Supplementary Material for 'MCMC computations for Bayesian mixture models using repulsive point processes'

Mario Beraha[*]
Department of Mathematics, Politecnico di Milano
Department of Computer Science, Università di Bologna
and
Raffaele Argiento
Department of Statistical Sciences, Università Cattolica del Sacro Cuore, Milano
and
Jesper Møller
Department of Mathematical Sciences, Aalborg University
and
Alessandra Guglielmi
Department of Mathematics, Politecnico di Milano

July 2, 2021

In the following, all equation and section numbers refer to the main article.

S1 Further details on the Metropolis-within-Gibbs sampler used for posterior simulation

This section provides additional details for the Metropolis-within-Gibbs (M-w-G) sampler in Section 5.

S1.1 The choice of the proposal distribution

For most choices of the point process density $p(\boldsymbol{\mu} \mid \xi)$ and the mixture kernel $k(\cdot \mid \cdot)$, the update of the allocated means $\mu_h^{(a)}$ requires sampling from an unnormalized distribution, which we do via a Metropolis-Hastings step. As proposal distribution we use a mixture of two normal distributions with means equal to the current value of $\mu_h^{(a)}$ but with different variances so that

$$p(\mu'; \mu_h^{(a)}) = \kappa \mathcal{N}(\mu' \mid \mu_h^{(a)}, \underline{\sigma}^2 I) + (1 - \kappa) \mathcal{N}(\mu' \mid \mu_h^{(a)}, \bar{\sigma}^2 I), \tag{S1}$$

---

[*] Jesper Møller was supported by the 'Danish Council for Independent Research – Natural Sciences' grant DFF – 7014-00074 'Statistics for point processes in space and beyond'

where $\kappa = 0.9$, $\underline{\sigma} = 0.1$, and $\overline{\sigma} = 1.5$ when $q = 1, 2$ and $\overline{\sigma} = 1.5q$ when $q > 2$. The intuition that led us to consider such a proposal is as follows, where for ease of notation we drop the superscript (a) when considering a current value of $\mu_h^{(a)}$, denoted μ_1, and another cluster centre μ_2. Suppose that μ_1 and μ_2 are close and far from the remaining points in $\boldsymbol{\mu}$. If the number of observations allocated to μ_1 is small, we want a proposal distribution $p(\mu_1'; \mu_1)$ that gives significant mass to values that are far from μ_2, so that, given the repulsiveness of the point process, this proposal is likely to be accepted. This is the case when we sample from the second component of (S1) (in fact, if μ_1' is far from μ_1, with sufficiently large probability it is far from μ_2 as well). On the other hand, if the number of observations allocated to μ_1 is large, we want a proposal that gives significant mass to a neighborhood of of the current value of μ_1, to get a precise fit of the data. This is what happens if we sample from the first component of (S1).

For the second component in (S1), instead of fixing $\overline{\sigma}$ as we do, an alternative is to exploit the properties of $g(\cdot \mid \xi)$ as follows. Suppose we condition on sampling from $\mathcal{N}(\mu_1' \mid \mu_1, \overline{\sigma}^2 I)$ in (S1). Then $\|\mu_1' - \mu_1\|^2/\overline{\sigma}^2 \sim \chi^2(q)$, the chi-squared distribution with q degrees of freedom. Considering the Strauss density, a possibility is to fix $\overline{\sigma}$ to give sufficiently high mass to values of μ_1' that are outside the range of interaction of μ_1, i.e., such that $P(\|\mu_1' - \mu_1\|^2 > \delta) > p_0$ for some fixed p_0, with the intuition that this gives a positive probability to μ_1' being distant at least δ also from μ_2. Considering the DPP density instead, the same argument holds but replacing δ with the range of correlation r_0, cf. Lavancier et al. (2015). That is, (12) implies that C is of the form $C(\mu_1, \mu_2) = C_0(r)$ with $r = \|\mu_1 - \mu_2\|$, and defining the corresponding correlation function $R(r) = C_0(r)/C_0(0)$, r_0 is chosen such that $R(r)$ is effectively zero.

