A regularized-entropy estimator to enhance cluster interpretability in Bayesian nonparametrics

Uno stimatore a entropia regolarizzata per migliorare l'interpretabilità dei cluster in bayesiana nonparametrica

Beatrice Franzolini, Giovanni Rebaudo

Abstract Bayesian nonparametric mixture models are widely used to cluster observations. However, one of the major drawbacks of the approach is that the estimated partition often presents only a few dominating clusters and a large number of sparsely-populated ones. This feature translates into results that are uninterpretable unless we accept to ignore a relevant number of observations and clusters. Here, we explain this phenomenon through the study of the cost functions involved in the estimation of the partition. Moreover, we propose a post-processing procedure to reduce the number of sparsely-populated clusters. The procedure takes the form of entropy-regularization of posterior cluster allocations. While being computationally convenient with respect to alternative strategies, it is also theoretically justified as a correction to the Bayesian loss function used for point estimation and, as such, can be applied to any posterior distribution of clusters, regardless of the specific Bayesian model used.

Abstract *I modelli Bayesiani nonparametrici con misture sono ampiamente utilizzati per effettuare cluster analysis. Tuttavia, uno dei principali limiti è il fatto che spesso identifichino un ampio numero di cluster poco popolati. Questa caratteristica si traduce in risultati di difficile interpretazione a meno che non si accetti di ignorare un numero di osservazioni e cluster. In questo lavoro, spieghiamo questo fenomeno attraverso lo studio delle funzioni di costo coinvolte nella stima della partizione. Inoltre, proponiamo una procedura di post-processing volta a ridurre il numero di cluster scarsamente popolati. La procedura prende la forma di una regolarizzazione dell'entropia dell'allocazione in cluster. La proposta appare computazionalmente conveniente rispetto a strategie alternative e trova giusticazione teorica in quanto correzione della funzione di perdita bayesiana impiegata nella stima puntuale, e, proprio per questa ragione, può essere adottata a prescindere dallo specifico modello utilizzato.*

Key words: Bayesian nonparametrics, Exchangeable partition probability function, Entropy, Clustering, Dirichlet process mixture

Beatrice Franzolini
Agency for Science, Technology and Research, Singapore, e-mail: franzolini@pm.me
Giovanni Rebaudo
Department of Statistics and Data Sciences, the University of Texas at Austin, USA,
e-mail: rebaudo.giovanni@gmail.com

1 Introduction

Clustering methods are used to detect patterns by partitioning observations into different groups. What are desirable characteristics of clusters depends on the specific applied problem at hand (see e.g., Hennig, 2015). Nonetheless, clustering methods are typically motivated by the idea that observations are more similar within the same cluster than across clusters (accordingly to a certain definition of similarity).

Clustering has been proved useful in a large variety of fields including but not limited to image processing, bio-medicine, marketing, and natural language processing. Clustering methods are used not only to detect sub-groups of subjects, but also for dimensionality reduction (Blei et al., 2003; Petrone et al., 2009), outlier-detection (Shotwell and Slate, 2011; Ngan et al., 2015; Franzolini et al., 2022), and data pre-processing (Zhang et al., 2006). Among clustering techniques, we can distinguish two main classes: model-based and non model-based.

Contrary to other popular clustering techniques, as k-means, model-based clustering methods allow us to perform inference via rigorous probabilistic assessments. Typically, model-based clustering frameworks are equivalent to the assumption that the observations $y_1,\dots y_n$ are extracted from an infinite population following a mixture

$$y_i \overset{iid}{\sim} \sum_{h=1}^{K} w_h k(\cdot;\theta_h) \qquad \text{i = 1,..., n}, \tag{1}$$

where the mixture components $k(\cdot;\theta_h)$ are probability kernels to be interpreted as distributions of distinct clusters in the infinite population, $(w_h,\,\theta_h)_{h=1}^K$ are unknown parameters that determine the relative proportion and the shape of such population clusters, and K is the total number of clusters in the population. K can be either a fixed value or an unknown parameter. However, the main goal of clustering techniques is to estimate a partition of the observed sample, more than the distribution of the whole ideal population in (1). The partition that one wants to estimate can be encoded using a sequence of subject-specific labels $(c_1,\dots,c_n)$ taking value in the set of natural numbers such that $c_i = c_j = c$ if and only if y_i and y_j belong to the same cluster and follow the same mixture component $k(\cdot;\theta_c)$, i.e. $y_i \mid c_i \overset{ind}{\sim} k(\cdot;\theta_{c_i})$ for $i = 1,\dots,n$. The indicators $(c_1,\dots,c_n)$, as just defined, are affected by the label switching problem (see, for instance, Stephens, 2000; McLachlan et al., 2019; Gil-Leyva et al., 2020). To overcome the issue, in the following, we assume them to be encoded in order of appearance. The likelihood for $\boldsymbol{c} = (c_1,\dots,c_n)$ and $\boldsymbol{\theta} = (\theta_1,\dots,\theta_{K_n})$ is

