The Conditional-Potts Clustering Model

Alejandro Murua, and Nicolas Wicker

Abstract

A Bayesian kernel-based clustering method is presented. The associated model arises as an embedding of the Potts density for label membership probabilities into an extended Bayesian model for joint data and label membership probabilities. The method may be seen as a principled extension of the so-called super-paramagnetic clustering. The model depends on three parameters: the temperature, the kernel bandwidth and the joint pair-membership probability threshold. We elucidate an informative prior based on random graph theory and kernel density estimation. The clustering is estimated automatically by setting the parameters at the modes of their posteriors. Two stochastic Wang-Landau-like algorithms are presented to estimate the posteriors. We also develop an efficient clustering procedure based only on the modes of our informative prior.

Keywords: Super-paramagnetic clustering, random cluster model, kernel-based method, MAP estimator, Wang-Landau, population Monte Carlo.

1 Introduction

Clustering with the Potts model is a rather recent nonparametric technique introduced by Blatt, Domany and Wiseman (1996a,b, 1997) under the name of super-paramagnetic clustering. There is a large literature on the subject in the Physics community (Wiseman, Blatt and Domany 1998; Agrawal and Domany 2003; Ott et al. 2004; Ott et al. 2005; Reichardt and Bornholdt 2004). Its impact has reached the medical (Stanberry, Murua and Cordes 2008), bioinformatics (Getz et al. 2000; Domany 2003; Einav et al. 2005) and the computer science and machine learning (Domany et al. 1999; Quiroga, Nadasdy and Ben-Shaul 2004) communities as well. It was introduced to the statistical community as Potts model clustering in (Murua, Stanberry and Stuetzle 2008), where its link with other kernel-based methods and nonparametric density estimation was presented. A similar, simpler model has also been used as a probabilistic framework for k-nearest-neighbor classification (Cucala, Marin, Robert and Titterington 2009). One of the main advantages of super-paramagnetic clustering over other kernel-based methods is the simultaneous estimation of the clustering and the number of clusters. That is, there is no need to specify the number of clusters a priori. The method uncovers it automatically. From the statistical point of view, one of the advantages of the method is its connection to the well-known Potts model density, also known as the Boltzmann’s density. Having a probabilistic framework aids in making inferences about the clusters and their number. In fact, the clustering is estimated by Monte Carlo simulation of the labels' distribution.

1Alejandro Murua is Associate Professor, Département de mathématiques et de statistique, Université de Montréal, CP 6128, succ. centre-ville, Montréal, Québec H3C 3J7 Canada (e-mail: murua@dms.umontreal.ca); Nicolas Wicker is Assistant Professor, University of Strasbourg, IGBMC, Laboratoire de Bioinformatique et Génomique Intégrative, 1, rue Laurent Fries, 67404 Illkirch, France (e-mail: nicolas.wicker@unistra.fr).

This research was partially supported by NSERC (Natural Sciences and Engineering Research Council of Canada) grant 327689-06. The first author would like to thank the Institut de Mathématiques of l'Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland, for hosting him during the preparation of the manuscript.
Let \( X = \{ x_i \in \mathbb{R}^p, i = 1, \ldots, n \} \) be our data. In the super-paramagnetic clustering framework, the observations form the vertices of a graph. Let us denote this data graph by \( G(X) \). We say that two observations \( x_i, x_j \) are neighbors if their corresponding vertices are joined by an edge of the graph \( G(X) \). In this case, we will write \( i \sim j \). The edge weights are given by the similarities between the neighboring points. We assume that the similarities are given by a Mercer kernel \( k_{ij} = k(x_i, x_j) \), i.e., a continuous symmetric non-negative definite kernel (Girolami, 2002). The similarities usually depend on the distances between the observations \( ||x_i - x_j|| \), and a scale parameter \( \sigma \), the bandwidth, that controls the relative spread of the distances, so that

\[
k_{ij} = k_{ij}(\sigma) = k(x_i, x_j; \sigma) = k(||x_i - x_j||/\sigma).
\]

In practice, the data graph is made rather sparse by setting \( k_{ij} = 0 \) for observations that are too distant. One way to accomplish this is by using the k-nearest-neighbor graph, that is, a graph where the only edges associated with each \( x_i \) are those associated with the k-nearest-neighbor points of \( x_i \).

In many applications, such as in Bioinformatics (e.g., microarray data), the similarities correspond to the correlation between two signals (e.g., signals coming from two different probe-sets or genes that measure the expression levels under several experimental conditions or over certain elapsed time). In this case, \( ||x_i - x_j|| = 2(1 - \text{corr}(x_i, x_j)) \). When the Euclidean distances are used, one may want to control for different variations in spread of the different covariates making up each data point \( x_i \). This may be achieved by considering a multi-bandwidth model (one \( \sigma_j \) for each covariate \( j \in \{1, \ldots, p\} \)), or by standardizing the covariates. In practice, it is hard to estimate just one bandwidth \( \sigma \), let alone \( p \) of them. We prefer to standardize the covariates, or better yet, to use covariate-wise adaptive bandwidths. A general version of the adaptive bandwidth first suggested in (Murua et al. 2008) within the context of the super-paramagnetic clustering procedure is discussed below in Section 4.5.

The super-paramagnetic clustering is based on the Potts model. This assigns labels \( \ell_i \in \{1, \ldots, q\} \) to each observation \( x_i, i = 1, \ldots, n \), so that observations similar to each other are likely to be assigned the same label. It achieves this by penalizing neighbors that do not share the same label. Statistically, let \( z_{\ell i} = 1 \) if \( x_i \) has been assigned to the \( \ell \)-th label class, and zero, otherwise. The Potts model assumes the following density on the labels \( \{z_{\ell i}\} \), given the data \( X \) (or equivalently, the edge-weights \( \{k_{ij}(\sigma)\} \))

\[
p(\{z_{\ell i}\} | \sigma, T, X) = (Z(\sigma, T, X))^{-1} \exp\left\{ -\frac{1}{T} \sum_{i \sim j} k_{ij}(\sigma)(1 - \delta_{ij}) \right\}
\]

where \( \delta_{ij} = \sum_{\ell=1}^q z_{\ell i}z_{\ell j} = 1 \) if and only if \( x_i \) and \( x_j \) are assigned the same label; \( T \), the temperature, is one of the main parameters of interest in this work, and \( Z(\sigma, T, X) \) denotes the normalizing constant of the Potts density. Since any label configuration \( \{z_{\ell i}\} \) defines a partition of the data, we will refer to them simply as partitions. They can be efficiently sampled by Markov Chain Monte Carlo (MCMC) algorithms (Swendsen & Wang 1987, Wolff 1989). The labels do not necessarily indicate cluster membership. However, two observations lying in the same cluster must share the same label. The labels are rather auxiliary variables that help to merge similar observations together. The clusters are obtained by considering a threshold on
the probabilities that two given observations lie in the same cluster (under the Potts model)

\[ Q_{ij} = Q_{ij}(\sigma, T) = p(z_{\ell i} = z_{\ell j} \text{ for some } \ell \in \{1, \ldots, q\} | \sigma, T) = p(\delta_{ij} = 1 | \sigma, T). \]  

(3)

We refer to the \( Q_{ij} \)'s as the membership adjacency probabilities. In practice, the \( Q_{ij} \)'s are estimated from several MCMC partition samples.

The Potts model is closely linked to the random cluster model (Sokal, 1996, Edwards and Sokal 1988). It is this link that allows the estimation of the clustering in an efficient way. Whereas, it is through the Potts density that one imposes a penalty on neighboring data points, and hence a model on the data, it is through the connected components of the sub-graph yielded by the associated random cluster model that the clusters are estimated (see Section 2 for further details on how the sub-graph is obtained). The final clustering is done by consensus. Having generated several partitions of the data (e.g. MCMC samples from the Potts/random-cluster model), the probabilities of cluster membership adjacency of two data points are readily estimated as the proportion of times these two data points are in the same component. A new graph, the consensus graph, is constructed using these probabilities. An edge between two data points exists in the consensus graph if and only if the frequency of cluster membership adjacency between these points is larger than a predefined threshold, which is usually set to 0.5. The final “consensus” clustering is formed by the connected components of this graph. The full super-paramagnetic clustering procedure is explained in Blatt et al. (1996) and Murua et al. (2008) (see also Section 2).

**The Conditional-Potts model.** Note that the dependency on data in the super-paramagnetic clustering is present only in the distances (or similarities) through the interactions or edge weights \( k_{ij}(\sigma) \) (for convenience in the notation, we will sometimes drop the \( \sigma \) from the \( k_{ij} \)'s). In what follows, it will be useful to think of the distances (or similarities) as the data. We set our likelihood for the parameters \((\sigma, T)\) to

\[ L(\sigma, T|X, \{z_{\ell j}\}) = p(X, \{z_{\ell j}\} | \sigma, T) \propto \exp\{-\frac{1}{T} \sum_{i \sim j} k_{ij}(\sigma)(1 - \delta_{ij})\}. \]

Given a prior distribution for the parameters, \( \pi(\sigma, T) \), we consider the Bayesian embedding of the super-paramagnetic clustering whose density is proportional to

\[ \exp\{-\frac{1}{T} \sum_{i \sim j} k_{ij}(\sigma)(1 - \delta_{ij})\} \pi(\sigma, T). \]  

(4)

As a consequence, in this framework, the Potts density corresponds to the full conditional of the labels given the data and the parameters \( p(\{z_{\ell j}\} | \sigma, T, X) \). We will refer to the model given by (4) as the **Conditional-Potts clustering model** as opposed to the Potts model or super-paramagnetic clustering procedure given by (2).

A key parameter in both the super-paramagnetic and Conditional-Potts clustering procedures is the “temperature”. It plays the role of a smoothing clustering parameter (Murua et al. 2008). The random edges of the sub-graph yielded by the random cluster model are generated with a temperature-dependent probability. A fundamental problem is to estimate the right (or critical) temperature to obtain an optimal
clustering. The current techniques for the super-paramagnetic clustering rely on multiple Monte Carlo simulations performed at each temperature on a large grid of temperatures. Moreover only ad hoc methods to choose this critical temperature have been reported so far (Blatt et al, 1996; Wiseman et al., 1998; Murua et al, 2008; Stanberry et al., 2008). In contrast, our Conditional-Potts clustering model presents two efficient alternative ways to estimate the clustering. These are obtained as a result of estimating the critical temperature either (1) as the mode of the posterior density of the temperature, or (2) simply as the mode of our suggested very informative prior of the temperature. These two procedures are fully detailed later on in this work.