S1.2 The exchange algorithm and perfect simulation

With the same notation as Section 5.3, the exchange algorithm (Murray et al., 2006) consists of the following steps:

1. Propose $\xi' \sim p(\xi'; \xi)$.
2. Generate an auxiliary variable $\boldsymbol{\mu}^{\text{aux}} \sim g(\boldsymbol{\mu} \mid \xi')/Z_{\xi'} \propto g(\boldsymbol{\mu} \mid \xi')$.
3. Accept ξ' with probability $\min\{1, \alpha^*\}$ where

$$\alpha^* \equiv \alpha^*(\xi; \xi' \mid \cdots) = \frac{p(\xi')g(\boldsymbol{\mu} \mid \xi')p(\xi; \xi')}{p(\xi)g(\boldsymbol{\mu} \mid \xi)p(\xi'; \xi)} \times \frac{g(\boldsymbol{\mu}^{\text{aux}} \mid \xi)}{g(\boldsymbol{\mu}^{\text{aux}} \mid \xi')}.$$

Comparing α^* to the acceptance ratio in (22), note that the ratio $Z_\xi/Z_{\xi'}$ has been replaced by a ratio of unnormalized densities, evaluated in the auxiliary variable $\boldsymbol{\mu}^{\text{aux}}$. The main difficulty is sampling $\boldsymbol{\mu}^{\text{aux}}$, which must follow the distribution of $\boldsymbol{\mu}$ given ξ'. To this end, we employ the stochastic dominated coupling from the past algorithm in Kendall and Møller (2000), which extends the coupling from the past algorithm in Propp and Wilson (1996) to uncountable partially ordered spaces. Specifically, we employed in our code Algorithm 11.7 in Møller and Waagepetersen (2004).

S2 Additional simulation studies

In addition to the simulation studies in the main article, below we discuss different aspects of the M-w-G sampler and posterior inference.

S2.1 Comparison of run-times and posterior inference when using DPP and Strauss process priors

For $q = 1, 2, \ldots, 5$, we simulated $n = 200$ observations from (24) with $\mu_0 = (-5, \ldots, -5)$, $\Sigma_0 = I_q$, $\omega = 1$, $\mu_1 = 5$, and $\sigma_1 = 1$. Then we applied our M-w-G sampler when the marginal prior for $\boldsymbol{\mu}$ is either the DPP or the Strauss process, with hyperparameters as in Section 6. Here, we considered two truncation levels for the approximation of the DPP density in (12), namely $N = 5$ and $N = 10$ (for comparison, Bianchini et al. (2020) suggested $N = 50$ when $q = 1$).

Figure S1 shows the per-iteration run-times of the M-w-G sampler as a function of the dimension q under either the DPP or Strauss process prior for $\boldsymbol{\mu}$. For each value of N, the computational cost associated to the DPP grows exponentially fast as the dimension q increases, unlike in the case of the Strauss process. In fact, the unnormalized density of the Strauss process is almost immediate to compute, and since the Strauss prior is quite informative on the number of components, cf. Section 6.1, the perfect simulation algorithm (see Section 5.3) does not impact significantly on the computational cost. Although not appreciable from Figure S1, the computational cost of our algorithm increases significantly with data dimension q also when we consider the Strauss process; in this case, the per-iteration computational cost goes from 0.0016 sec when $q = 1$ to 0.07 sec when $q = 5$, i.e., it increases by a factor of roughly 50.

As a further comparison, we simulated 500 univariate observations from model (23) and made again posterior computations under the Strauss process or the DPP prior for $\boldsymbol{\mu}$, where for the DPP density we fixed $\beta = 10$ (corresponding to the highest ESS in Table 1). For both cases of prior models, we ran the M-w-G sampler for $100,000$ iterations discarding the first $50,000$ as a burn-in and keeping one every ten iterations, for a final sample size of $5,000$. Figure S2 shows the true data generating density, together with Bayesian mixture density estimates and posterior distributions of the number of clusters under the two point process priors. Note that the two density estimates, as well as the two posterior distributions of the number of clusters, overlap almost perfectly. The Strauss process seems a good choice to model the prior of $\boldsymbol{\mu}$ since it, for a

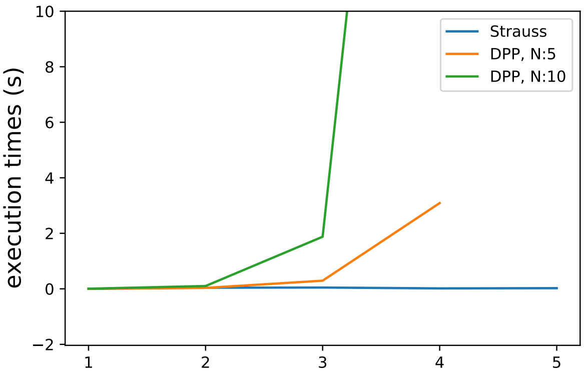


Figure S1: Per-iteration run-times as a function of data dimension q in case of DPP (with truncation levels $N = 5$ or 10) and Strauss process priors for $\boldsymbol{\mu}$.