$$\mathscr{L}(\boldsymbol{c},\boldsymbol{\theta};\boldsymbol{y}) = \prod_{c=1}^{K_n}\prod_{i:c_i=c} k(y_i;\theta_c). \tag{2}$$

An important and typically unknown parameter is the number of clusters K_n observed in the sample, i.e., the number of occupied components. Obviously, $K_n \leq K$. For this reason, when we let n vary, finite fixed values for K are usually to be avoided and K is either fixed to $+\infty$ (e.g. in Dirichlet process mixtures, Ferguson, 1983; Lo,

1984) or it is estimated from the data (e.g. mixtures of finite mixtures, see Miller and Harrison, 2018; Argiento and De Iorio, 2019).

When K_n is unknown, the clustering labels in (2) cannot be estimated with a standard frequentist approach. In fact, when the maximum likelihood estimator (MLE) for (2) exists, it coincides with the vector of MLEs $(\hat{\theta}_1,\dots,\hat{\theta}_n)$, where each $\hat{\theta}_i$ is obtained considering one observation at a time and the independent models $y_i \sim k(y_i \mid \theta_i)$, for $i = 1,\dots,n$. Moreover, note that under typical mixture model assumptions for clustering, we have that $\hat{\theta}_1 \neq \ldots \neq \hat{\theta}_n$. For instance, when k is a multivariate Gaussian density and θ is the pair of mean vector and variance matrix of the Gaussian component, the MLE entails a number of clusters equal to the number of distinct observed values, that by model's assumptions equals n with probability 1. Thus, no information on clusters can ever be gained through MLE and overfitting is unavoidable unless one relies on strong restrictions of the parameter space. In this regard, note that maximizing (2) is not the same as computing the nonparametric maximum likelihood estimator (Lindsay, 1995; Polyanskiy and Wu, 2020; Saha and Guntuboyina, 2020) for the mixture model in (1).

Differently, Bayesian models, and in particular Bayesian nonparametric (BNP) models, are largely used for model-based clustering, since priors act as penalties shrinking the number of distinct clusters.

The content of the paper is organized as follows. Section 2 presents the study of the cost functions involved in BNP clustering models and explains a common drawback, i.e., the presence of noisy and sparsely populated clusters typically observed in the posterior estimates of these models. Then, a computationally convenient and theoretically justified solution to reduce the number of sparsely populated clusters is presented in Section 3 and showcased on simulated and real data, respectively in Sections 4 and 5.

2 Implied costs functions in Bayesian nonparametric clustering

The vast majority of Bayesian models for clustering rely on a prior for $\boldsymbol{c}$ and K_n defined through an exchangeable partition probability function (EPPF) (see, Pitman, 1996) and, independently, a prior P is used for the unique values $(\theta_1,\dots,\theta_{K_n})$. Therefore, the corresponding posterior distribution is

$$p(K_n,\boldsymbol{c},\boldsymbol{\theta} \mid \boldsymbol{y}) \propto \prod_{c=1}^{K_n} \prod_{i:c_i=c} k(y_i;\theta_c) \times \mathrm{EPPF}(n_1,\dots,n_{K_n}) \times P(d\boldsymbol{\theta}), \tag{3}$$

which can be equivalently represented as the cost function $-\log(p(K_n,\boldsymbol{c},\boldsymbol{\theta} \mid \boldsymbol{y}))$, i.e.

$$C(K_n,\boldsymbol{c},\boldsymbol{\theta};\boldsymbol{y}) = C_{\text{lik}}(K_n,\boldsymbol{c},\boldsymbol{\theta};\boldsymbol{y}) + C_{\text{part}}(K_n,\boldsymbol{c};\alpha) + C_{\text{base}}(K_n,\boldsymbol{\theta}),$$

which is the sum of three terms, that in the following are named respectively likelihood cost, partition cost, and base cost.

