiana, Switzerland.
alberto.caimo@usi.ch

²) Department of Civil Engineering,
University of Bristol, United Kingdom.
isabella.gollini@bristol.ac.uk
In this chapter we review some of the most recent computational advances in the rapidly expanding field of statistical social network analysis (see [26] for a recent review) using the R open-source software.

In particular we will focus on Bayesian estimation for two important families of models: exponential random graph models (ERGMs) and latent space models (LSMs) and we will provide the R code used to produce the results obtained in this chapter.

The chapter is organised as follows: in Section 1.2, we introduce the basic notation for social network analysis. In Section 1.3, we highlight the basic statistical work on social networks citing recent references to enable interested readers to learn more. In particular, our interest lies on describing exponential random graph models and latent space models. In Section 1.4, we discuss Bayesian analysis for these two families of models and computational methods on a well-known dataset using the R software. Predictive goodness-of-fit diagnostics are also described at the end of the section. We conclude in Section 1.6 with a discussion of some future challenges.

1.2 Social networks as random graphs

Networks are relational data that can be defined as a collection of nodes interacting with each other and connected in a pairwise fashion. In typical applications, the nodes represent a set actors of various kind (people, organisations, countries, etc.) and the set edges represent a specific relationship between them (friendship, collaboration, etc.).

From a statistical point of view, networks are relational data represented as mathematical graphs. A graph consists of a set of $n$ nodes and a set of $m$ edges which define some sort of relationships between pair of nodes called dyads.

The connectivity pattern of a graph can be described by an $n \times n$ adjacency matrix $y$ encoding the presence or absence of an edge between node $i$ and $j$:

$$y_{ij} = \begin{cases} 1, & \text{if } (i, j) \text{ are connected,} \\ 0, & \text{otherwise.} \end{cases}$$

Two nodes are adjacent or neighbours if there is an edge between them. If $y_{ij} = y_{ji}, \forall i, j$ then the adjacency matrix is symmetric and the graph is undirected, otherwise the graph is directed and it is often called digraph. Edges
connecting a node to itself (self-loops) are generally not allowed in many applications and will not be considered in this context. The nature of the edges between nodes can take a range of values indicating the strength, frequency, intensity, etc. of the relation between a dyad. In this paper we consider only binary networks. According to the generally used notation, $y$ will be used to indicate both a random graph and its adjacency matrix.

1.3
Statistical modelling approaches to social network analysis

Many probability models have been proposed in order to summarise the connectivity structure of social networks by utilising their network statistics. The family of exponential random graph models (ERGMs) is a generalisation of various models which take different assumptions into account: the Bernoulli random graph model [5] in which edges are considered independent of each other; the $p_1$ model [12] where dyads are assumed independent, and its random effects variant the $p_2$ model [30]; and the Markov random graph model [6] where each pair of edges is conditionally dependent given the rest of the graph. The family of latent space models has been proposed by [11] under the assumption that each node of the graph has a unknown position in a latent space and the probability of the edges are functions of those positions and node covariates. The latent position cluster model of [10] represents a further extension of this approach that takes account of clustering. Other latent variable modelling approaches are represented by stochastic blockmodels [22] that involve block model structures whereby network nodes are partitioned into latent classes and the presence of any relationship between them depends on their block membership.

1.3.1
Exponential random graph models (ERGMs)

Introduced by [12] to model individual heterogeneity of nodes and reciprocity of their edges, the family of exponential random graph models (ERGMs) was generalised by [6], [31] and [29]. ERGMs constitute a broad class of network models (see [25] for an introduction) that assume that the topological structure of an observed network $y$ can be explained in terms of the relative prevalence of a set of overlapping subgraph configurations $s(y)$ called network
This equation states that the probability that an observed network \( y \) given the set of parameters \( \theta \) is equal to the exponent of an observed vector of network statistics \( s(y) \) multiplied by its associated vector of unknown parameters \( \theta \) divided by a normalising constant \( z(\theta) \) to make all probabilities sum to one. The latter is calculated as the sum over all possible network configurations on the same set of \( n \) nodes of the observed network. In practice \( z(\theta) \) is computationally infeasible to calculate for non trivially-small networks.