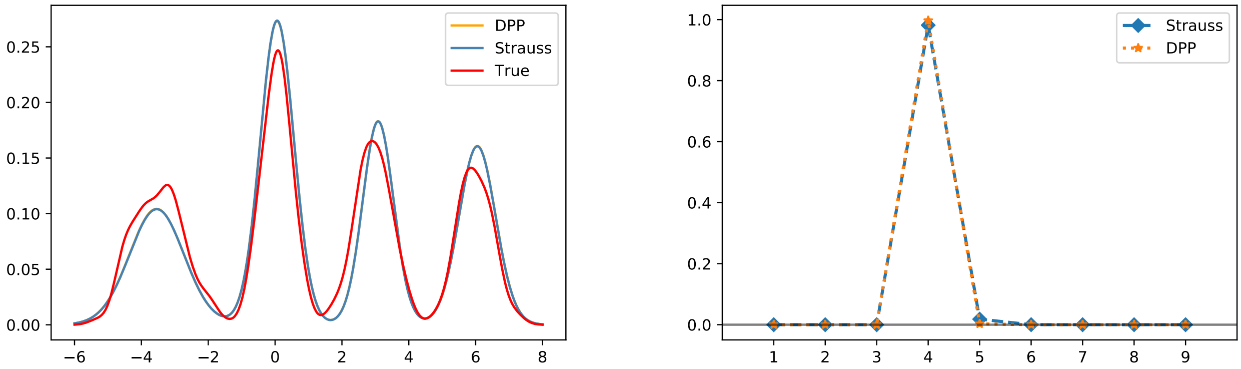
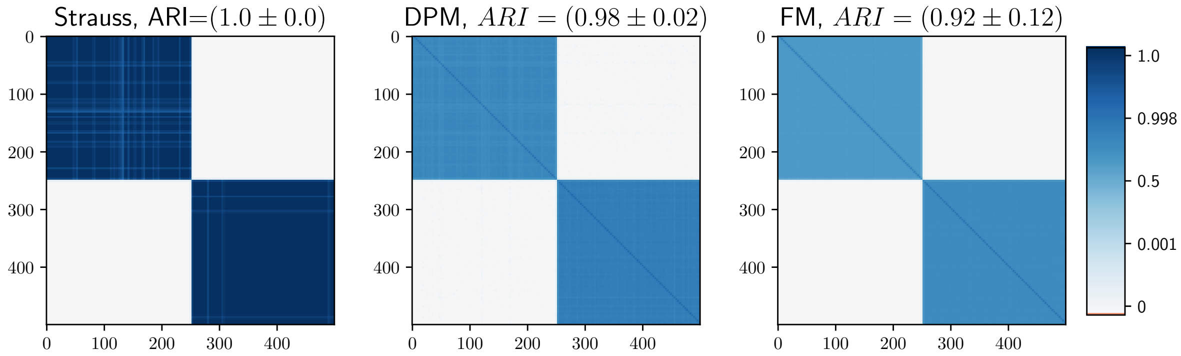


Figure S2: Bayesian mixture density estimates (left) and posterior distributions of the number of clusters (right) under the Strauss process (blue lines) and DPP (orange lines) priors for $\boldsymbol{\mu}$, together with the true mixture density which has four components. The orange lines overlap almost perfectly with the blue lines so that they are hardly visible.

much smaller computational cost, provides same posterior summaries as the DPP.

S2.2 Accuracy of cluster estimates

Figure S3 shows the posterior similarity matrices and the Adjusted Rand Index (ARI) scores for the univariate mixture of t and skew-normal distribution discussed in Section 7.2. The ARI is computed from the cluster labels $\boldsymbol{c}$ at each iteration of the MCMC chain as a measure of similarity between the estimated clusters and the true cluster. It is bounded by 1 and the larger value it assumes, the more similar is the estimated cluster to the true one. We report the posterior mean of the ARI ± one standard deviation on top of each posterior similarity matrix in Figure S3. The difference in the posterior similarity matrices is not so pronounced, but our repulsive mixture model gives the best ARI.