As already mentioned, the minimum of the likelihood cost

$$C_{\text{lik}}(K_n, \boldsymbol{c}, \boldsymbol{\theta}; \boldsymbol{y}) = -\sum_{c=1}^{K_n} \sum_{i:c_i=c}^{n} \log k(y_i; \theta_c)$$

typically corresponds to K_n equal to the number of distinct observed values. The remaining two costs are those defined by the prior of the model and their marginal behavior is described here below. Clearly, any inference result has to be derived based on the whole posterior distribution in (3), which is the result of the joint, and not marginal, effect of all three costs. Nonetheless considering one cost at a time allows us to gain insights regarding the estimation procedure and the frequentist penalties induced by the prior. A lot of attention in the literature has been devoted to the choice of the EPPF and many alternatives are available (see, for example, Lijoi et al., 2007; Lijoi and Prünster, 2010; De Blasi et al., 2013; Greve et al., 2022), while, except for few cases (Petralia et al., 2012; Xu et al., 2016; Beraha et al., 2021), the role of the base cost appears partially overlooked within the Bayesian methodology literature.

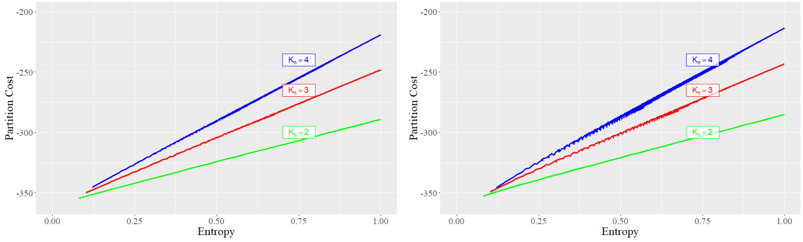
However, when BNP clustering methods are applied in practice, the choice of an appropriate base distribution is known to be crucial. The most common choice is to use an independent prior on the unique values so that $\theta_c \overset{iid}{\sim} P_0$ and

$$C_{\text{base}}(K_n, \boldsymbol{\theta}) = -\sum_{c=1}^{K_n} \log P_0(d\theta_c),$$

where the variance of the distribution P_0 is known to play an important role in the estimation process and, typically, the higher the variance of P_0 the lower the number of clusters identified by the posterior (cfr., e.g. Gelman et al., 2013, p. 535). This phenomenon can be explained by looking at the joint distribution induced by P_0 on the unique value. For instance, when P_0 is set to be a univariate normal distribution centered in μ and with variance σ^2, we have

$$C_{\text{base}}(K_n, \boldsymbol{\theta}) = \frac{K_n}{2}\log(2\pi) + \frac{K_n}{2}\log\sigma^2 + \frac{1}{2}\sum_{c=1}^{K_n} \frac{(\theta_c - \mu)^2}{\sigma^2}.$$

When the variance is increased from σ^2 to λ^2, intuitively the base cost increases for those vectors $(\theta_1, \dots, \theta_{K_n})$ whose components are similar and it decreases for vectors with more diverse components, thus ultimately favoring the variability of the unique values and penalizing many overlapping clusters. More formally, defining the K_n-sphere $\boldsymbol{\theta} \in \mathbb{R}^{K_n}$ such that $\sum_{c=1}^{K_n}(\theta_c - \mu)^2 = K_n \frac{\log(\lambda^2/\sigma^2)\sigma^2\lambda^2}{\lambda^2 - \sigma^2}$, we have that the cost increases for vectors $(\theta_1, \dots, \theta_{K_n})$ corresponding to points inside the sphere and decreases for those vectors corresponding to points outside the sphere. In practice, P_0 is usually set to be a continuous scale mixture, where the mixed density is conjugate to the kernel k for computational convenience, while the mixing density is used to increase appropriately the marginal scale of the mixture P_0.



(a) (b)

Fig. 1: Partition cost as function of entropy in a DPM model with $\alpha = 1$ (panel a) and in a PYPM model with $\alpha = 1$ and $\sigma = 0.5$ (panel b) for $n = 100$ observations clustered into 2 (blue line), 3 (red line), and 4 (green line) clusters.

Finally, let us comment on the partition cost C_{part}. Its behavior is less straightforward and we consider here only two important and widely used cases: Dirichlet process mixtures (DPM) and Pitman-Yor process (Pitman and Yor, 1997) mixtures (PYPM). With a DPM model, up to an additive constant, we have

$$C_{\text{part}}(K_n, \boldsymbol{c}; \alpha) = -K_n \log \alpha - \sum_{c=1}^{K_n} \log \Gamma(n_c),$$

where α is the concentration parameter of the Dirichlet Process. The DPM partition cost tends to favor parsimonious values of K_n (wrt to the likelihood cost that in general tends to favor $K_n = n$). However, contrary to the base cost, it depends also on clusters' frequencies.

