Powered Dirichlet Process

Controlling the “Rich-Get-Richer” Assumption in Bayesian Clustering

Gaël Poux-Médard[a], Julien Velcin, Sabine Loudcher

[a] https://gaelpouxmedard.github.io/

The Dirichlet Process is one of the most widely used priors in Bayesian clustering. It works by sampling an *a priori* cluster for a datapoints that come in sequential order. The “rich-get-richer” property is key in this process. It states that the *a priori* probability of sampling a given a cluster depends linearly on its population.
However, such hypothesis is not necessarily accurate. Cluster sampling probabilities may depend on the number of observations in a nonlinear fashion, if they depend on it at all. As an answer to this statement, we derive the Powered Dirichlet Process and derive some of its properties (expected number of clusters, convergence). Unlike state-of-the-art efforts in this direction, this new formulation allows for direct tuning of the importance of the “rich-get-richer” hypothesis. From a broader point of view, this work invites to rethink some of the most widespread clustering models (LDA, IGMM, …) in the light of alternative prior formulations.

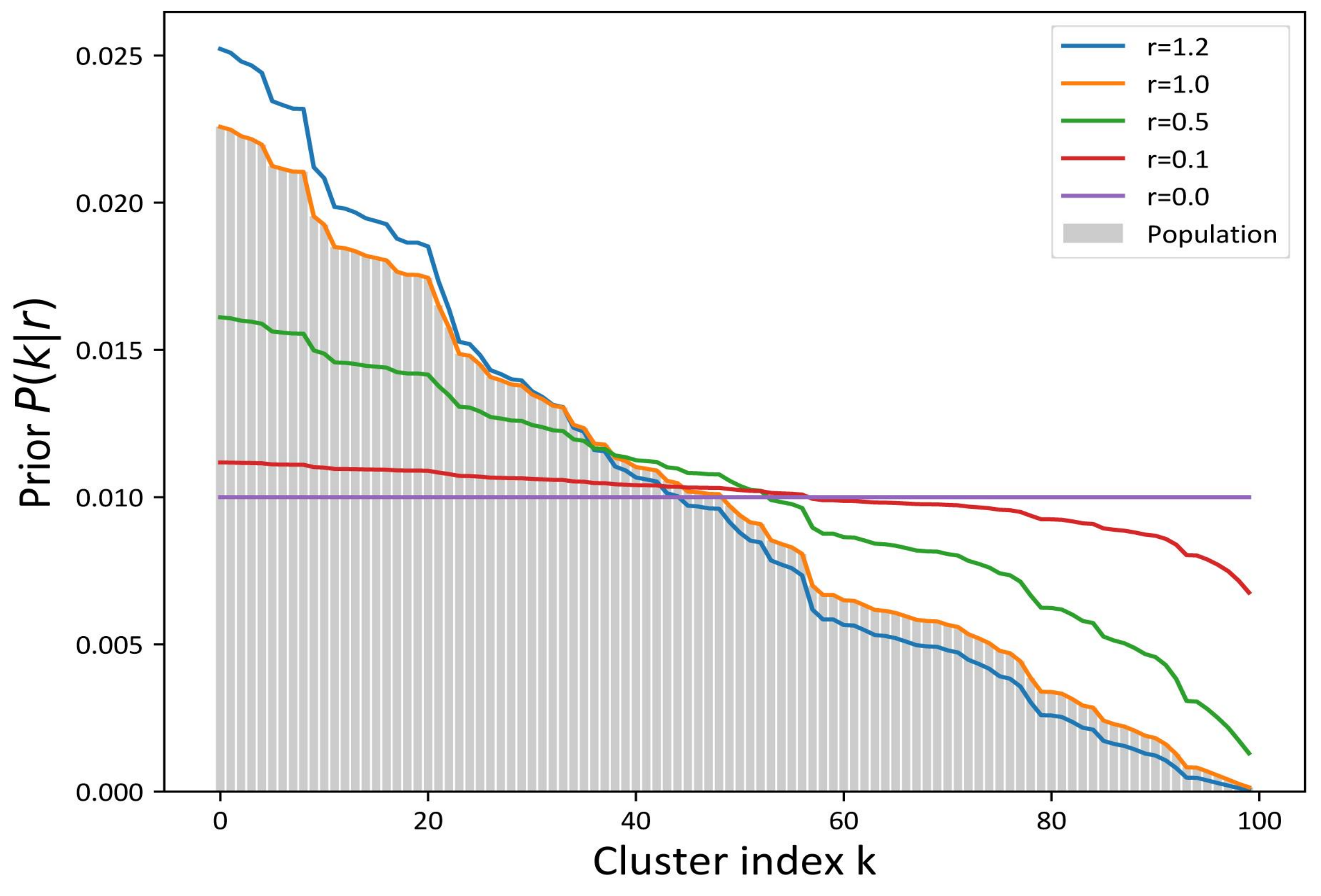


Figure 1 – In the vanilla Dirichlet Process, the prior probability of sampling a cluster depends linearly on its population. Our formulation relaxes this hypothesis and allows nonlinear dependences by tweaking a single hyperparameter r. When r=1, we recover the vanilla Dirichlet Process (rich-get-richer) ; when r<1, populated clusters have less chances to get sampled (rich-get-less-richer) ; when r>1, populate clusters have even more chances to get sampled (rich-get-more-richer) ; when r=0, the dependence on population disappears (rich-get-no-richer).

A generalization of previous works

Dirichlet Processes are convenient priors for clustering streams of data. Each new observation is considered sequentially and associated to one cluster among an infinity. This choice depends on the cluster’s likelihood times its *a priori* probability of being sampled. Once this observation’s cluster has been sampled, its population increases by 1 and the process continues with the next data point. In most cases, the *a priori* probability for a cluster to be sampled is proportional to its population; this is the **Dirichlet Process**.