Strauss, ARI=(1.0 ± 0.0) DPM, $ARI = (0.98 \pm 0.02)$ FM, $ARI = (0.92 \pm 0.12)$

Figure S3: Posterior similarity matrices and ARI scores under the three models for the mixture of the univariate t and skew-normal distributions discussed in Section 7.2. The colors are on a logit scale to highlight differences around one.

S2.3 The effect of the number of clusters

We consider how the number of clusters affects the performance of our M-w-G sampler. When $\boldsymbol{\mu}$ is distributed as the Strauss process, at every step of the MCMC algorithm a perfect simulation of $\boldsymbol{\mu}$ is required. The perfect simulation algorithm we use has a finite but random computational cost and, as argued in Section 5.3, it might become infeasible for a large number of clusters. On the other hand, when $\boldsymbol{\mu}$ is a DPP, the approximation of its density requires computing the determinant of the matrix C' in (10), which scales cubically with m. Furthermore, for the specific DPP considered in (12) computing C' requires the evaluation of $O(N^q m^2)$ inner products.

We generated $n = 500$ observations from a mixture of $m = 5, 9, 17, 25$ bivariate Gaussian densities, with locations given in Figure S4 (left), equal covariance matrices given by $0.5I_2$, and with equal mixture weights. We compared the run-times (in seconds) required to complete 200 iterations with our M-w-G sampler when $\boldsymbol{\mu}$ is distributed either as the Strauss or the determinantal point process. Prior hyperparameters are fixed as in Section 6 (with $M_{\text{max}} = 5m$) and Section 7. For the DPP, we considered two truncation levels of the spectral density, $N = 10, 50$. For each choice of m we generated 50 independent datasets and for the 200 M-w-G sampler iterations we used fixed and different independent random seeds.

In Figure S4 (right) for each m the run-times over the 50 independent datasets are denoted by dots, the median times by diamonds, and the median times are connected by a dashed line. We see that the DPP with $N = 50$ is the most computationally demanding model for all values of m. When $m = 5, 9$, the Strauss process is significantly faster (up to 10 times faster) than the DPP with $N = 10$; instead, when $m = 17$, they have comparable computational costs. When $m = 25$, the perfect simulation algorithm starts to become more demanding; for example, the computational cost for the Strauss process is almost twice the one for the DPP with $N = 10$.

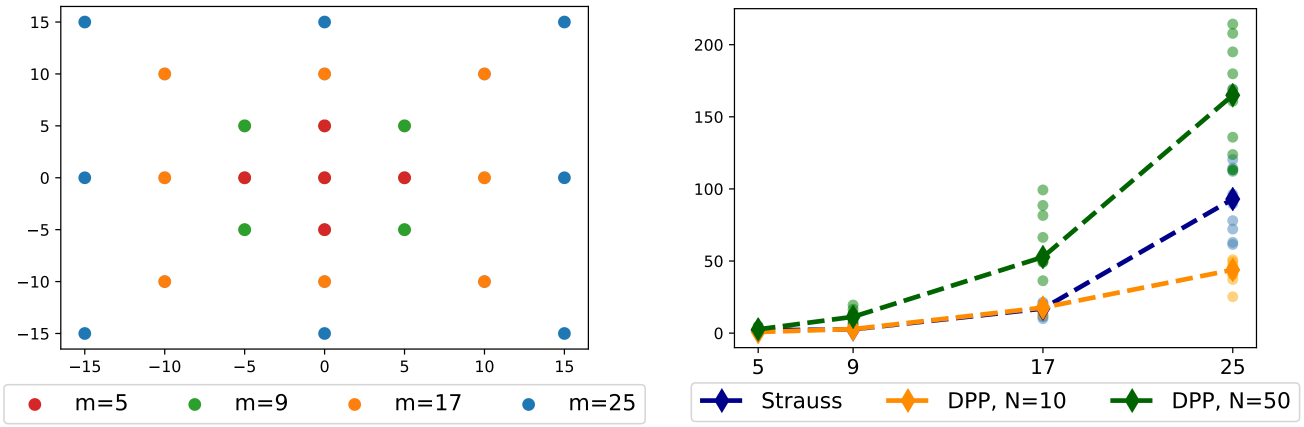


Figure S4: Locations of the true data generating process (left) and run-time comparison (right). The plot of the locations should be intended as follows: for $m = 5$ only the points labelled accordingly are considered, for $m = 9$ the points labeled as $m = 5$ and $m = 9$ are considered and so on. The run-times (in seconds) over 50 independently simulated datasets for each value of m are denoted by dots, we also report the median times as diamonds with a dashed line connecting them.