Figure 1(a) showcases the partition cost of DPM for different values of what we refer henceforth to as the entropy of the frequencies $(n_1, \ldots, n_{K_n})$, i.e.

$$S(n_1, \ldots, n_{K_n}) = -\sum_{c=1}^{K_n} \frac{n_c}{n} \log_{K_n} \frac{n_c}{n}.$$

Overall the EPPF acts favoring frequencies $(n_1, \ldots, n_{K_n})$ with low entropy and thus, roughly speaking, higher sample variance of the frequencies. However, this feature ultimately results in two distinct effects: one acting on the total number of occupied clusters K_n and another acting on the variance of the clusters' frequencies $(n_1, \ldots, n_{K_n})$. Even though these two features both favor a reduced entropy, they entail very different scenarios in terms of estimated clustering structure, especially from an applied and practical point of view. Penalizing large numbers of clusters is typically desirable in applications because an elevated number of clusters may be difficult to interpret, however a partition with few dominating clusters and many sparsely populated clusters is highly undesirable because it is hard to interpret unless one decides to ignore all the information contained in the small clusters and focus only on the dominating ones. See also Green and Richardson (2001) for a study of the posterior entropy in the Dirichlet process mixture and Greve et al. (2022) for more details on entropy in mixture of finite mixture models. In the case of a PYPM the partition cost, up to an additive constant, equals

$$C_{\text{part}}(K_n, \boldsymbol{c}; \alpha, \sigma) = -\sum_{c=1}^{K_n} \log(\alpha + \sigma(c-1)) - \sum_{c=1}^{K_n} \log \Gamma(n_c - \sigma) + K_n \log \Gamma(1-\sigma).$$

Despite that the EPPFs are different, Figure 1 shows in both processes a closely similar behavior in terms of entropy penalization.

Note that Figure 1 provides us with insights into the behavior of the EPPFs evaluated in a vector of clusters' frequencies $(n_1, \dots, n_{K_n})$, i.e., the probability of a specific clustering configuration with unordered frequencies $\{n_1, \dots, n_{K_n}\}$. Note that the vectors $(n_1, \dots, n_{K_n})$ are not in a one-to-one correspondence with the partitions and the number of partitions corresponding to certain frequencies varies across vectors. The same is true for other marginal quantities such as the number of clusters K_n. For instance, the number of possible partitions rapidly increases with K_n accordingly to Stirling numbers of the second kind. Importantly, this information must also be considered combined with the partition cost evaluated in a specific partition, represented in Figure 1, if we are interested in fully understanding the impact of the EPPF on prior and posterior distributions of functionals of the partition, e.g., on the marginal distribution of K_n. Note that combining the two features the typical partition cost strongly penalized too many clusters suggested by the likelihood costs, i.e. $K_n = n$, but favors a small number of clusters with respect to n that adaptively increases with the sample size n, (See e.g., De Blasi et al., 2013). Considering both aspects is also important if we want to understand the effect of the partition cost on a point estimate of the clustering that is different from the MAP (maximum a posteriori) of the partition, but minimizes the Bayesian risk, i.e., posterior expected loss, according to flexible loss as discussed in the next section.

3 Regularized-entropy estimator

Once the posterior distribution $\mathbb{P}(\boldsymbol{c} \mid y_{1:n})$ over the space of partitions is obtained, typically thanks to a Markov Chain Monte Carlo algorithm, a point estimate $\hat{\boldsymbol{c}}$ of the partition can be obtained accordingly to the decision-theoretic approach of Bayesian analysis. More precisely, $\hat{\boldsymbol{c}}$ is obtained by minimizing the Bayesian risk, i.e, the expected value of a loss function $L(\boldsymbol{c}, \hat{\boldsymbol{c}})$ with respect to the posterior:

$$\boldsymbol{c}^* = \underset{\hat{\boldsymbol{c}}}{\operatorname{argmin}} \mathbb{E}[L(\boldsymbol{c}, \hat{\boldsymbol{c}}) \mid y_{1:n}] = \underset{\hat{\boldsymbol{c}}}{\operatorname{argmin}} \sum_{\boldsymbol{c}} L(\boldsymbol{c}, \hat{\boldsymbol{c}}) \mathbb{P}(\boldsymbol{c} \mid y_{1:n}),$$

where $L(\boldsymbol{c}, \hat{\boldsymbol{c}})$ is the loss in which we incur using $\hat{\boldsymbol{c}}$ as estimates when the partition takes the value $\boldsymbol{c}$. How to interpret and elicit the loss in practice can change according to the philosophical point of view. Often in parameter estimation the loss is interpreted as the cost of choosing $\hat{\boldsymbol{c}}$ instead of the ideally optimal parameter value $\boldsymbol{c}$ (sometimes interpreted as the *truth*). In a more subjective Bayesian framework,