Another key parameter that has not received much attention in the literature is the threshold on the frequencies of cluster membership adjacency chosen to create the consensus clustering. In this work, we show that, together with the temperature, this is a very critical parameter. Our experiments of Section 6 show dramatic clustering differences depending on its value. Therefore, it is advantageous to conceive a more suitable way to choose it. A third parameter to consider is the kernel bandwidth $\sigma$. Most of the literature treats it as fixed. It is usually set to the mean of the data point similarities. However, Murua et al. (2008) showed that treating it as variable may give better clustering results. They suggest the use of an adaptive bandwidth (Abramson, 1982).

Our Conditional-Potts clustering model introduces a complete principled way to select these three critical parameters (temperature, bandwidth and membership adjacency probability threshold) so as to obtain a good consensus clustering. This is accomplished through the elucidation of a very informative prior for the model parameters, and by the development of a stochastic algorithm to estimate the parameter posteriors. By using the link between the Potts and the random cluster models, we elucidate via random graph theory an informative prior for the temperature. Through the connection between kernel-based methods and kernel density estimation, we elucidate an informative prior for the kernel bandwidth. We estimate the posterior densities of the temperature and kernel bandwidth parameters with an algorithm adapted from a Wang-Landau flat-histogram-like algorithm (2001) and the stochastic population Monte Carlo algorithm suggested by Atchade, Lartillot and Robert (2008). Our algorithm is described in Section 4.3. The final data clustering is estimated as the consensus clustering associated with the posterior modes $(\sigma_M, T_M)$ of both the kernel bandwidth and temperature. The corresponding threshold on the frequencies of cluster membership adjacency is set to the mode of the full posterior conditional density of these frequencies given $\sigma_M$ and $T_M$. We also show a variant of this procedure that does not assume a fixed (and unknown) kernel bandwidth $\sigma$, but replaces it by a data-driven adaptive bandwidth $\sigma(x_i, x_j)$.

Note that contrary to several clustering procedures estimated using either MCMC techniques or the EM algorithm, the switch-label problem does not affect the Conditional-Potts clustering algorithm. The clustering is estimated through the consensus graph which in turns depends on estimates of the membership adjacency probabilities $Q_{ij}$. These are the quantities that need to converge and need to be monitored during the evolution of the stochastic algorithm.

Our experiments of Section 6 show that there is no need to pinpoint an optimal value of $(\sigma, T)$
with great precision. Indeed, it seems that a rough idea of the location of the optimal clustering in the temperature-bandwidth space is enough to find a good data clustering. This suggests that good candidates for \((\sigma, T)\) can be found just in a discrete grid of the temperature-bandwidth space. In computational terms, this means that a (simple) discrete version of the Wang-Landau algorithm may be suitable for clustering with the Conditional-Potts clustering model. We show this point in the discussion section. Besides the reduction in complexity of the stochastic algorithm, the discrete Wang-Landau version of our algorithm allows the comparison of the results obtained by using the informative prior with the results obtained without using a prior at all. Our experiments show that our informative prior do not hinder the results. It rather helps to locate good values of the parameters.

Another very important contribution of this work is the introduction of a fully automated and much less computational intensive procedure to do the clustering with the Potts model. The procedure is heavily based on our elucidated informative priors to choose the critical parameters. The final data clustering is estimated as the consensus clustering associated with the modes \((\sigma_p, T_p)\) of the kernel bandwidth and temperature informative priors. The corresponding threshold on the frequencies of cluster membership adjacency is set to the mode of the full conditional density of these frequencies given \((\sigma_p, T_p)\). We refer to this latter procedure as an informed Conditional-Potts clustering. We show in Section 6 that its performance is similar to that of the clustering methods based on the posterior modes. We suggest the use of the informed Conditional-Potts clustering whenever the computational cost is an issue, and especially when dealing with very large data sets.

We note that in contrast to the previous work on clustering with the Potts model, all procedures presented in this work are fully automated procedures in that there is no need to set the kernel-bandwidth nor the membership adjacency frequency threshold a priori, and there is no need to simulate the model at multiple temperatures on a given temperature-grid.

The paper is organized as follows. Section 2 briefly describes the basic facts and issues about the parameters involved in our clustering procedure. The elucidation of the informative priors for the temperature and kernel bandwidth parameters for our proposed Bayesian embedding of the super-paramagnetic clustering is presented in Section 3. The stochastic procedures to estimate the posterior densities of the parameters are described in Section 4. The estimation of the optimal consensus clustering is described in Section 5. The results of the application of our methodology to certain artificial and real data sets are shown in Section 6. Section 7 presents some conclusions and a discussion. The proof of the lemma that leads to the elucidation of an informative prior for the temperature is deferred to Section A of the appendix.

2 The three key parameters for clustering

2.1 The kernel bandwidth

From now on we will assume that the kernel is of the form (1). A common kernel used in applications is the Gaussian kernel, \(k(x) = \exp(-x^2)\).
The choice of the kernel bandwidth is a rather difficult problem that has not been sufficiently addressed in the literature. In most kernel based methods the bandwidth is assumed known and fixed. A common choice is the average distance among the data points. Here we present two sound ways to address the problem. First, as one of the parameters of the Conditional-Potts clustering model, it can be estimated through its posterior density. But, we may also suppose that \( \sigma \) is not a fixed parameter but rather a function that depends on the data around the points \( x_i, x_j \). In this case, we can estimate this function using an adaptive bandwidth (Abramson, 1982) based on a coarse estimate of the data density. This idea was first introduced by Murua et al. (2008) within the context of Potts model clustering. In our experiments, this choice of bandwidth yielded comparable results to those that assume that the bandwidth is a model parameter.

### 2.2 The threshold for consensus clustering

As mentioned in the introduction, the final clustering is estimated by consensus. A consensus graph is constructed by first making a copy of the original data graph \( G(X) \). However, some edges are now removed from the graph according to whether or not the \( Q_{ij} \)'s (see (3)) surpass a certain threshold \( Q \). The edge between the neighbors \( x_i \) and \( x_j \) is kept in the consensus graph if and only if \( Q_{ij} \) is larger than certain threshold probability \( Q \). The connected components of the resulting graph correspond to the clusters. In the current literature, \( Q \) is usually set at \( Q = 0.5 \), without further ado about this choice. We will see later that fixing \( Q \) a priori may not be a good idea.

In practice, the consensus clusters are based on MCMC estimates \( \hat{Q}_{ij}(\sigma, T) \) of the probabilities \( Q_{ij}(\sigma, T) \). The MCMC generates samples \( \{z_{\ell i}\} \) from the label distribution given by the Potts density (2). There are efficient algorithms based on variable augmentation techniques to perform the sampling, e.g. the Swendsen Wang algorithm (Swendsen & Wang, 1987), the Wolff algorithm (Wolff, 1989). Associated with each edge of the graph, a new Bernoulli variable \( b_{ij} \), the bond, is introduced. The bond is set to 1 with probability \( p_{ij}(\sigma, T) = p_{ij}(\sigma, T, \{k_{ij}\}) = 1 - \exp(-k_{ij}(\sigma)/T) \), if \( i \sim j \) and \( x_i \) and \( x_j \) share the same label; otherwise the bond is set to zero. The bonds are auxiliary variables that help split the clusters. The resulting joint label/bond model is known as the Fortuin-Kasteleyn-Swendsen-Wang model (Sokal 1996). Its density is given by

\[
p(\{z_{\ell i}\}, \{b_{ij}\}|\sigma, T, \{k_{ij}(\sigma)\}) \propto \prod_{i \sim j} \{(1 - p_{ij}(\sigma, T))(1 - b_{ij}) + p_{ij}(\sigma, T)b_{ij}\delta_{ij}\}. \tag{5}
\]

The marginal density over the labels is exactly the Potts model. The marginal over the bonds is known as the random-cluster model. Later on, we will exploit the link between these two models to elucidate a prior for \( T \). The \( Q_{ij} \)'s are directly related to the number of times any two given observations occur in the same connected component arising from the random cluster model (5) (Edwards and Sokal 1988).

### 2.3 The temperature

A key problem is to find the “right” temperature \( T \) associated with the “true” clustering. If \( T \) is set too low, then fewer clusters are favored by the model. Whereas if \( T \) is set too hot, then a multitude of clusters
are favored by the model. This can be seen from the fact that the log-density of the random clusters model for given \( T \) and \( \sigma \) is proportional to

\[
\sum_{i \sim j, b_{ij} = 1} \log(1 - e^{-k_{ij}(\sigma)/T}) - \sum_{i \sim j, b_{ij} = 0} k_{ij}(\sigma)/T + C(\{b_{ij}\}) \times \log(q),
\]

where \( C(\{b_{ij}\}) \) denotes the number of connected components given the current values of the bonds, and \( q \) is the number of possible labels (Sokal 1996). The current methodology consists of trying several values of \( T \) on a large preset grid of admissible values of \( T \). At each temperature \( T \) some statistics linked to the current clustering status are computed. The trajectories of these \( T \)-dependent statistics are then followed over the grid of values of \( T \). The most common statistics is the change in size of the largest cluster found at temperature \( T \). This is the variance of the size of the largest cluster (Murua et al. 2008) or the variance of the magnetization (Blatt et al. 1996). Other statistics have been suggested as well, such as the number of clusters, and the similarity between clusterings at consecutive temperatures (Stanberry et al. 2008). There are several problems with this approach. The main ones are (1) the cost of computing the trajectories and (2) the subjectivity in choosing the “right” temperature. The trajectories are usually estimated through large MCMC simulations, and the right temperature is located by selecting an appropriate “good” local maximum on these trajectories. With the Conditional-Potts clustering model, we eliminate these two problems by posing an alternative model that is also based on the Potts density. The problem of finding a good temperature for clustering is transformed to estimating the posterior density of the temperature.