1.3.2 Latent space models (LSMs)

Latent space models were introduced by [11] under the basic assumption that each node has an unknown position \( z_i \) in a \( d \)-dimensional Euclidean latent space. Network edges are assumed to be conditionally independent given the latent positions, and the probability of an edge between nodes is modelled as a function of their positions. Generally, in these models the smaller the distance between two nodes in the latent space, the greater their probability of being connected. The likelihood function of latent space models can be written as follows:

\[
p(y|z, \alpha) = \prod_{i \neq j} \frac{\exp(\alpha - d_{ij}y_{ij})}{1 + \exp(\alpha - d_{ij})}
\]

The standard metric is the Euclidean distance (ED in Table 1.3) and is defined as: \( d_{ij} = |z_i - z_j| \). As an alternative the squared Euclidean distance (SED in Table 1.3) is defined as: \( d_{ij} = |z_i - z_j|^2 \) and has been proposed by [8] for computational reasons (see 1.5.2). The latent positions are assumed to be Normally distributed, or having a Gaussian mixture model structure in case of the latent position cluster models (LPCMs), a generalisation of latent space models where latent clusters are assumed to be useful to explain data heterogeneity. For strongly asymmetric graph, it is suggested to use the bilinear latent model setting \( d_{ij} = z_i'z_j \) so that the probability of a link depends on the angle between two actors. This model is available in the latentnet package through the \texttt{bilinear} argument included in the \texttt{ergmm} function. All the
presented latent space network models can be extended to incorporate covariate informations $x_{ij}$ introducing a parameter $\beta$, or the degree heterogeneity in sending or receiving links, these parameters are called sender and receiver if the network is directed, or sociality if the network is undirected [18].

1.4 Bayesian inference for social network models

The Bayesian approach to statistical problems is probabilistic. Inference is based on the posterior distribution which is the conditional probability of the unknown quantities $\Omega$ given the data $y$. The posterior distribution extracts the information in the data and provide a complete summary of the uncertainty about the unknowns via Bayes’ theorem:

$$p(\Omega|y) = \frac{p(y|\Omega) p(\Omega)}{p(y)} \quad (1.2)$$

Bayesian analysis is able to give us a full probabilistic framework of uncertainty and this is something which is essential in the context of complex statistical modelling. Moreover recent research in social network analysis has demonstrated the advantages and effectiveness of probabilistic Bayesian approaches to relational data. In this chapter we will focus on parameter inference so the uncertainties $\Omega$ will refer to the ERGM parameters $\theta$ or the LSM parameters $\alpha$ and latent positions $z$.

1.4.1 R-based software tools

Applied researchers interested in Bayesian statistics are increasingly attracted to R [23] because of the ease of which one can code algorithms to sample from posterior distributions as well as the significant number of packages contributed to the Comprehensive R Archive Network (CRAN) that provide tools for Bayesian inference.

R represents a useful tool for social network analysis with many advantages over traditional software packages. With a little coding and patience, one can produce ad hoc analyses and visualisations for the problem under study. Moreover R has a huge set of statistical libraries so that end users can complement their social network analysis research with any analysis of your choosing within R environment.
In this section we briefly review Bayesian tools for ERGMs and LSMs:

- The *Bergm* package (version 3.0.1) [3] implements Bayesian analysis for ERGMs using the methods proposed by [1,2,4]. The package provides a comprehensive framework for Bayesian inference and model selection using Markov chain Monte Carlo (MCMC) algorithms.

- The *latentnet* package (version 2.5.1) [16, 17], which is part of the *statnet* suite of packages [9], provides comprehensive toolsets for Bayesian analysis for latent position and cluster network models using MCMC procedures.

- The *VBLPCM* package (version 2.4.3) [27] contains a collection of functions implementing variational Bayesian Inference for the latent position cluster model.

- The *lvm4net* package (version 0.2) [7] contains a collection of functions implementing fast variational Bayesian inference for latent space models.

Other *R* implementations of Bayesian methods for statistical social network models include: *RSiena* [24] implementing stochastic actor-based models; *hergm* [28] implementing hierarchical ERGMs with local dependence; *sna* (belonging to the *statnet* suite of packages) generating posterior samples from Butt’s Bayesian network accuracy model using Gibbs sampling.