While being convenient, this implies a strong *a priori* hypothesis on the data generation process; populated clusters tend to be even more populated, or “rich-get-richer”. Consequently, the number of non-empty clusters grows as the logarithm of the number of observations. In many use cases of Dirichlet Processes, this is an unreasonable assumption. When clustering a stream of news articles, for instance, there is no *a priori* reason for the number of clusters to be limited by a logarithmic growth. On the contrary, an intuitive guess would be that new information topics (non-empty clusters) emerge at a constant rate. The **Uniform Process** has been developed as an answer to this problem. Here, the probability of sampling any non-empty cluster is proportional to a constant i.e. does not depend on its population: rich-get-no-richer.
Now, we argue that there is a range of intermediate hypotheses between the Uniform and the Dirichlet Process. Taking back the example of news stream clustering, original topics may not emerge at a constant rate. Recurrent topics, for instance, can be grouped in a single cluster; however, there is no reason for novelty to appear at a logarithmic rate either. The most fit hypothesis may lie between Uniform and Dirichlet Processes. This is why we formulate the **Powered Dirichlet Process** for any new observation as follows (full demonstration starting from the Dirichlet and Multinomial distributions in the main paper):

$$P(\text{cluster}|N,r) \propto \begin{cases} N_c^r \text{ for non-empty clusters} \\ \alpha \text{ for any empty cluster} \end{cases}$$

where N_c is the population of cluster c, α the concentration parameter and r a hyperparameter. When r=1, this formulation is identical to the Dirichlet Process (rich-get-richer). When r=0, this formulation is identical to the Uniform Process (rich-get-no-richer). But in between those values lie novel flavours of hypotheses: when 0<r<1, rich-get-less-richer, when r>1, rich-get-more-richer, and when r<0, poor-get-richer. The implications are illustrated in Figure 1.

Remarks

Other Dirichlet-based processes

The Dirichlet Process is the base of numerous alternative clustering processes. Some modify it to favour the apparition of new clusters by adding a discount parameter (Pitman-Yor process) or to allow mixed-membership among clusters (Indian buffet process), other combine layers of Dirichlet Processes to get more coherent clusters (Hierarchical DP) or to get a tree-like clustering result (Nested DP). Other variants such as the Gamma Generalized Process use a different approach to favour the apparition of new clusters. But one thing all these processes share is their linear dependence on the population of non-empty cluster. Due to being based on the Dirichlet Process, our works comes not as a replacement for each of them, but as a complement; each can be revisited in the light of a nonlinear dependence on clusters’ population.