3 Elucidating sensible priors for the parameters

Searching for local maxima in the \( T \)-dependent trajectories (see the previous section) corresponds to searching for cluster splits. For example, a large variance in the size of the largest cluster indicates an imminent split of the cluster. There is a vast literature on this subject for the random cluster model with constant edge-weights. In this model, \( p_{ij}(\sigma, T) = p(\sigma, T) \), for all \( x_i, x_j \). The probability of having a bond \( b_{ij} = 1 \) is then constant. The corresponding random cluster model density reduces to

\[
q^n C p(\sigma, T)^{n_1} (1 - p(\sigma, T))^{n_0},
\]

where \( n_1 = \) number of edges with bond \( b_{ij} = 1 \), \( n_0 = \) number of edges with bond \( b_{ij} = 0 \) among neighboring data points, and \( C = C(\{b_{ij}\}) = \) number of connected components in the graph. This corresponds to a random graph with parameter \( p = p(\sigma, T) \). If the original data graph is a \( r \)-nearest-neighbors graph, then the resulting random graph is a \( r \)-regular random graph, i.e., each vertex has exactly \( r \) edges. The search for a cluster split in the data corresponds to the appearance of a giant (connected) component in the graph (Bollobas, 2001, p. 138, Chapter 6), that is a connected component consisting of more than half the number of vertices of the graph. It can be shown that if \( p \) is large enough, then this is the only non-trivial component of the graph. The appearance of a giant component is also referred to as a phase transition in the graph. The probability \( p \) causing it is the phase transition probability or the critical probability. Since, in general, our weights \( k_{ij}(\sigma) \) are not constant, we cannot directly use these results to
find a good temperature for clustering. However, we can still devise a method to elucidate a good prior density for the sought temperatures based on random graph theory. Let \( G(k_{ij}(\sigma)) \) be the graph with the same vertices and edges of the original data graph \( G(X) \), but whose edges have now constant weight, so that 
\[
p = \exp\{-k_{ij}(\sigma)/T\}
\]
for all existing edges in \( G(X) \) (we stress here that all edge weights are equal to the constant \( k_{ij}(\sigma) \), i.e., the pair of indexes \((i, j)\) is fixed). If \( m \) is the number of edges in \( G(X) \), then there are about \( m \) graphs \( G(k_{ij}(\sigma)) \), one for each different edge weight. The idea is to find estimates of the random graph phase transition probabilities for each of the graphs \( G(k_{ij}(\sigma)) \), \( i, j \in \{1, \ldots, n\} \). These are then used to find estimates of the critical temperatures \( T_{ij} \) associated with each particular value of the edge weight \( k_{ij}(\sigma) \). We show below in Section 3.1 the explicit connection between these probabilities and the critical temperatures. The sample of temperatures so obtained is used to construct a density estimate of critical temperatures. This density depends on all edge weights present in the data graph, and hence, on a given bandwidth \( \sigma \). Consequently, we are able to elucidate a conditional prior that will be denoted by \( \pi(T|\sigma) \). Its formal derivation is given below. Obviously, due to its dependence on the similarities between the observations, our critical temperature prior is a data-driven prior.

### 3.1 The temperature prior: the connection to random graphs

The critical temperatures are inferred by using the random cluster model associated with the Potts model with equal edge weights. We consider random \( r \)-regular graphs of order \( n = \) the number of data points in \( X \). In the random cluster model the degree \( r = r(T) \) of the vertices depends on the probabilities of having bonds equal to one \((b_{ij} = 1)\), which are given by 
\[
p_{ij} = 1 - \exp\{-k_{ij}(\sigma)/T\}.\]
The idea is to increase the degree \( r \) until a giant component appears. This would signal a merging of the clusters, and hence a cluster transition. Since the critical degree \( r_c \) depends on the temperature, the transition must occur at a particular temperature \( T_c = r^{-1}(r_c) \). This computation is done separately for every value \( p_{ij} \). The prior for the critical temperatures is then obtained by estimating the density associated with the set of critical temperatures \( \{T_c(i, j)\} \).

To simplify the calculation, we suppose that each possible \( r \)-regular graph \( G(n, r) \) generated from the data has the same probability. The problem can be formulated as finding the value of \( r \) so that, with high probability, \( G(n, r) \) is connected. To find a bound on this probability, we follow the second proof of Theorem 7.3 in (Bollobas, 1985). This gives the probability that a giant component appears as the number of edges \( r \) increases. Let \( P_G \) denote the probability of having a connected component in \( G(n, r) \) of order at most \( n/2 \). Note that there cannot be a component of less than \( r + 1 \) vertices as each vertex is connected to \( r \) other vertices.

**Lemma.** Let
\[
s = n^{1-r/2} \sqrt{2\pi} e^{1-r^2/4} 2^{-r/2}.\]

Then
\[
P_G \leq s^{r+1}/\sqrt{2\pi}(r + 1)^{5/2}(1 - s)\]
The proof is postponed to the Appendix. Suppose now that this bound is upper bounded by $p_G$. Then with probability at least $1 - p_G$ there is only a giant component, as isolated vertices cannot exist. We choose the smallest value of $r$, say $r_c = r(p_G)$, that satisfies

$$s^{r+1}/\sqrt{2\pi(r+1)^{5/2}}(1-s) \leq p_G.$$ 

Typically, we set $p_G$ to be equal to a small value such as 0.01.

Note that the number of edges necessary for the connectivity of the graph is then: $r_c n/2$. Assuming that all edges are equally likely to occur, we have $p = p_c = r_c n/(2m)$, where $m$ is the number of possible edges in the graph. For a $k$-nearest-neighbor graph, $m$ is bounded by $nk$. The corresponding critical temperature is then given by:

$$T_c = T(r_c, p_c) = T(p_c) = -k_{ij}/\log(1-p_c) = -k_{ij}/\log \left(1 - \frac{r_c n}{2m}\right).$$

### 3.2 The bandwidth prior: the connection with kernel density estimation

Exploiting the connection between Potts model clustering and kernel density estimation (Murua et al. 2008), we derive a prior for the bandwidth parameter based on an adaptive bandwidth kernel density estimator. In many applications the data graph corresponds to a $k$-nearest-neighbor graph. A quick estimate $\hat{p}_{knn}(x)$ of the data density is readily available in this case. The $k$-nearest neighbors adaptive bandwidth at the observation $x_i$ is given by (Breiman, Meisel and Purcell 1977, Abramson 1982, Silverman 1986 Section 5.3)

$$\sigma_{knn}(x_i) \propto \exp \left\{ -0.5 \left( \log [\hat{p}_{knn}(x_i)] - \frac{1}{n} \sum_{j=1}^{n} \log [\hat{p}_{knn}(x_j)] \right) \right\}. \quad (6)$$

Together, the adaptive bandwidths can be considered as a sample of possible values for the bandwidth parameter. We use an empirical estimate of the density associated with the bandwidths $\{\sigma_{knn}(x_i)\}$ as our prior for $\sigma$, which we will denote by $\pi(\sigma)$. In practice, in order to construct the $k$-nearest-neighbor graph, we set the value of $k$ to 10 or 20. Smaller values of $k$ loose information about the connectivity of the complete graph (the graph with all possible edges), whilst larger values of $k$ augment the computational cost considerably (from order $O(n)$ to order $O(n^2)$ if $k$ is set to $n - 1$). Blatt et al. (1997) also argued that a moderate value of $k$ is preferred.

In our experiments of Section 6, we have used a kernel density estimator for both $\pi(T|\sigma)$ and $\pi(\sigma)$. The kernel employed was the Epanechnikov kernel with bandwidths set to the standard deviations of the corresponding samples. Figure 2 shows the priors for some of the data sets used in our experiments of Section 6.

### 4 Sampling from the posterior distribution

The normalizing constant $Z(\sigma, T)$ of the Conditional-Potts clustering model is unknown and intractable when the data size is moderately large. Therefore, sampling directly from the posterior $\pi(\sigma, T|X)$ is
not possible. In addition, even MCMC methods such as the Metropolis-Hastings algorithm to generate samples \((\sigma', T')\) from the posterior do not work, since knowledge of the ratio \(Z(\sigma, T)/Z(\sigma', T')\) is necessary.

The literature suggests several possibilities to overcome this problem. The most popular technique seems to be based on path sampling (Ogata, 1989; Gelman & Meng, 1998; Richardson & Green, 1997). However, this involves the estimation of the normalizing constant for all \((\sigma, T)\), or at least for a reasonable grid of values of the parameters; it does not take into account the uncertainty in the estimators arising from the Monte Carlo integration. Instead, we follow a different adaptive MCMC procedure suggested by Atchade, Lartillot and Robert (2008). Section 4.3 describes our adaption of their algorithm for the Conditional-Potts clustering model. It relies on the Wang-Landau algorithm (Wang and Landau 2001) to produce a “flat-histogram” on the parameter space. Atchade et al’s algorithm (Atchade et al. 2008) generates a stochastic process for which the marginal of \((\sigma_m, T_m)\) converges to the desired posterior distribution as \(m \to +\infty\). The process is not necessarily Markovian, although it involves Metropolis-Hastings-like sampling. The main idea is to replace the normalizing constant \(Z(\sigma, T)\) by a series of stochastic approximations of it. These are derived through population Monte Carlo techniques using a small fixed grid of values of the parameter space. A variant of the Wang-Landau algorithm is used to sample on this parameter-space grid.

### 4.1 The Wang-Landau algorithm

In the original Wang-Landau algorithm, the goal is to sample from the target distribution, say \(\pi(u)\), for \(u\) in one of the “energy” states \(g \in \{1, 2, \ldots, d\}\). But \(\pi(u)\) is only known up to a normalizing constant. Wang and Landau (2001) suggested sampling instead from \(\pi_c(u) \propto \sum_{g=1}^{d} \pi(u)_{c(g)} 1_g(u)\), where \(c(\cdot)\) is a function of the energies. Once an energy state \(g\) is visited, \(c(g)\) is modified so as to make another visit to \(g\) more unlikely. In fact, at iteration \(m + 1\) of the sampler, \(c_{m+1}(g) = c_m(g)(1 + \gamma_m 1_g(U_{m+1}))\), where \(U_{m+1}\) denotes the current state of \(U\). The sequence \(\gamma_m\) is a slowly decreasing random sequence that controls the amount of penalty given to current visited energies. The goal is to make visits to all energy states uniformly as \(m \to \infty\). If this is the case, \(\pi_{c_m}(u \in g) \approx \frac{1}{d} \propto \frac{\pi(u)}{c_m(g)}\), which implies that \(c_m(g) \propto \pi(u \in g)\).