1.5

Data

We demonstrate ideas and examples throughout the paper using the Dolphin network dataset, an undirected social network of frequent associations between 62 dolphins in a community living off Doubtful Sound, New Zealand (see Figure 1.1), as compiled by [19]. The results presented in this paper have been obtained using *R* version 3.1.3. To create, manipulate and visualise the observed network data \( y \) we can use the function `network` and `plot` included in the *statnet* suite of packages.

```r
y <- read.table("http://moreno.ss.uci.edu/dolphins.dat", skip = 130)
```
Bayesian inference for exponential random graph models

Bayesian inference for ERGMs is based on the posterior distribution of $\theta$ given the data $y$:

$$p(\theta|y) = \frac{p(y|\theta) p(\theta)}{p(y)} = \frac{\exp\{\theta^t s(y)\} p(\theta)}{z(\theta) p(y)}$$

(1.3)

where $p(y)$ is the evidence or marginal likelihood of $y$ which is typically intractable.

Standard MCMC methods such as the Metropolis-Hastings algorithm, can deal with posterior estimation as long as the target posterior density is known up to the model evidence $p(y)$. Unfortunately in the ERGM context the posterior density $p(\theta|y)$ of non-trivially small ERGMs includes two intractable
normalising constants, the model evidence \( p(y) \) and \( z(\theta) \). For this reason, the ERGM posterior density is “doubly intractable” [21].

In order to carry out Bayesian inference for ERGMs, the Bergm package makes use of a combination of Bayesian algorithms and MCMC techniques [1,2]. The approximate exchange algorithm circumvents the problem of computing the normalising constants of the ERGM likelihoods, while the use of multiple chains and efficient adaptive proposal strategies are able to speed up the computations and improve chain mixing quite significantly.

The approximate exchange algorithm implemented by the bergm function can be summarised in the following way:

For each chain, repeat until converge:

1) generate \( \theta' \) (using some proposal strategy)

2) simulate \( s(y') \) from ERGM likelihood (using standard MCMC procedures such as [20])

3) update \( \theta \rightarrow \theta' \) with the log of the probability:

\[
\min \left( 0, \ [\theta - \theta']^t [s(y') - s(y)] + \log \left[ \frac{p(\theta')}{p(\theta)} \right] \right)
\]

Let us consider the following three dimensional model including the number of edges and two new specification statistics e.g.: geometrically weighted edgewise shared partners (gwesp) and geometrically weighted non-edgewise shared partners (gwnsp) [14]:

\[
\begin{align*}
gwesp &= e^{\phi_u} \sum_{k=1}^{n-2} \left\{ 1 - (1 - e^{-\phi_u})^k \right\} EP_k(y) \\
gwnsp &= e^{\phi_v} \sum_{k=1}^{n-2} \left\{ 1 - (1 - e^{-\phi_v})^k \right\} NEP_k(y)
\end{align*}
\]

where the scale parameters \( \phi_v = \phi_u = 0.6 \), and \( EP_k(y) \) and \( NEP_k(y) \) are respectively the number of connected and non-connected pairs of nodes with exactly \( k \) common neighbours.
We can use the `bergm` function to sample from the posterior distribution using the MCMC algorithm described above. In this example we use the parallel adaptive direction sampling (ADS) procedure described in [1] for step 1 and 1,200 iterations (`main.iters`) for each chain. We set the number of MCMC chains to 9 by using the argument `nchains`. The number of iterations used to simulate network statistics $s(y')$ at step 2 is defined by the argument `aux.iters` and it is set to 3,000.

```r
model <- y ~ edges +
    gwnsp(.6, fixed = TRUE) +
    gwesp(.6, fixed = TRUE)

post <- bergm(model,
              main.iters = 1200,
              aux.iters = 3000,
              nchains = 9)

bergm.output(post, lag = 200)
```

The `bergm.output` function produces MCMC diagnostic plots (Figure 1.2) and the estimated posterior means, standard deviations, and acceptance rates for each of the 9 chains and for the aggregated overall chain.