S2.4 The effect of the data dimension

Below we compare our repulsive mixture model, the finite mixture model (FM) in Argiento and De Iorio (2019), and the Dirichlet process mixture model (DPM). See Section 7.2 for further details on how posterior inference is performed under the different models. In particular, we fix the hyper-parameters according to Sections 6 and 7.

If the sample size n is not significantly larger than the data dimension q, the use of commonly employed MCMC algorithms may be problematic for the following reasons. If $k(\cdot \mid \cdot)$ is a multivariate Gaussian density with non-zero correlations, the number of parameters to be estimated is much larger than n. Further, the curse of dimensionality, common to all clustering problems (Kriegel et al., 2009), implies a poor mixing of the algorithms. In addition to that, when considering a repulsive mixture model, things might be further complicated by either the need of perfect simulation to update possible hyperparameters ξ (when $\boldsymbol{\mu}$ follows the Strauss process) or the computation of the spectral density (when $\boldsymbol{\mu}$ follows the DPP given by (12)) which becomes prohibitive even for moderate values of q, as shown in Figure S1. Therefore, below we consider only the Strauss process and perform a simulation to assess the performance of repulsive versus non-repulsive mixtures when $q = 2, 5, 10, 15, 20, 25, 30$ increases. Moreover, we simulated $n = 200$ observations from

$$y_i \overset{\text{iid}}{\sim} 0.5\mathcal{N}(-5/\sqrt{q}\mathbf{1}_q, I_q) + 0.5\mathcal{N}(5/\sqrt{q}\mathbf{1}_q, I_q)$$

where $\mathbf{1}_q$ denotes the vector in $\mathbb{R}^q$ with elements all equal to one.

Table 1 reports posterior summaries as q increases for the three models. MCMC chains were

		$q=5$	$q=10$	$q=15$	$q=20$	$q=25$	$q=30$
Strauss	ARI	1.0	1.0	1.0	1.0	1.0	1.0
	ESS	240.3	250.1	0.0	0.0	0.0	0.0
	$\mathbb{E}[k \mid \text{data}]$	2.01	2.005	2.0	2.0	2.0	2.0
FM	ARI	1.0	1.0	1.0	1.0	1.0	1.0
	ESS	7.4	0.0	0.0	0.0	0.0	0.0
	$\mathbb{E}[k \mid \text{data}]$	2.01	2.0	2.0	2.0	2.0	2.0
DPM	ARI	1.0	0.0	0.0	0.0	0.0	0.0
	ESS	0.0	0.0	0.0	0.0	0.0	0.0
	$\mathbb{E}[k \mid \text{data}]$	2.00	1.0	1.0	1.0	1.0	1.0

Table 1: Adjusted Rand Index (ARI), effective sample size for the chain of the number of clusters k and posterior mean of k under the repulsive mixture model (Strauss), the non repulsive finite mixture model (FM) and the Dirichlet process mixture model (DPM).

run for $11,000$ iterations discarding the first $10,000$ as burn-in, so that the effective sample size must be referred to a total number of MCMC iterations equal to $1,000$. It is clear from Table 1 that as q increases, the mixing of the chains becomes progressively worse for all the models. In particular, the table shows the effective sample size (ESS) for the three cases: For our repulsive mixture model, the number of clusters k is constant for all the MCMC iterations when $q \geq 15$, and so ESS is zero; for FM, the ESS is zero when $q \geq 10$; and for DPM the ESS is zero for all values of q. The difference in the ARI scores is simply explained by the different strategy of initialization of the different software we ran: In our code for the M-w-G sampler, observations are initially randomly subdivided into 10 clusters; in the package `AntMAN`, which we used to fit the FM model, one cluster per observation is created; in the package `BNPMix`, used to fit the DPM model, all observations are initially allocated to one single cluster. In the latter case, the proposal of a new cluster is never accepted. Using our software or the package `AntMAN` instead, after a few MCMC iterations the observations are (correctly) partitioned into $k = 2$ clusters and no additional cluster is ever created.

Considering the effective sample size of k, Table 1 shows that repulsive mixture models might offer an advantage over non-repulsive mixture models when $q \leq 10$. We believe that the poor performance of FM and DPM is due to prior assumptions for the following reason. Note that both models assume that the parameters $\{(\mu_h, \gamma_h)\}_h$ are a priori iid and normal inverse-Wishart distributed, with $\mathbb{E}[\mu_h] = 0$. Thus, as q increases, the multivariate Gaussian distribution becomes more and more concentrated around the mean, due to the so-called curse of dimensionality, so that proposing a new value for μ_h from the prior that is near to any of the observations becomes less likely. Instead, when considering a Strauss point process as prior for $\boldsymbol{\mu}$, the proposed means are not concentrated around the origin, which led to a better mixing when $q = 5, 10$. When $q \geq 15$, we believe that the volume of the rectangle containing all observations becomes so large that also the repulsive mixture models suffer from the curse of dimensionality.