Hence, \(c_m(g)/\sum_{g'=1}^{d} c_m(g')\) may be used as an estimate of the probability of the energy state \(g\). The algorithm seems to be very efficient (Shell, Debenedetti and Panagiotopoulos 2002; Dayal et al. 2004; Ghulghazaryan, Hayryan and Hu 2006; Zhou et al. 2006). Atchade et al. (2008) have extended it to more sophisticated target densities without assuming a discrete energy space. We have adapted this latter procedure to our problem.

### 4.2 Decomposition of the normalizing constant

As mentioned above, the data \(X\) enters into the model through the distances \(D_{ij} = ||x_i - x_j||\) (or similarities) used to evaluate the edge weights. Without loss of generality, we may assume that all distances lie in the interval \([0, 1]\) (the scale parameter \(\sigma\) will absorb the differences in scaling). To simplify the calculation, we will assume that there is no interaction among the distances (or similarities). This
assumption yields

\[
Z(\sigma, T) = \int_{D_{ij}} dD_{ij} \sum_{\{z_{ti}\}} \exp\left[-\frac{1}{T} \sum_{i \neq j} k_{ij}(\sigma) (1 - \delta_{ij})\right]
\]

\[
\approx \sum_{\{z_{ti}\}} \prod_{i \neq j} \int_{D_{ij}} \exp\left[-\frac{1}{T} k_{ij}(\sigma) (1 - \delta_{ij})\right] dD_{ij}
\]

\[
= \sum_{\{z_{ti}\}} \left[ \prod_{\delta_{ij}=1} \int_0^1 dD_{ij} \right] \left[ \prod_{\delta_{ij}=0} \int_0^1 e^{-\frac{k_{ij}(\sigma)}{T}} dD_{ij} \right]
\]

\[
= \sum_{\{z_{ti}\}} \left[ \int_0^1 e^{-\frac{k_{ij}(\sigma)}{T}} dD_{ij} \right] \sum_{i \neq j} (1 - \delta_{ij}) = \sum_{\{z_{ti}\}} \exp\left[-\beta(\sigma, T) \sum_{i \neq j} (1 - \delta_{ij})\right],
\] (7)

where \(\beta(\sigma, T) = -\log(\int_0^1 e^{-\frac{k(D_{ij}/\sigma)}{T}} dD_{ij})\) (this quantity can be easily computed for all \((\sigma, T)\) that need to be evaluated). Note that \(Z(\sigma, T)\) turns out to be approximately the normalizing constant of the Potts density with constant interaction \(\beta(\sigma, T)\). Below, we will use this approximation to build our stochastic algorithm.

Let \(\kappa(\sigma, T; \sigma', T')\) be a kernel function summing up to unity on a fixed grid \(\{(\sigma^g, T^g)\}_{g=1}^d\), i.e.,

\[
\sum_{g=1}^d \kappa(\sigma, T; \sigma^g, T^g) = 1
\]

for all \((T, \sigma)\). A population Monte Carlo estimate of \(Z(\sigma, T)\) can be derived from the following identity that combines (7) with the decomposition of the normalizing constant suggested in Atchade et al. (2008)

\[
Z(\sigma, T) \approx \sum_{g=1}^d \kappa(\sigma, T; \sigma^g, T^g) Z(\sigma^g, T^g)
\]

\[
\times \sum_{\{z_{ti}\}} \exp\left[-(\beta(\sigma, T) - \beta(\sigma^g, T^g)) \sum_{i \neq j} (1 - \delta_{ij})\right] \{\exp(-\beta(\sigma^g, T^g) \sum_{i \neq j} (1 - \delta_{ij}))\}/Z(\sigma^g, T^g)
\]

\[
= \sum_{g=1}^d \kappa(\sigma, T; \sigma^g, T^g) Z(\sigma^g, T^g) E_{\{\beta(\sigma^g, T^g)\}} \{\exp[-(\beta(\sigma, T) - \beta(\sigma^g, T^g)) \sum_{i \neq j} (1 - \delta_{ij})]\}
\] (8)

This hints at estimating \(Z(\sigma, T)\) by sampling from the Potts densities \(p(\cdot|\beta(\sigma^g, T^g))\), \(g = 1, \ldots, d\). However, (8) still requires the knowledge of the \(Z(\sigma^g, T^g)\)’s. This problem is overcome by considering instead a stochastic Monte-Carlo algorithm whose proposals \((\sigma', T')\) are generated from approximating full conditional distributions. The size of the grid, \(d\) is a parameter to be chosen that depends on the problem at hand. Atchade et al. (2008) suggest using a moderate value, e.g. 100 (see also Section 6).

### 4.3 A population Monte Carlo algorithm for Potts model clustering (PMC2)

The algorithm of Atchade et al. (2008) appropriately modified for our model has the following steps.

Given the current state of the parameters at iteration \(m\), \(\{z_{ti}^m\}, T^m, \sigma^m\), and \(I^m = g, g \in \{1, \ldots, d\}\):

1. Generate \(\{z_{ti}^{m+1}\}\) from the constant-interaction Potts density \(p(\cdot|\beta(T^m, \sigma^m))\).
2. Generate $I^{m+1}$ from the multinomial distribution on \{1, \ldots, d\} whose probabilities are proportional to $\exp[-\sum_{i \sim j} \beta(T^g, \sigma^g)(1 - \delta_{ij}^{m+1}) - c_m(g)]$, where $\delta_{ij}^{m+1} = \sum_{t=1}^q z_{ti}^{m+1} z_{tj}^{m+1}$, and $c_m(g)$, an estimate of $\log Z(\sigma^g, T^g)$, is given by (9) below.

3. Update the estimate of $\log(Z(\sigma^g, T^g))$ by

$$c_{m+1}(g) = c_m(g) + \gamma_m \frac{\exp[-\sum_{i \sim j} \beta(T^g, \sigma^g)(1 - \delta_{ij}^{m+1}) - c_m(g)]}{\sum_{g' = 1}^d \exp[-\sum_{i \sim j} \beta(T^{g'}, \sigma^{g'})(1 - \delta_{ij}^{m+1}) - c_m(g')]}.$$  

(9)

4. Generate $(\sigma^{m+1}, T^{m+1})$ from

$$\pi_{m+1}(\sigma, T) \propto Z_{m+1}^{-1} \exp[-\sum_{i \sim j} \frac{1}{T} \kappa_{ij}(\sigma)(1 - \tilde{\delta}_{ij}^m)] \times \pi(T|\sigma)\pi(\sigma),$$

where $Z_{m+1}$ is given by

$$Z_{m+1}(\sigma, T) = \sum_{g=1}^d \kappa(\sigma, T; \sigma^g, T^g) e^{c_{m+1}(g)}$$

$$\times \left\{ \sum_{a=1}^{m+1} \exp[-\sum_{i \sim j} (\beta(\sigma^g, T^g) - \beta(\sigma, T))(1 - \delta_{ij}^a)] \mathbf{1}_g(I^a) \right\} / \sum_{a=1}^{m+1} \mathbf{1}_g(I^a),$$

(10)

with $\mathbf{1}_g(a) = 1$ if and only if $g = a$, and is zero otherwise; and $\tilde{\delta}_{ij}^m$ is computed with the labels \{\tilde{z}_{ij}^m\} (see next step).

5. Generate the new labels \{\tilde{z}_{ij}^{m+1}\} from the Potts density (2) with parameters $(\sigma^{m+1}, T^{m+1})$.

We will refer to this algorithm as Population Monte Carlo for Potts Model Clustering, and will denote it for short by PMC2. In the original setup of Atchade et al.’s algorithm (2008), the labels \{\tilde{z}_{ij}^m\} (or, equivalently, \{\tilde{\delta}_{ij}^m\}) are known and correspond to the data. However, in our case, these labels are unknown. Hence, they need to be generated at each iteration.

Note that equation (11) resembles equation (8). The expectation has been approximated by the population Monte Carlo sampling estimate.

Steps (2), (3) and (4) correspond to the original Wang-Landau algorithm, except that this time instead of only sampling from the grid of parameter values, the sampling is also done on the whole space of parameter values (see Step (4)). Step (2) ensures that the sampling on the grid is done so as to obtain on the long run a flat-histogram, i.e. that on the long run all points in the grid are equally likely to be sampled. At convergence $e^{c_m(g)}$ is proportional to the corresponding approximation to the normalizing constant $Z(\sigma^g, T^g)$. The rejuvenation step-size parameter $\gamma_m$ controls the convergence of $e^{c_m(g)}$. It should slowly decrease to zero. We choose to update it according to the heuristic suggested in (Atchade et al. 2008, Section 2.2).
4.4 Implementation of steps (4) and (5)

We implemented step (4) of the PMC2 algorithm with a Metropolis-Hastings step. We note that in the Wang-Landau algorithm this step is replaced by a single Metropolis-Hastings update (Wang and Landau 2001; Atchade and Liu, 2004). In our experiments we have used both a single update (for computational efficiency), and multiple sequential Metropolis-Hastings updates (to try to generate a good candidate draw from $\pi_{m+1}(\sigma, T)$). There does not appear to be a significant difference in the results. A possible explanation for this result is that the priors already put most of the weight in suitable regions of the parameters. As we have mentioned earlier, the priors alone give strong hints of good values of the parameters. In fact, we have developed a very competitive clustering procedure (see Section 5.1) that relies only on the priors to find a good clustering of the data.

We have experimented with three different proposal distributions to update the parameters $(\sigma, T)$.

P1 The parameters are proposed according to their prior distribution $\pi(T|\sigma)\pi(\sigma)$. In our experiments, this proposal does not perform well. It seems that the extent of our prior is too large in comparison with the concentration of values observed in the posterior.