MCMC results for Model:
\[
y \sim \text{edges} + \text{gwnsp}(0.6, \text{fixed} = \text{TRUE}) + \text{gwesp}(0.6, \text{fixed} = \text{TRUE})
\]

**Posterior mean:**

|          | theta1  | theta2   | theta3   |
|----------|---------|----------|----------|
| Chain 1  | -2.3512 | -0.1864  | 0.7521   |
| Chain 2  | -2.3889 | -0.1800  | 0.7516   |
| Chain 3  | -2.3363 | -0.1779  | 0.7069   |
| Chain 4  | -2.5628 | -0.1646  | 0.7892   |
| Chain 5  | -2.3709 | -0.1799  | 0.7316   |
| Chain 6  | -2.5407 | -0.1646  | 0.7799   |
| Chain 7  | -2.4301 | -0.1783  | 0.7698   |
| Chain 8  | -2.4524 | -0.1799  | 0.8017   |
| Chain 9  | -2.3682 | -0.1789  | 0.7424   |

**Posterior sd:**

|          | theta1  | theta2   | theta3   |
|----------|---------|----------|----------|
| Chain 1  | 0.3324  | 0.0395   | 0.1130   |
| Chain 2  | 0.4319  | 0.0466   | 0.1158   |
| Chain 3  | 0.3734  | 0.0408   | 0.1062   |
| Chain 4  | 0.4111  | 0.0496   | 0.1178   |
| Chain 5  | 0.4886  | 0.0537   | 0.1490   |
| Chain 6  | 0.3679  | 0.0405   | 0.1349   |
In this example, we can observe a low density effect expressed by the negative value of the posterior mean of the edge effect parameter ($\theta_1$) combined with the negative value of multiple connectivity ($\theta_2$) and positive value of transitivity parameter ($\theta_3$).

### 1.5.2 Bayesian inference for latent space models

A fully Bayesian approach for latent space models allows the estimation of all the parameters and latent positions simultaneously e.g. via MCMC sampling or variational approximation.

In this paragraph, we perform an empirical Bayesian analysis in order to compare different computational approaches for the visualisation and prediction properties of LSMs with and without clustering. To carry out this type of analysis we can use the following R packages: latentnet [16], VBLPCM [27] and lvm4net [7]. Their main features of these packages are shown in Table 1.3.

The latentnet package uses Bayesian MCMC algorithms so the model estimation is computationally very expensive, and the times to estimate the model can become extremely large even for networks of the order of hundreds of nodes. For this reason the variational Bayes approach to estimate the latent space model and the latent position cluster model in order to make fea-
MCMC output for Model: \( y \sim \text{edges} + \text{gwnsp}(\tau, \text{fixed} = \text{TRUE}) + \text{gwesp}(\tau, \text{fixed} = \text{TRUE}) \)

Tab. 1.2 MCMC diagnostics for the overall chain. The 3 plot columns are: estimated marginal posterior densities (left), traces (center) and autocorrelation plots (right).

Tab. 1.3 Comparison of the main features of the packages for latent space modeling

| Package       | Model ED | Model SED | Inference Method MCMC | Inference Method Variational | Clustering |
|---------------|----------|-----------|------------------------|-------------------------------|------------|
| latentnet     | ✓        | x         | ✓                      | x                             | ✓          |
| VBLPCM        | ✓        | x         | ✓                      | ✓                             | ✓          |
| lvm4net       | x        | ✓         | ✓                      | ✓                             | x          |

possible the modelling of larger networks \([8, 27]\). The basic idea of this method
is to find a lower bound of the log-likelihood by introducing a variational posterior distribution $q$ and maximize it [15]. The posterior probability of the unknown parameters $(z, \alpha)$ can be written in the following form:

$$p(z, \alpha | y) = Cp(y | z, \alpha) p(\alpha) \prod_{i=1}^{N} p(z_i),$$

where $C$ is the unknown normalising constant. In the VBLPCM package, a hierarchical prior structure is taken into consideration.

In [8], the variational posterior $q(z, \alpha | y)$ is defined in the following way:

$$q(z, \alpha | y) = q(\alpha) \prod_{i=1}^{N} q(z_i),$$

where $q(\alpha) = \mathcal{N}(\tilde{\xi}, \tilde{\psi}^2)$ and $q(z_i) = \mathcal{N}(\tilde{z}_i, \tilde{\Sigma})$.

The idea of using the squared Euclidean distance in the LSM was proposed by [8] in order to have less approximation to be made in the variational estimation procedure.