Algorithm 1 Entropy-regularized estimates

```
**Input**: MCMC chain of partitions $\{\boldsymbol{c}_m, m = 1,\dots,M\}$, $\lambda$
**Output**: point estimate $\boldsymbol{c}^*$
Compute $S(\boldsymbol{c}_m)$ for $m = 1,\dots,M$
Compute $w_m = \exp\{\lambda S(\boldsymbol{c}_m)\}$ for $m = 1,\dots,M$
$\bar{w}_m \leftarrow w_m / \sum_m w_m$ for $m = 1,\dots,M$
Generate $\{\tilde{\boldsymbol{c}}_m, m = 1,\dots,M\}$, sampling with replacement from $\{\boldsymbol{c}_1,\dots,\boldsymbol{c}_M\}$ with prob. $\{\bar{w}_m, m = 1,\dots,M\}$
$\boldsymbol{c}^* \leftarrow \operatorname{argmin} \sum_{m=1}^{M} \sum_{\hat{\boldsymbol{c}}} L(\tilde{\boldsymbol{c}}_m, \hat{\boldsymbol{c}})$
```

it can be interpreted, together with the model and prior, in terms of the preferences implied on the possible parameter values $\boldsymbol{c}$ via the Bayesian risk. Finally, also in a more frequentist framework the loss can be chosen in terms of the implied properties of the estimator of the unknown parameter $\hat{\boldsymbol{c}}$.

Despite the different philosophical justifications, rarely in applied Bayesian clustering analysis a 0-1 loss function and the resulting MAP estimator are employed due to the large support of the posterior and the fact that the 0-1 loss function does not reflect different levels of distance between two non-coinciding partitions. Widely used alternatives in applications are Binder loss (Binder, 1978) or variation of information loss (see, Meilă, 2007; Wade and Ghahramani, 2018; Dahl et al., 2021).

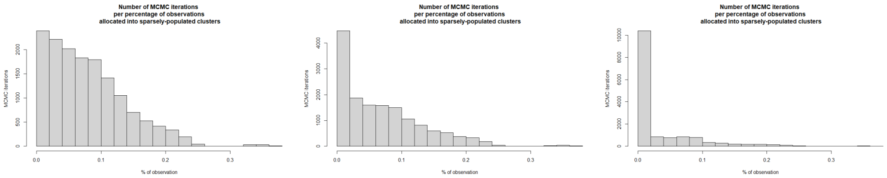
We have already stressed how a large presence of noisy clusters is typically undesirable in practice and we claim that this aspect should be reflected in the loss function used for point estimation, so that the loss of each partition is proportional to its entropy. To do so, consider any possible loss function $L(\boldsymbol{c}, \hat{\boldsymbol{c}})$ one would like to use to derive the estimate, we can define a new loss function, that we named entropy-regularized, as

$$\bar{L}(\boldsymbol{c}, \hat{\boldsymbol{c}}) = \exp\{\lambda S(\boldsymbol{c})\} L(\boldsymbol{c}, \hat{\boldsymbol{c}}),$$

where, with a little abuse of notation wrt the previous section, $S(\boldsymbol{c})$ is the entropy of the partition identified by $\boldsymbol{c}$ and $\lambda \in \mathbb{R}$. Recall that the base of the logarithm involved in the computation of $S(\boldsymbol{c})$ changes with the argument $\boldsymbol{c}$ and it is equal to the number of unique values in $\boldsymbol{c}$, so that $S(\boldsymbol{c}) = 1$ can be obtained for any number of non-empty clusters $K_n \geq 2$ (provided that $n/K_n \in \mathbb{N}$). Clearly, when λ is positive, for any candidate estimate $\hat{c}$, the loss function is inflated in correspondence of partitions $\boldsymbol{c}$ with high entropy, as desired.