About α

The concentration parameter α is often fine-tuned to get a satisfying number of clusters. This ad-hoc practice works for fixed-size datasets but fails as soon as new data is added. In the Dirichlet Process, the number of expected clusters grows with the number of observations N as $\alpha \cdot \log(N)$. Tuning it rescales the number of clusters, but does not change the logarithmic dependence on N. Therefore, when N grows, α must be fine-tuned once again.
On the contrary, fine-tuning the “rich-get-richer” hypothesis (that is, the hyperparameter r) only once is enough, as it changes the dependency of the expected number of clusters on the number of observations N.

TL;DR

- Dirichlet Processes clustering priors come with a questionable “rich-get-richer” hypothesis.
- The Powered Dirichlet Process is the first approach to allow for its direct control.
- New data can now be fed to models without the need to fine-tune the hyperparameters another time.
- Our work generalizes existing results on convergence and expected number of clusters of DP.
- Numerical experiments support our claims that “rich-get-richer” hypothesis is not always optimal.
- PDP invites to rethink and extend most of existing Bayesian clustering methods

Key properties

We derive elementary results on the Powered Dirichlet Process. Full demonstrations are presented in the main paper.

Convergence

The resulting distribution of the Powered Dirichlet Process when $N \to \infty$ can be formulated as $P(cluster|N \to \infty, r)$ (illustrated Figure 2-left):

- $r = 0$: the PDP is by essence a uniform distribution (Uniform Process)
- $r = 1$: the PDP converges to a Dirichlet distribution.
- $0 < r < 1$: the PDP converges to a uniform distribution.
- $r > 1$: the PDP converges to a Dirac distribution.

Expected number of clusters

The expected number of non-empty clusters of the Powered Dirichlet Process when N is finite, noted $E(K|N,r)$, is proportional to:

- $r \leq 1$: $E(K|N,r) \propto \sum_{n=1}^{N} n^{-\frac{r^2+1}{2}} \coloneqq H_{\frac{r^2+1}{2}}(N)$
- $r \geq 1$: $E(K|N,r) \propto \sum_{n=1}^{N} n^{-r} \coloneqq H_r(N)$

where $H_m(n)$ is the generalized harmonic number.

From this result, we derive (illustrated Figure 2-right):

- $r = 1$: $E(K|N,r) \propto \log(N)$, a standard DP result.
- $r < 1$: $E(K|N,r) \propto N^{\frac{1-r^2}{2}}$
- $r > 1$: $E(K|N,r) \propto \zeta(\frac{r^2+1}{2})$ where $\zeta(x)$ is the Riemann zeta function.