In the latentnet package, we use the function called `ergmm` to estimate the posterior distribution of the LSM parameters and latent positions. The argument $d$ refers to the dimension of the latent space, which we set equal to 2 to make the visualisation of the latent positions of the nodes easier.

```r
post.latentnet <- ergmm(y ~ euclidean(d = 2))
```

In the VBLPCM package, we can use the function called `vblpcmfit` to estimate the posterior distribution of the LPCM by specifying the number of clusters. In order to estimate a LSM we consider one cluster by setting the argument $G$ equal to 1.

```r
post.vblpcm <- vblpcmfit(vblpcmstart(y, G = 1, d = 2))
```

It is important to notice that the variational maximisation algorithm is subject of the risk of reaching local maximum.

The package VBLPCM provides a special function called `vblpcmstart` to generate initial latent positions. This algorithm is based on the Fruchterman-Reingold method by default (argument `START`), but there is also the possibility of using random values, geodesics distances or Graph Laplacian methods. In this function other model features such as sociality effects, and node covariates can also be specified.
In lvm4net we use the function called lsm to estimate the posterior distribution of the LSM parameters and latent positions using a variational inferential approach. This function makes use of the Fruchterman-Reingold method to set the initial positions by default. Multi-start procedure can be implemented by changing the value associated to the argument nstart and only the values reaching the maximum are stored. It is also possible to start from random initial positions by setting the argument randomZ equal to TRUE.

From Table 1.4, we can see that the lsm function is much faster than the ergmm function. In this case, the squared Euclidean distance is used.

```
post.lvm4net <- lsm(y[,,], D = 2)
```

|                  | Time in sec. |
|------------------|--------------|
| latentnet        | 111.20       |
| VBLPCM           | 14.02        |
| lvm4net          | 6.47         |

The latent positions are invariant under rotation, reflection and translations. For this reason we can match the rotations using the rotXtoY function (included in lvm4net) in order to visualise and compare the latent positions estimated by the three approaches using the plotY function (included in lvm4net).

```
Z <- post.lvm4net$lsmEZ
Zm <- rotXtoY(post.latentnet$mkl$Z,Z)$X
Zv <- rotXtoY(post.vblpcm$V_z,Z)$X
plotY(y[,,], EZ = Zm, main = "latentnet")
plotY(y[,,], EZ = Zv, main = "VBLPCM")
plotY(y[,,], EZ = Z, main = "lvm4net")
```

In Figures 1.5, 1.6, 1.7 we can see the estimated latent positions obtained using the three packages. In this example, the visualisation results obtained from latentnet and lvm4net are similar even though the distance model adopted is different.

Latent position cluster models (LPCMs) are latent space models which incorporate a Gaussian mixture model structure for the latent positions of nodes
in the latent space in order to accommodate the clustering of nodes in the network. The `latentnet` and `VBLPCM` packages can be used to estimate latent position cluster models by fixing the number of clusters (G). For our toy example, we choose 2 clusters.

```r
post.latentnet.G2 <- ergmm(y ~ euclidean(d = 2, G = 2))
post.vblpcm.G2 <- vblpcmfit(vblpcmstart(y, G = 2, d = 2))
```

In Figures 1.8 and 1.9 we can see the latent positions and the clusters returned by the two packages. The two algorithms give very similar results as they find latent groups differing of just one node.
Tab. 1.6 Latent positions obtained by using the VBLPCM package.

Tab. 1.10 Timings in seconds to fit LPCM with two clusters (G = 2).

| Package      | Time in sec. |
|--------------|--------------|
| latentnet    | 86.27        |
| VBLPCM       | 11.95        |

From Table 1.10 it is possible to notice that the VBLPCM package is much faster than the latentnet package. The latentnet package gives exact estimates as they are based on MCMC simulations from the posterior distribution. However it only allows to deal with small networks whereas the approximate approaches of the VBLPCM and lvm4net packages are able to handle networks on thousands of nodes.
Tab. 1.7 Latent positions obtained by using the lvm4net package.

1.5.3 Predictive goodness-of-fit (GoF) diagnostics

An important feature of the Bayesian approach is to make available procedures to establish whether the estimated parameter posterior of the model achieves a good fit to the key topological features of the observed network.

The function included in the Bergm package provides a useful tool for assessing Bayesian goodness-of-fit so as to examine the fit of the data to the posterior model obtained by the bergm function. The observed network data are compared with a set of networks simulated from independent parameter
values of the posterior density estimate. This comparison is made in terms of high-level network statistics not explicitly included in the model [13].