Minimizing the expected entropy-regularized loss function $\bar{L}(\boldsymbol{c}, \hat{\boldsymbol{c}})$ with respect to the posterior is equivalent to minimizing the original loss function $L(\boldsymbol{c}, \hat{\boldsymbol{c}})$ with respect to an entropy-regularized version $\bar{\mathbb{P}}[\boldsymbol{c} \mid y_{1:n}]$ of the posterior distribution, i.e.

$$\bar{\mathbb{P}}[\boldsymbol{c} \mid y_{1:n}] \propto \exp\{\lambda\, S(\boldsymbol{c})\} \mathbb{P}[\boldsymbol{c} \mid y_{1:n}].$$



(a) Without entropy regularization.

(b) With entropy regularization for $\lambda = 10$.

(c) With entropy regularization for $\lambda = 20$.

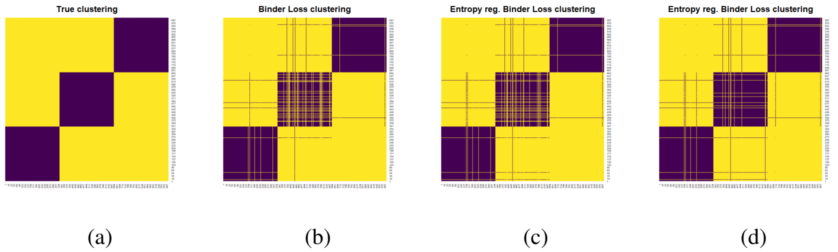
Fig. 2: Percentage of observations in sparsely-populated clusters before and after entropy-regularization.

This result, while immediate to prove, is highly desirable, because it allows implementation of the entropy-correction in a very straightforward and computationally feasible way which is described in Algorithm 1.

4 Simulation study

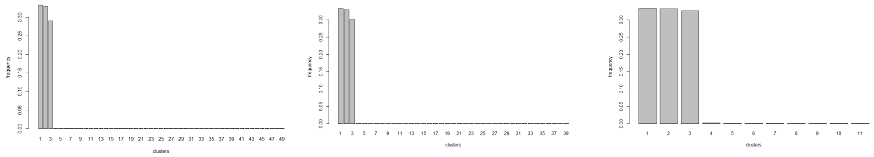
We provide here a simulation study, where $n = 1000$ observations are sampled from 3 different univariate Gaussian distributions. Here we refer to "true" clustering as the one implied by the memberships indicators of the Gaussian kernels under the data generating truth. We employ a normal-normal DPM and we compare the posterior estimates obtained minimizing the Binder loss function and the entropy-regularized Binder loss function. We set the concentration parameter $\alpha = 1$, perform 20 000 MCMC simulations, and use the first 5000 as burnin. Defining as sparsely populated clusters those clusters containing 10% or less of observations, we found that in almost a third (4755 out 15 000) of the MCMC iterations, 10% or more of the observations are allocated into sparsely populated clusters, while in almost two thirds (9306 out of 15 000) of MCMC iterations, 5% or more of the observations are allocated into sparsely populated clusters, see Figure 2a. The same counts after entropy-regularization of the posterior (as described in the previous section) are, with $\lambda = 10$, 3981 and 7825 out 15 000, see Figure 2b, and, with $\lambda = 20$, 1393 and 3366 out 15 000, see Figure 2c. However, notice that coherently with the interpretation of the regularization in terms of the loss function, the regularized posterior should be intended only as a computational tool to provide a summary of the posterior distribution and not as an alternative posterior. So that, for instance, uncertainty quantification should be computed using the original posterior.

Finally, Figure 3 shows the true and the estimated clusters with and without entropy regularization and they highlight how the regularization acts allocating observations from noisy clusters into dominating ones. Finally, Figure 4 shows the cluster frequencies for the three point estimates.



(a) (b) (c) (d)

Fig. 3: Estimated clustering for the simulation study darker squares denote couples of observations clustered together. Panel (a) shows the true clustering. Panel (b) shows the clustering minimizing the Binder loss. Panel (c) shows the clustering minimizing the entropy-regularized Binder loss for $\lambda = 10$ and panel (d) for $\lambda = 20$.



(a) Without entropy regularization.

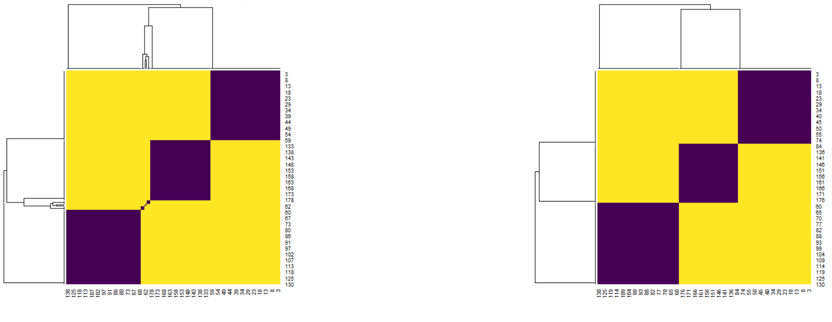
(b) With entropy regularization for $\lambda = 10$.