P2 The proposal for $(\sigma, T)$ is generated by a truncated-normal and a log-normal distribution, respectively. First, a log-bandwidth $\log(\sigma')$ is proposed from a normal distribution with mean $\sigma^m$ and variance, $\tau^2$, equal to half the observed variance of the empirical distribution of the bandwidths given by the adaptive bandwidth prior; that is

$$p(\sigma'|\sigma^m) = (\sigma'\tau\sqrt{2\pi})^{-1} \exp\{-0.5\tau^{-2}(\log(\sigma') - \log(\sigma^m))^2\}.$$ 

Then, a temperature $T'$ is proposed from a truncated normal derived from a normal distribution with mean $T^m$ and variance, $\tau^2(\sigma)$, equal to half the observed variance of the empirical distribution of the temperatures given the bandwidth $\sigma'$; that is

$$p(T'|T^m; \sigma') = (\Phi(T^m)\tau'\sqrt{2\pi})^{-1} \exp\{-0.5(\tau')^{-2}(T' - T^m)^2\},$$

where $\Phi(x)$ denotes the normal cumulative distribution evaluated at $x$. The truncated normal is used to ensure that the proposed temperature is positive.

P3 The bandwidth update is proposed as in P2. The temperature update is proposed by an inverse-Gamma$(\alpha, \beta)$ distribution with $\beta = \beta(\sigma) = \exp[\sum_{i\sim j} k_{ij}(\sigma)(1 - \delta_{ij}^m)]$. The parameter $\alpha$ is chosen so that the the modes of the prior $\pi(T|\sigma)$ and the inverse-Gamma proposal are the same. This is done so as to make the shape of the proposal similar to that of the prior. Noting that the mode of the inverse-Gamma$(\alpha, \beta)$ density occurs at $\beta/\alpha$, the shape parameter $\alpha$ is set to $0.5([\beta(\sigma')/\text{mode}(\pi(T|\sigma')) - 1] + [\beta(\sigma^m)/\text{mode}(\pi(T|\sigma^m)) - 1])$.

Once the pair $(\sigma', T')$ is chosen, the labels $\{z'_{ij}\}$ are generated from the Potts density (2) using the Swendsen-Wang algorithm (Swendsen and Wang 1987, Wang and Swendsen 1990).
4.5 Using an adaptive bandwidth (AB PMC2)

In many situations an adaptive bandwidth kernel is preferred to a fixed bandwidth kernel. In this case, \( \sigma \) is no longer a parameter of the model. Therefore, \( \beta(\sigma, T) \) becomes \( \beta(T) \) and the Metropolis-Hastings ratio of Step (4) simplifies to

\[
\min \{ 1, \frac{Z_{m+1}(T')}{Z_{m+1}(T)} \frac{\prod_{i \sim j} k_{ij}(1 - \delta_{ij}^m)(T_m)^{-1} - (T')^{-1})}{\prod_{i \sim j} k_{ij}(1 - \delta_{ij}^m)(T_m)^{-1} - (T')^{-1})}} \frac{\pi(T')}{\pi(T)} \frac{\Phi(T')}{\Phi(T)} \}
\]

for the truncated-normal-proposal P2, and to

\[
\min \{ 1, \frac{Z_{m+1}(T')}{Z_{m+1}(T)} \frac{\prod_{i \sim j} k_{ij}(1 - \delta_{ij}^m)(T_m)^{-1} - (T')^{-1})}{\prod_{i \sim j} k_{ij}(1 - \delta_{ij}^m)(T_m)^{-1} - (T')^{-1})}} \frac{\pi(T')}{\pi(T)} \frac{T'}{T_m}^{a+1} \}
\]

for the inverse-Gamma proposal P3. In our experiments, we have used the adaptive bandwidth given by (6). From a computational point of view, this is a more efficient model. It yields similar results to those of the model with fixed but unknown bandwidth.

4.6 The integrated conditional-Potts clustering algorithm (IPMC2)

As mentioned above, the labels \( \tilde{z} = \{ \tilde{z}_{ij} \} \) are unknown. A variant of Steps (4) and (5) of the PMC2 algorithm consists of looking at the labels \( \tilde{z}'s \) as nuisance parameters. Hence, they can be integrated out of the Conditional-Potts clustering model. The resulting model is given by

\[
p(k_{ij}|\sigma, T) \propto \frac{Z(\sigma, T, X)}{Z(\sigma, T)}.
\]

In this case, one can modify Step (4) of the PMC2 algorithm by generating the parameters \( (\sigma', T') \) from the target distribution

\[
\tilde{\pi}_{m+1}(\sigma, T) \propto \frac{Z(\sigma, T, X)}{Z_{m+1}(\sigma, T)} \pi(T|\sigma)\pi(\sigma).
\]

The Metropolis-Hastings step to generate the updated values for \( (\sigma, T) \) is as in PMC2, i.e., the proposal for the parameters \( (\sigma', T') \) is one of the proposal distributions P1, P2 or P3 as before. But now the ratio

\[
\frac{Z_{m+1}(T', \sigma')}{Z_{m+1}(\sigma', T')} \frac{\prod_{i \sim j} k_{ij}(\sigma')(1 - \delta_{ij}^m)}{\prod_{i \sim j} k_{ij}(\sigma^m)(1 - \delta_{ij}^m)}
\]

of PMC2 is replaced by

\[
\frac{Z_{m+1}(T', \sigma')}{Z_{m+1}(\sigma', T')} \frac{\prod_{i \sim j} k_{ij}(\sigma^m)(1 - \delta_{ij}^m)}{\prod_{i \sim j} k_{ij}(\sigma'(1 - \delta_{ij}^m))}
\]

Note that this algorithm requires the knowledge of the Potts density normalizing constant. In order to estimate the ratio \( Z(\sigma', T', X)/Z(T', \sigma', X) \) we suggest using an iterative approximation analogous to the one given by formula (11), so that \( Z_{m+1}(\sigma, T, X) \) is an approximation to the normalizing constant of the Potts density when the data is not integrated out. We replace the actual ratio by its estimate \( Z_{m+1}(\sigma', T', X)/Z_{m+1}(T'\sigma', X) \). We will refer to this version of PMC2 as IPMC2 for Integrated PMC2.
We note that this modified algorithm can also be derived from the original model by allowing the labels $\tilde{z}$ to be updated together with the parameters in Step (4) of the PMC2 algorithm. The modified algorithm works as follows: the proposal for the parameters $(\sigma', T')$ is one of the proposal distributions $P1, P2$ or $P3$ as before. Given these parameters, the proposed labels $\tilde{z}'$ are generated from the Potts density (2). It is straightforward to check that the new Metropolis-Hastings ratio is identical to the one used by the IPMC2 algorithm.

5 Clustering

The clustering chosen is given by the MAP (maximum a posteriori) estimators of the temperature $T_M$, and the kernel bandwidth, $\sigma_M$, if this latter is a parameter of the model (as opposed to the use of an adaptive bandwidth). The consensus clustering so obtained still depends on the value of the threshold $Q$ for the probabilities $Q_{ij}$ of cluster membership adjacency. It is a common practice to set it a priori to $Q = 0.5$. However, a more principled choice is $Q = Q(\sigma, T)$, the maximizer of the full conditional posterior density of $Q_{ij}$ given the data and the parameters $(\sigma, T)$. We show in our experiments that this choice of the threshold performs very well in comparison to simply setting $Q$ at 0.5. Therefore the final consensus clustering $P_M$ is built by fixing the parameters to the values $(\sigma_M, T_M, Q(\sigma_M, T_M))$. This choice of the parameters makes clustering with the Conditional-Potts clustering model a fully automated procedure.

5.1 The informed Conditional-Potts clustering (Prior + PMC)

One of the main results derived from our experiments (see next section) is that a good clustering may be obtained by maximizing our elucidated priors for the critical temperature and kernel bandwidth. In what follows we will refer to this way of estimating the optimal clustering as Prior+PMC.

Very often in our experiments, the MAP estimators $(\sigma_M, T_M)$ are close to the maximizer $(\sigma_p, T_p)$ of $\pi(\sigma)$ and $\pi(T|\sigma_p)$, respectively. Therefore, the clustering $P_p$ obtained by $(\sigma_p, T_p, Q(\sigma_p, T_p))$ is similar to the one yielded by $(\sigma_M, T_M, Q(\sigma_M, T_M))$. Sometimes, the consensus clustering $P_p$ is even better than $P_M$.

We say that a region is stable if there is high similarity between clusterings yielded by nearby points in the temperature-bandwidth space. Stanberry et al. (2008) observe that these regions are also stable regions for the number of clusters and may correspond to phase-free regions of the random cluster model (i.e., no giant component appears in these regions; see Section 3.1). We believe that the good results achieved by the Prior + PMC procedure indicate that (a) our prior for the parameters may resemble in shape the posteriors near the modes, and/or (b) our prior and posteriors present peaks in the same stable region of the temperature-bandwidth space, thus yielding similar clustering results. Based on our experiments (see next section) we recommend the use of the Prior + PMC procedure when the computation cost or computation time is an issue, for example in the case of large data sets.
6 Experimental results

Here we present a comparison of the clustering performance of the three clustering methods presented above, namely, PMC2, IPMC2 and Prior+PMC, with bandwidth as a parameter or with adaptive bandwidth. As is now customary in the machine learning and clustering literature, we measure the goodness-of-fit of the resulting clusterings with the adjusted Rand index (ARI). The ARI is a measure of similarity (agreement) between two clusterings (or partitions). It was first suggested by Rand (Rand 1975) and then corrected for randomness by Hubert and Arabie (Hubert and Arabie, 1995). A perfect match is signaled by an ARI score of 1.0: the closer the score is to one, the more similar the clusterings are.

In all the simulations, we set a 20-cell grid of values for the kernel bandwidth and a 50-cell grid of values for the temperature. The grids were formed by the corresponding percentiles of the bandwidth prior $\pi(\sigma)$ and the temperature given the bandwidth prior $\pi(T|\sigma)$, respectively. The temperature grid is finer because finding a good temperature is critical. In our experiments, and in general, the bandwidth does not present as wide a range of possible values (as judged by its prior) as the temperature. Also, finer grids will not necessarily yield better clustering results. As we note below, to find a good clustering, it is sufficient to generate a grid that covers stable regions of the temperature-bandwidth space.