The R code below is used to compare some high level network statistics observed in the Dolphin network with a series of network statistics simulated from 100 random realisations of the estimated posterior distribution `post.est` using 10,000 auxiliary iterations for the network simulation step. The `bgof` function included in the `Bergm` package returns the plots shown in Figure 1.11.

```r
bgof(post,
     n.deg = 20,
     n.dist = 15,
     n.esp = 15)
```

In Figure 1.11 we see, based on the various GoF statistics, that the networks simulated from the posterior distribution are in reasonable agreement with
the observed network. We can therefore conclude that the model is a reasonable fit to the data, despite its simplicity.

In the LSM context, it is possible to use the \texttt{gof} function included in \texttt{latentnet} and \texttt{VBLPCM} and the \texttt{goflsm} function included in the \texttt{lvm4net} package to perform posterior GoF diagnostics. The \texttt{GOF} argument can be used to set the types of GoF statistics we want to analyse. Figures 1.12, 1.13, and 1.14 display the GoF plots.

\begin{verbatim}
gf.latentnet <- gof(post.latentnet, 
    GOF = ~ degree + esp + distance)
plot(gf.latentnet)

gf.vblpcm <- gof(post.vblpcm, 
    GOF = ~ degree + esp + distance)
plot(gf.vblpcm)
\end{verbatim}

\textbf{Tab. 1.9} Estimated latent positions from LPCM with 2 clusters obtained by using the \texttt{VBLPCM} package.
Tab. 1.11 GoF diagnostics for ERGM (Bergm package): The red line displays the goodness of fit statistics for the observed data together with boxplots of GoF network statistics based on 100 simulated networks from the posterior distribution.

\[
gf.lvm4net <- goflsm(post.lvm4net, 
Y = y[,], 
stats = c("degree", "esp", "distance"), 
doplot = FALSE)
\]

plot(gf.lvm4net)

The GoF analysis indicates that the LSM estimated by using the variational approximation with squared Euclidean distance implemented in the lvm4net package displays a better fit of the model to the data compared to the other two approaches.

To display the GoF diagnostics for the LPCMs estimated above, we can use the same R functions.

\[
gf.latentnet.G2 <- gof(post.latentnet.G2, 
GOF = ~ degree + esp + distance)
\]

plot(gf.latentnet.G2)

\[
gf.vblpcm.G2 <- gof(post.vblpcm.G2, 
GOF = ~ degree + esp + distance)
\]
From Figures 1.15 and 1.16 we can see that the VBLPCM package has a better fit to the data in terms of edgewise shared partners distributions compared to the latentnet package. For this example, the inclusion of 2 clusters does not
Tab. 1.14 GoF diagnostics for LSM (lvm4net package): The red line displays the goodness of fit statistics for the observed data together with boxplots of GoF network statistics based on 100 simulated networks from the posterior distribution.

Tab. 1.15 GoF diagnostics for LPCM with 2 clusters (latentnet package): The solid black line displays the goodness of fit statistics for the observed data together with boxplots of GoF network statistics based on 100 simulated networks from the posterior distribution.

seem to produce a significant improvement in terms of GoF with respect to the LSM without clustering.
1.6 Conclusions

This chapter provided an overview of a number of social network models emphasising the computational perspective. In fact, the most important issue associated to statistical social network models is concerting their computational complexity which requires the development of efficient inferential algorithms and software able to deal with the increasing size of relational data available.

In particular, we have presented some recent advanced Bayesian approaches to parameter estimation of exponential random graph models and latent variable network models. We demonstrated that Bayesian inference is a very helpful and powerful approach allowing a formal treatment of uncertainty using the rules of probability.