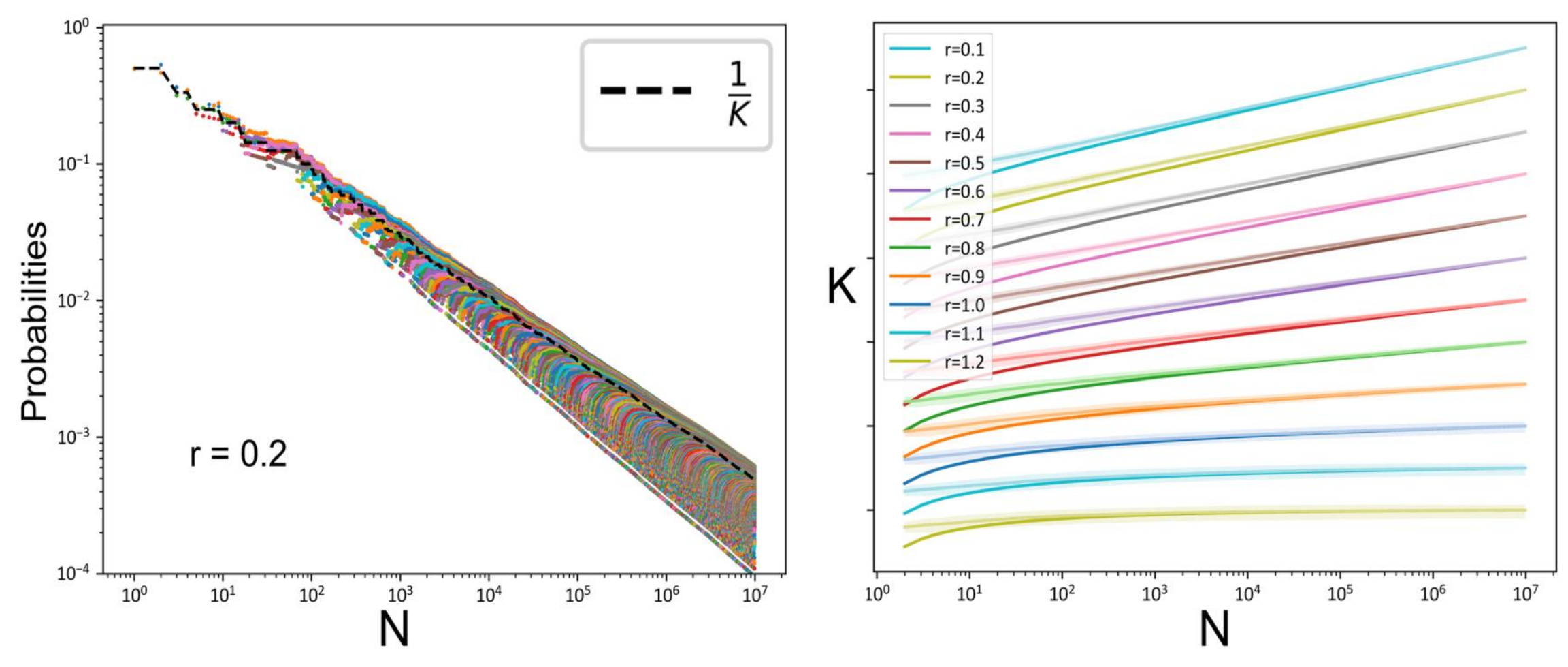
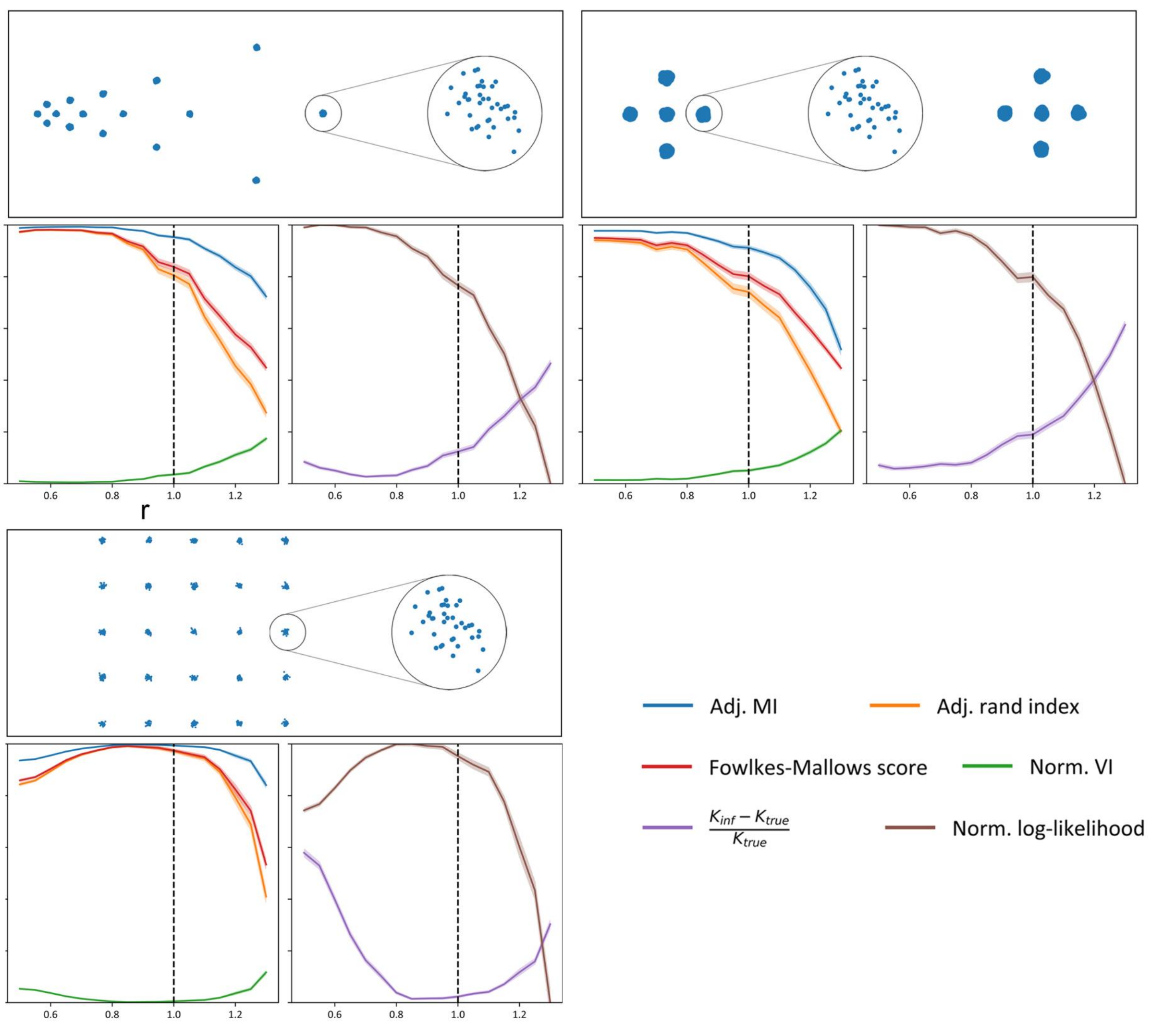