The sampling of the parameters was started after a burn-in period of at least 10,000 iterations. We considered only the last 1000 samples after the burn-in period as samples from the posterior distribution. The proposals for the temperature were generated with either a truncated-normal or inverse-Gamma distribution. Below we just show the results obtained with the inverse-Gamma distribution. The results with the truncated-normal were similar. The proposal for the bandwidths were generated with a log-Normal distribution. These proposals are described above. The final clusterings were constructed by consensus clustering (as described in Section 2) at the MAP estimators of the parameters, and at the mode of the full conditional posterior of the $Q_{ij}$ membership adjacency probabilities. We note that in certain cases the posterior distributions (or the full conditional posterior, for the threshold) presented several local maxima. In our experiments the global maximum was always clearly visible and much larger that the other local maxima. We tried the clusterings suggested by all the local maxima, but the best results were always achieved with the first or second mode. In the results below we marked with a “(2)” the results obtained with the second mode when this one yielded better results than the first mode.

6.1 Performance on real and simulated data

In this section we report the results of a simulation carried out to study the performance of the Conditional-Potts clustering model on three different artificially generated data sets and three real data sets. We report the best published ARI scores for the the real data sets below. These are the scores to be compared with the results given in Table 2. Based on these scores, the reader may get an idea of how difficult is to cluster some data sets into the groups selected by some experts (see for example the yeast cycle data below).

The artificial data sets were:

- A 5-clump-3-arc data set which consists of 340 points mixed in five Gaussian clumps and three arcs with
uniformly scattered points. These data were first used in (Murua et al., 2008) to study the performance of Potts model clustering. Besides their variation in shape and distribution, the clusters are not very well separated.

- The Bull’s eye data set. This was obtained as in (Blatt et al., 1997). It consists of three ring clusters generated by Gaussians \( N(R, 0.025) \). The radius of the rings, \( R \), are set to 1, 2 and 3, for the first, second and third cluster, respectively. The cluster sizes are 800, 1600, and 2400, respectively. These data are a real challenge for most clustering methods.

- A 50-component Gaussian mixture data set. The Gaussian means were generated uniformly on the square \([0, 500] \times [0, 500] \). The variances were set to 1.0 for the first 25 clusters, and to 9.0 for the others. The difference in cluster volume may produce difficulties when choosing the appropriate temperature-bandwidth parameters.

The data are plotted in Figure 1. The results are shown in Table 1.

| Algorithm   | 5-clump-3-arc (BM=0.29) | Bulls Eye (BM=0.20) | 50-Gaussians (BM=0.025) |
|-------------|--------------------------|----------------------|-------------------------|
|             | T | B | Q | ARI | T | B | Q | ARI | T | B | Q | ARI |
| Prior+T+Q   | 0.22 | AB | 0.37(2) | 0.59 | 0.30 | AB | 0.14 | 0.99 | 2.3 | AB | 0.11 | 0.98 |
| Prior+T+B+Q | 0.62 | 0.24 | 0.31(2) | 0.64 | 0.82 | 0.18 | 0.12 | 0.99 | 14.9 | 0.02 | 0.10 | 0.98 |
| T PMC2      | 0.27 | AB | 0.50 | 0.53 | 0.32 | AB | 0.50 | 0.02 | 2.2 | AB | 0.50 | 0.77 |
| T+Q PMC2    | 0.27 | AB | 0.13 | 0.68 | 0.32 | AB | 0.14 | 0.99 | 2.2 | AB | 0.11 | 0.98 |
| T IPMC2     | 0.26 | AB | 0.50 | 0.58 | 0.32 | AB | 0.50 | 0.18 | 3.2 | AB | 0.50 | 0.75 |
| T+Q IPMC2   | 0.26 | AB | 0.13 | 0.63 | 0.32 | AB | 0.14 | 0.99 | 3.2 | AB | 0.11 | 0.98 |
| T+B PMC2    | 0.49 | 0.28 | 0.50 | 0.56 | 0.54 | 0.13 | 0.50 | 0.14 | 0.7 | 0.014 | 0.50 | 0.98 |
| T+B+Q PMC2  | 0.49 | 0.28 | 0.14 | 0.88 | 0.54 | 0.13 | 0.11 | 0.99 | 0.7 | 0.014 | 0.11(2) | 0.98 |
| T+B IPMC2   | 0.71 | 0.30 | 0.50 | 0.43 | 0.81 | 0.22 | 0.50 | 0.02 | 8.8 | 0.024 | 0.50 | 0.73 |
| T+B+Q IPMC2 | 0.71 | 0.30 | 0.14 | 0.73 | 0.81 | 0.22 | 0.15 | 0.96 | 8.8 | 0.024 | 0.11(2) | 0.98 |

The three real data sets were:

- The well-known Iris data (Anderson,1935). It consists of 50 four-dimensional instances associated with each of three Iris types (Iris Setosa, Versicolor and Virginica). As is common in the machine learning literature, Iris versicolor and Iris virginica were considered as forming only one class. These data are very
simple to cluster. We have used them here because of its popularity with the statistical and machine learning community, and also to check that the procedures work.

- The Olive oil data set contains 572 samples of olive oil (Forina, 1987) described by their content in eight fatty acids. Nine different types of oil origins are represented, namely South-Apulia, North-Apulia, Calabria, Sicily, Inland-Sardinia, Coast-Sardinia, Umbria, East-Liguria and West-Liguria. The best results report an adjusted Rand index of about 0.70 (Stuetzle and Nugent 2010).
- The Yeast Cell Cycle data (Cho et al. 1998) record the fluctuations of the expression levels of about 6000 genes over two cell cycles comprising 17 time points. We use the 5-phase subset of the data (Cho et al. 1998). It consists of 420 genes of which 386 have been assigned to one of five phases of the cells cycle. These data are very difficult to cluster correctly. The best results so far published report an adjusted Rand Index of about 0.45 (Yeung et al. 2001, Murua et al. 2008).

The results are shown in Table 2.

7 Discussion

In general, all the procedures performed similarly on all six data sets. There does not appear to be a significant difference in performance between the PMC2 and IPMC2 algorithms. More strikingly, the results presented in Tables 1 and 2 clearly show that the quality of the clustering estimated by our informed conditional-Potts clustering procedure, Prior + PMC, is equal to those estimated by the more computationally intensive procedures PMC2 and IPMC2. The modes in our informative priors, elucidated
Table 2: Real Data Results. See Table 1 for an explanation of the notation.

| Algorithm     | Iris (BM=0.29) | Olive Oil (BM=0.07) | Yeast Cycle (BM=0.29) |
|---------------|----------------|---------------------|-----------------------|
|               | T   | B   | Q   | ARI | T   | B   | Q   | ARI | T   | B   | Q   | ARI |
| Prior+T+Q     | 0.22| 0.14| 1.0 | 1.23| 0.42| 0.61| 0.29| 0.46| 0.48| 0.29| 0.46| 0.48|
| Prior+T+B+Q   | 0.68| 0.28| 0.14| 1.00| 0.45| 0.83| 0.97| 0.30| 0.45| 0.97| 0.30| 0.45|
| T PMC2        | 0.28| 0.50| 0.24| 1.07| 0.50| 0.70| 0.34| 0.50| 0.28| 0.34| 0.50| 0.28|
| T+Q PMC2      | 0.28| 0.13| 1.00| 1.07| 0.45| 0.83| 0.34| 0.42| 0.38| 0.34| 0.42| 0.38|
| T IPMC2       | 0.31| 0.50| 0.22| 1.12| 0.50| 0.51| 0.34| 0.50| 0.33| 0.34| 0.50| 0.33|
| T+B IPMC2     | 0.31| 0.13| 1.00| 1.12| 0.45| 0.70| 0.34| 0.43| 0.42| 0.34| 0.43| 0.42|
| T+B+Q PMC2    | 0.86| 0.28| 0.50| 0.23| 2.86| 0.07| 0.40| 0.81| 0.73| 0.29| 0.50| 0.20|
| T+B IPMC2     | 0.86| 0.13| 0.87| 2.86| 0.07| 0.40| 0.81| 0.73| 0.29| 0.35| 0.45|
| T+B+Q IPMC2   | 0.73| 0.33| 0.50| 0.25| 3.42| 0.08| 0.13| 0.72| 1.46| 0.21| 0.13| 0.44|

from the application of random graph theory and nonparametric kernel density estimation, fall near the modes of the posterior densities of the temperature and bandwidth parameters, respectively. Figure 2 shows the corresponding prior and posterior densities for the Iris and Olive Oil data sets. In particular, the prior and posterior modes seem to fall in a stable region of the temperatures, i.e., a region where there is high agreement (large ARI) between the clusterings associated to nearby temperatures.

![Figure 2: The prior and posterior densities for the Iris (top panel) and Olive Oil (bottom panel) data sets. The posterior densities were estimated with the IPMC2 algorithm.](image)

One of the main conclusions we gather from these experiments is that if the goal of the analyst is only to uncover a good data clustering without the burden of computation, then the informed Conditional-Potts clustering procedure is a viable alternative to the fully Bayesian procedure estimated by PMC2 or IPMC2. Any of the two models presented, i.e., with the estimation of a fixed bandwidth or the use of an adaptive bandwidth, could be used. Another remarkable finding is that, seemingly, there is always a model with a fixed kernel bandwidth for which the clustering performance is similar to and sometimes better than the one yielded with an adaptive bandwidth. Moreover, the optimal fixed kernel bandwidth
is close to the commonly used mean-distance bandwidth (denoted by BM in Tables 1 and 2).

Another important finding is that besides the temperature, the threshold parameter used to get the consensus clustering is also a critical parameter of the procedure. The commonly used threshold of 0.5 is most of the time suboptimal. A sound way of choosing the threshold consists of setting its value to one of the modes of the full posterior conditional distribution of the probabilities \( \{Q_{ij}\} \). As Tables 1 and 2 show, the clustering results at the mode and 0.5 are noticeably different (see for example, the results for the Bulls Eye, Iris and Olive Oil data sets).

We note that, in contrast to the procedures suggested in (Blatt et al, 1996) and (Murua et al, 2008), the clustering procedure presented here is fully automated. Besides the data, there is no need to input any parameters. The procedure is also efficient since the final consensus clustering is built at only one (optimal) value of the temperature (and kernel bandwidth, if this parameter is part of the model).