(c) With entropy regularization for $\lambda = 20$.

Fig. 4: Estimated clusters' frequencies.

5 Results for the wine dataset

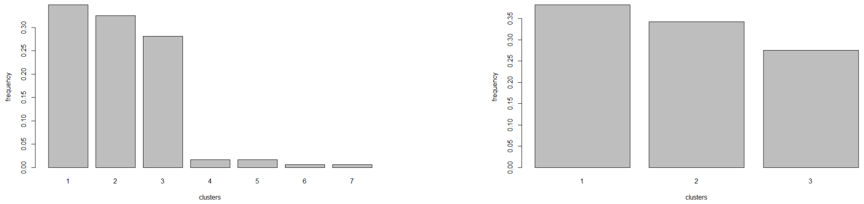
We test the performance of our method also on the wine dataset available on R, where data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. Here we refer to the clustering identified by the three types of wines as "ground truth". We use the 13 constituents to estimate a Dirichlet process mixture model with multivariate Gaussian kernels, and we try to recover the three groups of types of wine through the estimated clustering. After running the MCMC for 10000 iterations and using the first 2000 as burnin, the Binder loss function identifies a partition of seven clusters, while our estimator for $\lambda = 20$ identifies three clusters. See Figure 5 and Figure 6. Lastly, Figure 7 compares the clustering based on three groups of types of wine with the two estimates.



(a) Estimated partition without entropy-regularization.

(b) Estimated partition after entropy-regularization.

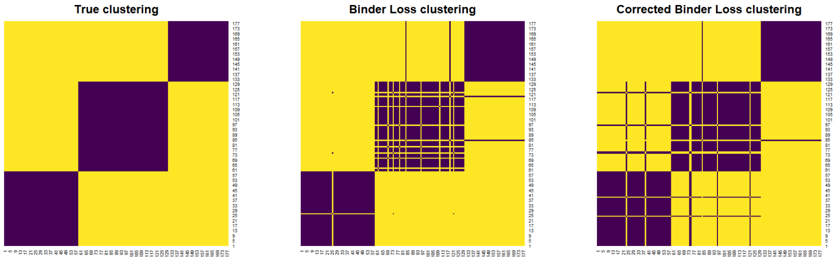
Fig. 5: Estimated partitions for the wine dataset. Darker squares denote couples of observations clustered together, observations are ordered based on co-clustering.



(a) Without entropy-regularization.

(b) With entropy-regularization.

Fig. 6: Estimated clusters' frequencies for the wine dataset.



(a)

(b)

(c)

Fig. 7: Estimated clustering for the wine dataset. Darker squares denote couples of observations clustered together, observations are ordered based three groups of types of wine. Panel (a) shows the clustering ground truth. Panel (b) shows the clustering minimizing the Binder loss. Panel (c) shows the clustering minimizing the entropy-regularized Binder loss for $\lambda = 20$.

References

Argiento, R. and M. De Iorio (2019). Is infinity that far? A Bayesian nonparametric perspective of finite mixture models. *Preprint arXiv: 1904.09733*.

Beraha, M., R. Argiento, J. Møller, and A. Guglielmi (2021). MCMC computations for Bayesian mixture models using repulsive point processes. *J. Comput. Graph. Stat.*, in press.

Binder, D. A. (1978). Bayesian cluster analysis. *Biometrika 65*, 31–38.

Blei, D. M., A. Y. Ng, and M. I. Jordan (2003). Latent Dirichlet allocation. *J. Mach. Learn. Res. 3*, 993–1022.

Dahl, D. B., D. J. Johnson, and P. Müller (2021). Search algorithms and loss functions for Bayesian clustering. *Preprint arXiv: 2105.04451*.

De Blasi, P., S. Favaro, A. Lijoi, R. H. Mena, I. Prünster, and M. Ruggiero (2013). Are Gibbs-type priors the most natural generalization of the Dirichlet process? *IEEE Trans. Pattern Anal. Mach. Intell. 37*, 212–229.