Perfect simulation is not a bottleneck here, as the number of points in the Strauss process is small. However, in one of several independent simulations, an unlucky initialization led to a large value of m in the first few iterations. As a consequence, the perfect simulation algorithm took longer to coalesce and indeed caused an out-of-memory problem on a 32 GB laptop.

Finally, when $q \to +\infty$, Chandra et al. (2020) show how the posterior distribution under non repulsive mixture models either assigns all the observations to the same cluster or each observation to a separate cluster. These authors propose to consider mixtures in a latent space to overcome such issue, similarly to Ghahramani and Hinton (1996). Extensions of latent mixture models to account for repulsiveness are currently being investigated; see Ghilotti (2021).

S3 Removing the rectangular support assumption

Often we have assumed that the points of $\boldsymbol{\mu}$ have support given by a rectangular set R: For the theory in Sections 2–5, we made that assumption only for specificity and simplicity; in Section 7, we considered Gaussian mixture models and determined the rectangle R from the observations; while in Section 8, we considered the multivariate Bernoulli kernel and $R = [0, 1]^q$. Apart from the case of a DPP prior, it is often easy to modify everything without assuming R is rectangular and even compactness of R may be not be needed, In fact, the birth-death Metropolis-Hastings algorithm, which we always use to simulate the non-allocated process $\boldsymbol{\mu}^{(na)}$, can be specified in a very general setting, see Geyer and Møller (1994). On the other hand, for a DPP prior, compactness of R is needed when specifying a DPP density with respect to $\mathrm{d}\boldsymbol{\mu}$, and R needs to be a rectangle in order to use the spectral approach discussed in Lavancier et al. (2015). Recently, Poinas and Lavancier (2021) proposed a novel approximation of a general DPP density that does not require R to be rectangular (but still requires R is bounded).

References

Argiento, R. and De Iorio, M. (2019), “Is infinity that far? A Bayesian nonparametric perspective of finite mixture models,” Technical report, available at arXiv:1904.09733.

Bianchini, I., Guglielmi, A., and Quintana, F. A. (2020), “Determinantal point process mixtures via spectral density approach,” *Bayesian Analysis*, 15, 187–214.

Chandra, N. K., Canale, A., and Dunson, D. B. (2020), “Escaping the curse of dimensionality in Bayesian model based clustering,” Technical report, available at arXiv:2006.02700.

Geyer, C. J. and Møller, J. (1994), “Simulation procedures and likelihood inference for spatial point processes,” *Scandinavian Journal of Statistics*, 21, 359–373.

Ghahramani, Z. and Hinton, G. E. (1996), “The EM algorithm for mixtures of factor analyzers,” Technical report, CRG-TR-96-1, University of Toronto.

Ghilotti, L. (2021), *Bayesian clustering of high-dimensional data via latent repulsive mixtures*, Master’s thesis, Politecnico di Milano. Available upon request.

Kendall, W. S. and Møller, J. (2000), “Perfect simulation using dominating processes on ordered spaces, with application to locally stable point processes,” *Advances in Applied Probability*, 32, 844–865.

Kriegel, H.-P., Kröger, P., and Zimek, A. (2009), “Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering,” *ACM transactions on knowledge discovery from data*, 3, 1–58.

Lavancier, F., Møller, J., and Rubak, E. (2015), “Determinantal point process models and statistical inference,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 77, 853–877.

Møller, J. and Waagepetersen, R. P. (2004), *Statistical Inference and Simulation for Spatial Point Processes*, Chapman and Hall/CRC, Boca Raton.

Murray, I., Ghahramani, Z., and MacKay, D. J. C. (2006), “MCMC for doubly-intractable distributions,” in *Proceedings of the Twenty-Second Conference on Uncertainty in Artificial Intelligence*, UAI’06, Arlington, Virginia, USA: AUAI Press.

Poinas, A. and Lavancier, F. (2021), “Asymptotic approximation of the likelihood of stationary determinantal point processes,” Technical report, available at arXiv:2103.02310.

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