Concerning the two stochastic algorithms PMC2 and IPMC2 presented here, we note that (7) (and (8)) simplifies the Monte Carlo problem to a simulation of a Potts model with fixed interaction. In addition, the IPMC2 version of the algorithm eliminates the need to simulate label samples from the Potts model with non-constant interactions. Noting that there exist perfect sampling algorithms for the Potts density (Propp and Wilson 1996), one could implement perfect-sampling versions of PMC2 and IPMC2.

From a computational point of view, we note that the PMC2 and IPMC2 algorithms could present numerical problems when computing the ratio \( Z_{m+1}(\sigma', T')/Z_{m+1}(\sigma, T) \). To achieve a good precision, we had to normalize each term in the sum associated with \( Z_{m+1}(\sigma, T) \) in equation (11) by \( Z_{m+1}(\sigma', T') \) in a clever way: let

\[
B(g, \sigma, T) = \left[ \sum_{\ell=1}^{m+1} \exp\left[-\sum_{i \sim j} (\beta(\sigma^g, T^g) - \beta(\sigma, T))(1 - \delta_{ij}^g)\right] \right]^{1/(m+1)}
\]

Then

\[
Z_{m+1}(\sigma', T')/Z_{m+1}(\sigma, T) = \sum_{g'=1}^{d} \sum_{g=1}^{d} \kappa(\sigma, T, \sigma^g, T^g) \kappa(\sigma', T', \sigma'^g, T'^g) e^{m+1(g) - c_{m+1}(g')} B(g, \sigma, T) B(g', \sigma', T')^{-1}
\]

This computation takes most of the computing time of the procedures. We have tried less computing intensive ways of calculating this ratio (e.g. by multiplying and dividing each term in the sum (11) by a quantity that could scaled all the terms), but (12) seems to give the best results so far. In order to speed up the algorithm, one could consider estimating only the ratio of normalizing constants \( Z(\sigma, T, X)/Z(\sigma, T) \) on the grid \( \{(\sigma^g, T^g)\}^d_{g=1} \). The Wang-Landau algorithm can do this in an efficient way.

Since, judging from the results in the previous section, it appears that the consensus clustering is not very sensitive to the actual values of the parameters \((\sigma, T)\), as long as these values lie in the stable region of the Potts density, one could estimate the consensus clustering at the parameter values

\[
(\sigma_G, T_G) = \arg \max \{[Z_{m^*}(\sigma^g, T^g, X)/Z_{m^*}(\sigma^g, T^g)] \times \pi(\sigma^g, T^g), g = 1, \ldots, d\}
\]

where \( m^* \) indicates the last iteration of the IPMC2 algorithm. We have done this for the Yeast Cycle, Olive Oil, and 5-clump-3-arc data sets. The clustering results are shown in Table 3. One can see that the
results are similar to the ones given by the IPMC2 algorithm. Therefore, only using the Wang-Landau estimates seems to be an alternative and efficient way to build the consensus clustering.

Table 3 also shows the results yielded had we used a flat prior (i.e., a uniform prior) on the temperature-bandwidth space rather than our data-driven prior. These latter results are easily obtained by considering the consensus clustering at the parameter values

\[ (\sigma^F_{G}, T^F_{G}) = \arg \max \{ [Z_{m^*}(\sigma^g, T^g, X)/Z_{m^*}(\sigma^g, T^g)]_g = 1, \ldots, d \}, \]

since the normalizing constants do not depend on the prior. One can observe that the results obtained with the flat prior are almost identical to those obtained with the informative prior. It appears that the informative prior does not influence much the clustering results, i.e., the Conditional-Potts clustering model does not seem too sensitive to the prior used. The exception is the adaptive bandwidth model applied to the 5-clump-3-arc data. The informative prior performs much better in this case. This suggests that, at least for clustering purposes, the use of the informative prior does not degrade the results, and instead it may lead to improved results.

The prior and estimated posterior densities for these data sets are depicted in Figure 3 for the adaptive bandwidth model. Figure 4 shows, for the fixed kernel bandwidth model (i.e., the model that includes \( \sigma \) as a parameter), the contour images of the \((\sigma, T)\)-prior and estimated posterior densities for the Olive Oil and Yeast Cycle data sets. The contour images as well as the density plots show that there seems to be a rather large stable region on the temperature-bandwidth (or temperature) space for which the consensus clustering results are very similar and nearly optimal. This has already been observed by Stanberry et al. (2008) on fMRI data, and has been suggested by Blatt et al. (1996) as a result of what is known about the Potts model in ferromagnetism. The main finding here is that there seems to exist a relation between the optimal \( T \) and the size of the threshold \( Q \): the larger the \( T \), the smaller the threshold \( Q \). This can be seen in Tables 1 and 2. See for example, the result for the the Olive Oil and Yeast Cycle data sets. However, at least empirically (and perhaps due to numerical limitations, i.e. \( T \) becomes too small or too large), we have observed that it is not true that for every threshold \( Q \) there is a good \( T \) for clustering; for example, there is not always a good temperature associated to \( Q = 0.5 \). This again, leads us to the conclusion that \( Q \) is a critical parameter.

A main consequence of these observations is that as long as we choose a set of parameters in the temperature-bandwidth stable region, the clustering results will not change by much. The search for an optimal clustering reduces to a search for this region and not to a point search in the parameter space. Procedures that serve to uncover the totality or parts of this region will perform as well as as procedures that search for an optimal point in the parameter space. This may explain why just estimating the posterior on a grid of the parameter space or just optimizing a very informative prior does well. As long as the grid covers parts of the stable region, or the informative prior peaks inside the stable region, the clustering obtained will be a good clustering. The informative prior based on random graph theory gives us a strong hint on the location of the stable region of the Potts model. Since we derived this information from the study of many random cluster models with constant interaction, it is reasonable to expect further
improvements in clustering with the advancement of the knowledge of the phase transition region of a random cluster model with variable interactions between the data points.

Table 3: Wang Landau Estimates of the Posterior. +Prior indicates that the results were obtained using the informative priors. +Flat indicates that the results were obtained using a flat (uniform) prior. See Table 1 for further explanation on the notation.

| Algorithm       | 5-clump-3-arc | Olive Oil | Yeast Cycle |
|-----------------|---------------|-----------|-------------|
| T+Prior         | 0.14 AB 0.50 0.73 | 1.76 AB 0.50 0.41 | 0.27 AB 0.50 0.37 |
| T+Q+Prior       | 0.14 AB 0.20(2) 0.78 | 1.76 AB 0.12 0.41 | 0.27 AB 0.48(2) 0.47 |
| T+Flat          | 0.03 AB 0.50 0.36 | 1.76 AB 0.50 0.41 | 0.20 AB 0.50 0.46 |
| T+Q+Flat        | 0.03 AB 0.98 0.59 | 1.76 AB 0.12 0.71 | 0.20 AB 0.50 0.46 |
| T+B+Prior       | 0.96 0.20 0.50 0.36 | 1.76 AB 0.50 0.41 | 0.20 AB 0.50 0.46 |
| T+B+Q+Prior     | 0.96 0.20 0.12(2) 0.83 | 5.67 AB 0.13(2) 0.70 | 1.29 AB 0.13(2) 0.45 |
| T+B+Flat        | 0.96 0.20 0.50 0.36 | 1.76 AB 0.50 0.41 | 0.20 AB 0.50 0.46 |
| T+B+Q+Flat      | 0.96 0.20 0.12(2) 0.83 | 5.67 AB 0.13(2) 0.70 | 1.29 AB 0.13(2) 0.45 |

A Appendix: Proof of the lemma

\[ P_G \] is such that
\[
P_G \leq \frac{1}{|G(n, r)|} \sum_{k=r+1}^{n/2} \binom{n}{k} \kappa(G(k, r)) |G(n - k, r)|,
\]
where \( \kappa(G(k, r)) \) stands for the number of spanning trees in \( r \)-regular graphs. Note that in order to have a component of order \( k \), \( k \) vertices need to be selected and connected by a spanning tree. Using the bound (Biggs, 1994, page 42)
\[
\kappa(G(k, r)) \leq \frac{1}{k} \left( \frac{kr}{k-1} \right)^{k-1},
\]
it follows that
\[
P_G \leq \frac{1}{|G(n, r)|} \sum_{k=r+1}^{n/2} \binom{n}{k} \frac{1}{k} \left( \frac{kr}{k-1} \right)^{k-1} |G(n - k, r)|.
\]
Using the asymptotic approximation of Corollary 2.17 in (Bollobás, 2001, page 55)
\[
|G(n, r)| \sim \sqrt{2} e^{-(r^2-1)/4} \left( \frac{r^{r/2}}{e^{r/2} r!} \right)^n n^{r/2} \text{ as } n \to +\infty,
\]
we get
\[
P_G \leq \sum_{k=r+1}^{n/2} \binom{n}{k} \frac{1}{k} \left( \frac{kr}{k-1} \right)^{k-1} \left( \frac{r^{r/2}}{e^{r/2} r!} \right)^n (n-k)^{r(n-k)/2} \left( \frac{r^{r/2}}{e^{r/2} r!} \right)^{-n} n^{-r/2}.
\]
Next, using the Stirling inequality, we obtain
\[
P_G \leq \sum_{k=r+1}^{n/2} \binom{n}{k} \frac{1}{k} \left( \frac{kr}{k-1} \right)^{k-1} \left( \frac{r^{r/2}}{e^{r/2} r!} \right)^n \left(1 - k/n\right)^{rn/2} \frac{1}{(n-k)^{r/2}}.
\]
Figure 3: The prior and posterior densities for the 5-clump-3-arc (left), Olive Oil (center) and Yeast Cycle (right) data sets. The posterior densities were estimated with the Wang-Landau discrete space sampling portion of the IPMC2 algorithm.
Figure 4: Contour images of the logarithm of the prior (left column) and the flat prior posterior (right column) for the Olive Oil (top panel) and Yeast Cycle (lower panel) data sets. The posterior densities were estimated with the Wang-Landau discrete space sampling portion of the IPMC2 algorithm.
\[ P_G \leq \sum_{k=r+1}^{n/2} \binom{n}{k} \frac{1}{k} \left( \frac{kr}{k-1} \right)^{k-1} r^{kr/2} e^{-kr} \frac{\sqrt{2 \pi r}}{(n-k)^{r/2}}, \text{ as } 1 - k/n < 1 \]