Figure 2 – (left) For one run of PDP(r=0,2,N), probability for each cluster (each colour) to be chosen at each step. The distribution converges towards a uniform distribution over all clusters $P(cluster\ |r < 1, N \to \infty) = \frac{1}{K}$. (right) Experimental number of clusters (std over 100 runs) and theoretical evolution of $E(K|N,r)$ (rescaled)

Experimental results

Infinite Gaussian Mixture Model (IGMM) applied to three synthetic and three real-world datasets for various iterations of PDP(r).



		Adj.MI (↑)	Adj.RI (↑)	Norm.VI (↓)	$\frac{K_{inf}-K_{true}}{K_{true}}$ (↓)
Iris	PDP (r=0.90)	**0.868(4)**	**0.866(7)**	**0.057(2)**	**0.000(0)**
	DP (r=1.00)	0.843(6)	0.820(12)	0.065(2)	0.030(10)
	UP (r=0.00)	0.544(2)	0.295(3)	0.303(2)	2.777(32)
Wines	PDP (r=0.10)	**0.712(15)**	**0.637(20)**	**0.102(5)**	**0.157(17)**
	DP (r=1.00)	0.589(19)	0.461(16)	0.128(4)	0.327(13)
	UP (r=0.00)	**0.713(17)**	**0.657(21)**	**0.103(5)**	**0.147(17)**
Cancer	PDP (r=0.10)	**0.254(17)**	**0.278(21)**	**0.118(1)**	**0.000(0)**
	DP (r=1.00)	0.085(16)	0.094(19)	0.108(2)	**0.000(0)**
	UP (r=0.00)	**0.271(17)**	**0.300(21)**	**0.118(1)**	**0.000(0)**

Discussion and perspectives

The PDP is a generalization of the Dirichlet Process that allows for direct control of the “rich-get-richer” hypothesis. Because this is a very common assumption, the PDP opens a way to rethink and hopefully revisit existing clustering methods under a different light. The scalability of existing sequential approaches to data flows is an important point that our method offers to solve.
However, there is more to our work. Fine-tuning the “rich-get-richer” hypothesis can also be considered a way to control the informativeness of clusters’ population in the modelling, that is the informativeness of the prior probability as a whole. In Figure 1, the prior probability becomes flatter as r diminishes: a priori information is less discriminative. This is especially interesting for variants of Dirichlet Process where counts are replaced by other quantities. In the Dirichlet-Hawkes Process, Dirichlet-Counting Process, or other variants where counts are replaced by an alternative function, the Powered variant directly controls how much importance is given to this information.
In general, we believe –and hope– that the Powered Dirichlet Process will constitute an interesting tool to refine and question the hypotheses made in a large part of state-of-the-art clustering methods.