We discussed how Bayesian parameter estimation for exponential random graph models and latent space models is a computationally intensive problem that can be tackled using advanced MCMC and variational techniques. We illustrated the main capabilities of the Bergm, latentnet, VBLPCM and lvm4net packages for the open-source R software through a tutorial analysis of a well-known social network dataset. For each modelling approach we have also considered a Bayesian graphical test of goodness of fit to assess whether or not a given parametric model is compatible with the observed network data.
Advances in the Bayesian methodology and computing will prove crucial to effectively capture heterogeneity and organise different sources of information commonly available in social network data. For this reason, we believe statistical social network analysis will become fertile ground for interdisciplinary research in advanced statistics and social network analysis applications.
1. A. Caimo and N. Friel. Bayesian inference for exponential random graph models. *Social Networks*, 33(1):41 – 55, 2011.
2. A. Caimo and N. Friel. Bayesian model selection for exponential random graph models. *Social Networks*, 35(1):11 – 24, 2013.
3. A. Caimo and N. Friel. Bergm: Bayesian exponential random graphs in R. *Journal of Statistical Software*, 61(2):1–25, 2014.
4. A. Caimo and A. Mira. Efficient computational strategies for doubly intractable problems with applications to bayesian social networks. *Statistics and Computing*, 25(1):113–125, 2015.
5. P. Erdös and A. Rényi. On random graphs. *Publicationes Mathematicae*, 6:290–297, 1959.
6. O. Frank and D. Strauss. Markov graphs. *Journal of the American Statistical Association*, 81:832–842, 1986.
7. I. Gollini. *lvm4net: Latent Variable Models for Networks*. Latent variable models for network data using fast inferential procedures., 2015. R package version 0.2.
8. I. Gollini and T. B. Murphy. Joint modelling of multiple network views. *Journal of Computational and Graphical Statistics*, page to appear, 2014.
9. M. S. Handcock, D. R. Hunter, C. T. Butts, S. M. Goodreau, and M. Morris. statnet: Software tools for the representation, visualization, analysis and simulation of network data. *Journal of Statistical Software*, 24(1):1–11, 2007.
10. M. S. Handcock, A. E. Raftery, and J. M. Tantrum. Model-based clustering for social networks. *Journal Of The Royal Statistical Society Series A*, 170(2):301–354, 2007.
11. P. D. Hoff, A. E. Raftery, and M. S. Handcock. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97:1090–1098, 2002.
12. P. W. Holland and S. Leinhardt. An exponential family of probability distributions for directed graphs (with discussion). *Journal of the American Statistical Association*, 76:33–65, 1981.
13. D. R. Hunter, S. M. Goodreau, and M. S. Handcock. Goodness of Fit of Social Network Models. *Journal of the American Statistical Association*, 103(481):248–258, 2008.
14. D. R. Hunter and M. S. Handcock. Inference in curved exponential family models for networks. *Journal of Computational and Graphical Statistics*, page to appear, 2014.
15. Michael I Jordan, Z Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.
16. Pavel N. Krivitsky and Mark S. Handcock. Fitting position latent cluster models for social networks with latentnet. *Journal of Statistical Software*, 24(5), 2008.
17. Pavel N. Krivitsky and Mark S. Handcock. latentnet: Latent Position and Cluster Models for Statistical Networks. The Statnet Project (http://www.statnet.org), 2015. R package version 2.7.0.
18. Pavel N. Krivitsky, Mark S. Handcock, Adrian E. Raftery, and Peter D. Hoff. Representing degree distributions, clustering, and homophily in social networks with latent cluster random effects models. *Social Networks*, 31(3):204–213, 2009.
19. D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and Dawson S. M. The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations. *Behavioral Ecology and Sociobiology*, 54:396–405, 2003.
20. Martina Morris, Mark S Handcock, and David R Hunter. Specification of exponential-family random graph models: terms and computational aspects.
Iain Murray, Zoubin Ghahramani, and David MacKay. MCMC for doubly-intractable distributions. In Proceedings of the 22nd Annual Conference on Uncertainty in Artificial Intelligence (UAI-06), Arlington, Virginia, 2006. AUAI Press.

K. Nowicki and T. A. B. Snijders. Estimation and prediction for stochastic blockstructures. Journal of the American Statistical Association, 96:1077–1087, 2001.

R Development Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, 2011.

R. M. Ripley, T. A. B. Snijders, and P. Preciado. Manual for RSiena. University of Oxford: Department of Statistics, Nuffield College, 2011.

G. Robins, P. Pattison, Y. Kalish, and D. Lusher. An introduction to exponential random graph models for social networks. Social Networks, 29(2):169–348, 2007.

M. Salter-Townshend, A. White, I. Gollini, and T. B. Murphy. Review of statistical network analysis: models, algorithms, and software. Statistical Analysis and Data Mining, 5(4):243–264, 2012.

Michael Salter-Townshend and Thomas Brendan Murphy. Variational bayesian inference for the latent position cluster model for network data. Computational Statistics & Data Analysis, 57(1):661–671, 2013.