\[ \leq \sum_{k=r+1}^{n/2} \binom{n}{k} \frac{1}{k} \left( \frac{kr}{k-1} \right)^{k-1} r^{kr/2} \sqrt{2 \pi r} e^{-kr} \frac{1}{(n/2)^{r/2}}, \text{ as } k \leq n/2 \]

\[ \leq \sum_{k=r+1}^{n/2} \frac{n^{k-rk/2}}{k!} \frac{1}{k} \left( \frac{(r+1)r}{r} \right)^{k-1} r^{kr/2} \sqrt{2 \pi r} e^{-kr} \frac{1}{(n/2)^{r/2}}, \text{ as } \frac{k}{k-1} \text{ is a decreasing function of } k \]

\[ \leq \sum_{k=r+1}^{n/2} \frac{n^{k-rk/2}}{k!} \frac{1}{k} \left( \frac{(r+1)r}{r} \right)^{k-1} r^{kr/2} \sqrt{2 \pi r} e^{-kr} \frac{1}{(n/2)^{r/2}}, \text{ using the Stirling inequality} \]

\[ \leq \frac{1}{\sqrt{2 \pi}} \sum_{k=r+1}^{n/2} n^{k-rk/2} (r+1)^{-5/2} r^{(k-1)^2} \sqrt{2 \pi} e^{-k} e^{k(1-\frac{5}{2})+\frac{1}{r}/2} \]

\[ \leq \frac{1}{\sqrt{2 \pi} (r+1)^{5/2}} \sum_{k=r+1}^{n/2} s^k, \text{ with } s = n^{1-\frac{5}{2r}} r^{\frac{5}{2r}} \sqrt{2 \pi} e^{1-\frac{5}{2r}} \frac{1}{r}/2 \]

\[ \leq s^{r+1}/\sqrt{2 \pi} (r+1)^{5/2}(1-s). \]

References

[1] Abramson, I. S., “On bandwidth variation in kernel estimates - a square root law”, Ann. Statist., 10, 1217-1223 (1982).

[2] Agrawal, H. and Domany, E., “Potts ferromagnets on coexpressed gene networks: identifying maximally stable partitions,” Physical Review Letters, 90, 158102 (2003).

[3] Anderson, Edgar, “The irises of the Gaspé Peninsula,” Bulletin of the American Iris Society, 59: 2–5 (1935).

[4] Atchade, Y. F., and Liu, J. S., “The Wang-Landau algorithm for Monte Carlo computation in general state spaces,” To appear in Statistica Sinica, (2004).

[5] Atchade, Y. F., Lartillot, N., Robert, C. P., “Bayesian computation for statistical models with intractable normalizing constants,” http://arxiv.org/abs/0804.3152v1 (2008)

[6] Biggs, N., “Algebraic Graph Theory,” Cambridge Press (1994)

[7] Blatt, M., Wiseman, S. and Domany, E., “Super-paramagnetic clustering of data”, Physical Review Letters, 76, 3251–3255 (1996a).

[8] Blatt, M., Wiseman, S., and Domany, E., “Clustering data through an analogy to the Potts model,” Advances in Neural Information Processing Systems, 8, 416-422. Edited by D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, MIT Press (1996b).

[9] Blatt, M., Wiseman, S. and Domany, E., “Data clustering using a model granular magnet,” Neural Computation, 9, 1805-1842 (1997).

[10] Bollobas, B., “Random Graphs,” Academic Press (1985).

25
[11] Cho, R. J., Campbell, M. J., Winzeler, E. A., Steinmetz, L., Conway, A., Wodicka, L., Wolfsberg, T. G., Gabrielian, A. E., Landsman, D., Lockhart, D. J. and Davis, R. W., “A genome-wide transcriptional analysis of the mitotic cell cycle,” *Molecular Cell*, 2, 65–73 (1998).

[12] Cucala, L., Marin, J. M., Robert, C., and Titterington, D., “A Bayesian reassessment of nearest-neighbour classification,” *Journal of the American Statistical Association*, 104, 263-273 (2009).

[13] Dayal, P., Trebst, S., Wessel, S., Würtz, D., Troyer, M., Sabhapandit, S. and Coppersmith, S., “Performance Limitations of Flat Histogram Methods,” *Physical Review Letters*, 9, 097201 (1–4) (2004).

[14] Domany, E., Blatt, M., Gdalyahu, Y. and Weinshall D., “Superparamagnetic clustering of data: application to computer vision,” *Computer Physics Communications*, 121-122, 5-12 (1999).

[15] Domany, E., “Cluster analysis of gene expression data,” *Journal of Statistical Physics*, 110 (3-6), 1117 (2003).

[16] Edwards, R. G. and Sokal, A. D., “Generalization of the Fortuin-Kasteleyn-Swendsen-Wang representation and Monte Carlo algorithm,” *Physical Review D*, 38, 2009–2012 (1988).

[17] Einav, U., Tabach, Y., Getz, G., Yitzhaky, A., Ozbek, U., Amarglio, N., Izraeli, S., Rechavi, G. and Domany, E., “Gene expression analysis reveals a strong signature of an interferon-induced pathway in childhood lymphoblastic leukemia as well as in breast and ovarian cancer,” *Oncogene*, 24, 6367-6375 (2005).

[18] Forina M., Armanino, C., “Eigenvector projection and simplified non-linear mapping of fatty acid content of Italian olive oils,” *Annali di Chimica*, 72:127–141 (1982).

[19] Gelman, A., and Meng, X. L., “Simulating normalizing constants: from importance sampling to bridge sampling to path sampling,” *Statistica Sinica*, 13, 163–185 (1998).

[20] Getz, G., Levine, E., Domany, E. and Zhang, M., “Super-paramagnetic clustering of yeast gene expression profiles,” *Physica A*, 279 (1-4), 457-464 (2000).

[21] Ghulghazaryan, G., Hayryan, S., and Hu, C. K., “Efficient Combination of Wang-Landau and Transition Matrix Monte Carlo Methods for Protein Simulations,” *Journal of Computational Chemistry*, 28, 715–726 (2006).

[22] Girolami, M., “Mercer kernel based clustering in feature space,” *IEEE Transactions on Neural Networks*, 13, 669–688 (2002).

[23] Hubert, L. and Arabie, P., “Comparing Partitions,” *Journal of Classification*, 2, 193–218 (1985).

[24] Murua, A., Stanberry, L., and Stuetzle, W., “On Potts Model Clustering, Kernel K-Means and Density Estimation,” *Journal of Computational and Graphical Statistics*, 17(3), 629-658 (2008).

[25] Ogata, Y., “A Monte Carlo method for high dimensional integration” *Numerische Mathematik*, 55, 137–157 (1989).

[26] Ott, T., Kern, A., Schuffenhauer, A., Popov, M., Acklin, P., Jacoby, E. and Stoop, R., “Sequential superparamagnetic clustering for unbiased classification of high-dimensional chemical data,” *J. Chem. Inf. Comput. Sci.*, 44, 1358-1364 (2004).
[27] Ott, T., Kern, A., Steeb, W. and Stoop, R., “Sequential clustering: tracking down the most natural clusters,” *J. of Statistical Mechanics: theory and experiment*, 11, P11014 (2005).

[28] Propp, J. G., and Wilson, D. B., “Exact sampling with coupled Markov chains and applications to statistical mechanics,” *Random Structures & Algorithms*, 9, 223–252 (1996).

[29] Quiroga, R. Q., Nadasdy, Z., and Ben-Shaul, Y., “Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering,” *Neural Computation*, 16, 1661-1687 (2004).

[30] Rand, W. M., “Objective criteria for the evaluation of clustering methods,” *Journal of the American Statistical Association*, 66, 846–850 (1971).

[31] Reichardt, J., and Bornholdt, S., “Detecting Fuzzy Community Structures in Complex Networks with a Potts Model,” *Physical Review Letters*, 93, 218701 (1–4) (2004).

[32] Richardson, S., and Green, P. J., “On Bayesian analysis of mixtures with an unknown number of components,” *Journal of the Royal Statistics Society, Series B*, 59, 731–792 (1997).

[33] Shell, M. S., Debenedetti, P. G., and Panagiotopoulos, A. Z., “Generalization of the Wang-Landau method for off-lattice simulations,” *Physical Review E*, 66, 056703 (1–9) (2002).

[34] Silverman, B. W., “Density Estimation for Statistics and Data Analysis,” London, Chapman and Hall (1986).

[35] Sokal, A. D., “Monte Carlo methods in statistical mechanics: Foundations and new algorithms,” *Lectures at the Cargese Summer School on “Functional Integration: Basics and Applications”* (1996).

[36] Stanberry, L., Murua, A. and Cordes, D., “Functional connectivity mapping using the ferromagnetic Potts spin model,” *Human Brain mapping*, 29, 422-440 (2008).

[37] Stuetzle, W. and Nugent, R., “A generalized single linkage method for estimating the cluster tree of a density,” *Journal of Computational and Graphical Statistics*, 19, 397–418 (2010).

[38] Swendsen, R. H. and Wang, J. S., *Physical Review Letters*, 58, 86 (1987).

[39] Wang, S. and Swendsen, R. H., “Cluster Monte Carlo algorithms,” *Physica A*, 167, 565–579 (1990).

[40] Wang, F., and Landau, D. P., “Efficient, multiple-range random walk algorithm to calculate the density of states,” *Physical Review Letters*, 86, 2050–2053 (2001).

[41] Wiseman, S., Blatt, M., Domany, E., “Superparamagnetic clustering of data,” *Physical Review E*, 57, 3767-3783 (1998).

[42] Wolff, U., “Collective Monte Carlo updating for spin systems,” *Physical Review Letters*, 62, 361–364 (1989).

[43] Yeung, K. Y., Fraley, C., Murua, A., Raftery, A. E., and Ruzzo, W. L., “Model-based clustering and data transformations for gene expression data,” *Bioinformatics*, 17, 977-987 (2001).

[44] Zhou, C., Schulthess, C., Torbrügge, S., and Landau, D. P., “Wang-Landau Algorithm for Continuous Models and Joint Density of States,” *Physical Review Letters*, 96, 120201 (1–4) (2006).