Ferguson, T. S. (1983). Bayesian density estimation by mixtures of normal distributions. In *Recent advances in statistics*, pp. 287–302. Elsevier.

Franzolini, B., A. Lijoi, and I. Prünster (2022). Model selection for maternal hypertensive disorders with symmetric hierarchical Dirichlet processes. *Ann. Appl. Stat.*, in press.

Gelman, A., J. Carlin, H. Stern, D. Dunson, A. Vehtari, and D. Rubin (2013). *Bayesian data analysis*. Chapman and Hall/CRC.

Gil-Leyva, M. F., R. H. Mena, and T. Nicoleris (2020). Beta-Binomial stick-breaking non-parametric prior. *Electron. J. Stat. 14*, 1479–1507.

Green, P. J. and S. Richardson (2001). Modelling heterogeneity with and without the Dirichlet process. *Scand. J. Stat. 28*, 355–375.

Greve, J., B. Grün, G. Malsiner-Walli, and S. Frühwirth-Schnatter (2022). Spying on the prior of the number of data clusters and the partition distribution in Bayesian cluster analysis. *Aust. N. Z. J. Stat.*, in press.

Hennig, C. (2015). What are the true clusters? *Pattern Recognit. Lett. 64*, 53–62.

Lijoi, A., R. H. Mena, and I. Prünster (2007). Controlling the reinforcement in Bayesian non-parametric mixture models. *J. R. Stat. Soc. Series B Stat. Methodol. 69*, 715–740.

Lijoi, A. and I. Prünster (2010). Models beyond the Dirichlet process. In N. L. Hjort, C. Holmes, P. Müller, and S. G. Walker (Eds.), *Bayesian nonparametrics*. Cambridge University Press.

Lindsay, B. G. (1995). Mixture models: theory, geometry, and applications. In *NSF-CBMS Regional Conf. Series in Prob. and Stat.*, Volume 5.

Lo, A. Y. (1984). On a class of Bayesian nonparametric estimates: I. Density estimates. *Ann. Stat. 12*, 351–357.

McLachlan, G. J., S. X. Lee, and S. I. Rathnayake (2019). Finite mixture models. *Annu. Rev. Stat. Appl. 6*, 355–378.

Meilă, M. (2007). Comparing clusterings—an information based distance. *J. Multivar. Anal. 98*, 873–895.

Miller, J. W. and M. T. Harrison (2018). Mixture models with a prior on the number of components. *J. Am. Stat. Assoc. 113*, 340–356.

Ngan, H. Y., N. H. Yung, and A. G. Yeh (2015). Outlier detection in traffic data based on the Dirichlet process mixture model. *IET Intell. Transp. Syst. 9*, 773–781.

Petralia, F., V. Rao, and D. Dunson (2012). Repulsive mixtures. *Adv. Neural Inf. Process. Syst. 25*, 1889–1897.

Petrone, S., M. Guindani, and A. E. Gelfand (2009). Hybrid Dirichlet mixture models for functional data. *J. R. Stat. Soc. Series B Stat. Methodol. 71*, 755–782.

Pitman, J. (1996). Some developments of the Blackwell-MacQueen urn scheme. *Lect. notes-monogr. ser. 30*, 245–267.

Pitman, J. and M. Yor (1997). The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Ann. Probab. 25*, 855–900.

Polyanskiy, Y. and Y. Wu (2020). Self-regularizing property of nonparametric maximum likelihood estimator in mixture models. *Preprint arXiv: 2008.08244*.

Saha, S. and A. Guntuboyina (2020). On the nonparametric maximum likelihood estimator for Gaussian location mixture densities with application to Gaussian denoising. *Ann. Stat. 48*, 738–762.

Shotwell, M. S. and E. H. Slate (2011). Bayesian outlier detection with Dirichlet process mixtures. *Bayesian Anal. 6*, 665–690.

Stephens, M. (2000). Dealing with label switching in mixture models. *J. R. Stat. Soc. Series B Stat. Methodol. 62*, 795–809.

Wade, S. and Z. Ghahramani (2018). Bayesian cluster analysis: Point estimation and credible balls (with discussion). *Bayesian Anal. 13*, 559–626.

Xu, Y., P. Müller, and D. Telesca (2016). Bayesian inference for latent biologic structure with determinantal point processes (DPP). *Biometrics 72*, 955–964.

Zhang, C., Y. Qin, X. Zhu, J. Zhang, and S. Zhang (2006). Clustering-based missing value imputation for data preprocessing. In *4th IEEE Int. Conf. Industr. Inform.*, pp. 1081–1086.