Michael Schweinberger, Mark Handcock, and Pamela Luna. hergm: Hierarchical Exponential-Family Random Graph Models with Local Dependence, 2015.

T. A. B. Snijders, P. E. Pattison, G. L. Robins, and Handcock M. S. New specifications for exponential random graph models. Sociological Methodology, 36:99–153, 2006.

M. A. van Duijn, T. A. B. Snijders, and B. H. Zijlstra. p2: a random effects model with covariates for directed graphs. Statistica Neerlandica, 58:234–254, 2004.

S. Wasserman and P. Pattison. Logit models and logistic regression for social networks: I. An introduction to Markov graphs and p*. Psychometrika, 61:401–425, 1996.

Bibliography

1 A. Caimo and N. Friel. Bayesian inference for exponential random graph models. Social Networks, 33(1):41 – 55, 2011.

2 A. Caimo and N. Friel. Bayesian model selection for exponential random graph models. Social Networks, 35(1):11 – 24, 2013.

3 A. Caimo and N. Friel. Bergm: Bayesian exponential random graphs in R. Journal of Statistical Software, 61(2):1–25, 2014.

4 A. Caimo and A. Mira. Efficient computational strategies for doubly intractable problems with applications to bayesian social networks. Statistics and Computing, 25(1):113–125, 2015.

5 P. Erdös and A. Rényi. On random graphs. Publicationes Mathematicae, 6:290–297, 1959.

6 O. Frank and D. Strauss. Markov graphs. Journal of the American Statistical Association, 81:832–842, 1986.

7 I. Gollini. lvm4net: Latent Variable Models for Networks. Latent variable models for network data using fast inferential procedures., 2015. R package version 0.2.

8 I. Gollini and T. B. Murphy. Joint modelling of multiple network views. Journal
M. S. Handcock, D. R. Hunter, C. T. Butts, S. M. Goodreau, and M. Morris. statnet: Software tools for the representation, visualization, analysis and simulation of network data. *Journal of Statistical Software*, 24(1):1–11, 2007.

M. S. Handcock, A. E. Raftery, and J. M. Tantrum. Model-based clustering for social networks. *Journal Of The Royal Statistical Society Series A*, 170(2):301–354, 2007.

P. D. Hoff, A. E. Raftery, and M. S. Handcock. Latent space approaches to social network analysis. *Journal of the American Statistical Association*, 97:1090–1098, 2002.

P. W. Holland and S. Leinhardt. An exponential family of probability distributions for directed graphs (with discussion). *Journal of the American Statistical Association*, 76:33–65, 1981.

D. R. Hunter, S. M. Goodreau, and M. S. Handcock. Goodness of Fit of Social Network Models. *Journal of the American Statistical Association*, 103(481):248–258, 2008.

D. R. Hunter and M. S. Handcock. Inference in curved exponential family models for networks. *Journal of Computational and Graphical Statistics*, 15:565–583, 2006.

G. Robins, P. Pattison, Y. Kalish, and D. Lusher. An introduction to exponential random graph models for social networks. *Social Networks*, 29(2):169–348, 2007.

M. Salter-Townshend, A. White, I. Gollini, and T. B. Murphy. Review of statistical network analysis: models, algorithms,
and software. *Statistical Analysis and Data Mining*, 5(4):243–264, 2012.

27 Michael Salter-Townshend and Thomas Brendan Murphy. Variational bayesian inference for the latent position cluster model for network data. *Computational Statistics & Data Analysis*, 57(1):661–671, 2013.

28 Michael Schweinberger, Mark Handcock, and Pamela Luna. *hergm: Hierarchical Exponential-Family Random Graph Models with Local Dependence*, 2015.

29 T. A. B. Snijders, P. E. Pattison, G. L. Robins, and Handcock M. S. New specifications for exponential random graph models. *Sociological Methodology, 36*:99–153, 2006.

30 M. A. van Duijn, T. A. B. Snijders, and B. H. Zijlstra. p2: a random effects model with covariates for directed graphs. *Statistica Neerlandica*, 58:234–254, 2004.

31 S. Wasserman and P. Pattison. Logit models and logistic regression for social networks: I. An introduction to Markov graphs and p*. *Psycometrica*, 61:401–425, 1996.