Pitman-Yor Process and Pitman-Yor Mixture Models

Sara Capozio, Luca De Simone, Anna Petranzan

July 2024

1 Introduction

The Pitman-Yor process is a generalization of the Dirichlet process, applicable when modeling phenomena that follow asymmetric distributions with heavy tails. The Dirichlet process is inadequate for modeling such types of distributions because the average number of distinct clusters K_n generated by the process grows logarithmically as the sample size increases:

$$E[K_n] \approx a \log(n).$$

From this expression, it is clear that beyond a certain n he average number of clusters stabilizes, which leads, as n, increases, to very similar clusters in terms of size.
From an empirical perspective, this representation is unrealistic because, if we wanted to model phenomena with highly skewed distributions, such as the frequency of words or the number of followers per user on Twitter, K_n should follow a distribution with heavy tails, such as a power-law distribution.

Definition 1. A probability distribution P follows a power-law distribution if its density function has the following form:

$$p(x) = c \cdot x^{-a}.$$

The process that generates a number of distinct clusters distributed according to a power-law distribution is the Pitman-Yor process, denoted as $PY(d, c, G_0)$. Unlike the Dirichlet process, it is characterized by an additional parameter, d.

Definition 2. Let G_0 be the base probability measure on $\mathbb{X}$ (the support of the observations), $d \in [0, 1)$ the penalization coefficient, and $c > -d$ the concentration parameter. Then, the random probability measure

$$\tilde{P} = \sum_{j=1}^{\infty} p_j \cdot \delta_{X_j}$$

where:

$$(X_j)_{j \geq 1} \overset{\text{iid}}{\sim} G_0$$
$$(p_j)_{j \geq 1} \sim GEM(c, d)$$

is called the Pitman-Yor process.

In the following case as well, the weights of a Pitman-Yor process can be constructed using the stick-breaking process, where the components are defined as follows:

$$(W_j)_{j \geq 1} \overset{\text{ind}}{\sim} Beta(1 - d; c + j \cdot d)$$

$$p_1 = W_1$$
$$p_j = W_j \cdot \prod_{r=1}^{j-1}(1 - W_r).$$

2 Special Cases of the Pitman-Yor Process

The following section aims to illustrate two special cases of the Pitman-Yor process:
1. Case $d = 0$: By setting the penalization parameter to zero, the Pitman-Yor process coincides with the Dirichlet process. Let $E_i \overset{\text{iid}}{\sim} Exp(1)$ for $i = 1, 2, \ldots$, be the inter-arrival times of a homogeneous Poisson process with unit arrival rate, and let $\Gamma_k = E_1 + \cdots + E_k$ be the waiting times for the k-th event. Let γ_α be a gamma subordinator, i.e., a stochastic gamma process with stationary and independent increments, such that $\gamma_\alpha = \sum_{j=1}^{\infty} J_k$, where $J_1 > J_2 > \cdots > 0$ is the sequence of jumps ordered within the time interval $[0, \alpha]$. Let $\nu(x)$ be the Lévy measure associated with a gamma process that measures the frequency of jumps of size x.

$$\nu(x) = \alpha \int_x^{\infty} xe^{-x}\, dx$$

By exploiting the quantities just specified, it is possible to define an alternative method to stick breaking for determining the weights of a Dirichlet process:

$$\tilde{P} = \sum_{k=1}^{\infty} \frac{\nu(\Gamma_k)^{-1}}{\sum_{k=1}^{\infty} \nu(\Gamma_k)^{-1}} \cdot \delta_{X_k} = \sum_{k=1}^{\infty} \frac{J_k}{\sum_{k=1}^{\infty} J_K} \cdot \delta_{X_k} \overset{\text{d}}{=} V_1 \cdot \delta_{X_1} + \sum_{k=2}^{\infty} \left[\left(\prod_{j=1}^{k-1} (1 - V_j) \right) V_k \right] \cdot \delta_{X_k}$$

When the atoms are sampled from the base measure, the first variable drawn will be assigned a higher weight, as the waiting time required to observe X_1 is reduced with an associated frequency equal to the largest increment, i.e., J_1. From the Central Limit Theorem, it is known that the sum of N independent and identically distributed variables, as $N \to \infty$, follows a Normal distribution. For this reason, the Dirichlet process is expected to distribute symmetrically. On the other hand, when the goal is to model asymmetric phenomena, the Pitman-Yor process comes into play.

2. Case $c = 0$: By setting the concentration parameter to zero, the resulting process is a Pitman-Yor process $PY(d)$. Let $E_i \overset{\text{iid}}{\sim} Exp(1)$, the inter-arrival times of a homogeneous Poisson process with unit arrival rate, and $\Gamma_k = E_1 + \cdots + E_k$ the waiting times for the k-th event. Let γ_α be a stable subordinator. The key feature of this process is that it preserves the asymmetry of the phenomenon. In fact, it is known that the sum of N independent and identically distributed random variables follows a stable distribution with the same stability parameters, where for $\alpha < 2$, the stable distribution coincides with a power law. The Lévy measure associated with a process that follows a stable distribution is as follows:

$$\nu(x) = c \cdot x^{\alpha}.$$

By exploiting the quantities just specified, it is possible to define an alternative method to the stick-breaking method for sampling the weights of a $PY(d)$ process:

$$\tilde{P} = \sum_{k=1}^{\infty} \frac{\Gamma_k^{-1/\alpha}}{\sum_{k=1}^{\infty} \Gamma_k^{-1/\alpha}} \cdot \delta_{X_k} = \sum_{k=1}^{\infty} \frac{J_k}{\sum_{k=1}^{\infty} J_k} \cdot \delta_{X_k} \overset{\text{d}}{=} V_1 \cdot \delta_{X_1} + \sum_{k=2}^{\infty} \left[\left(\prod_{j=1}^{k-1} (1 - V_j) \right) V_k \right] \cdot \delta_{X_k}.$$

3 Simulative Analysis

Through the use of the R software, several simulations were carried out to understand the roles of the parameters c and d and to analyze the differences between the Dirichlet process and the Pitman-Yor process.

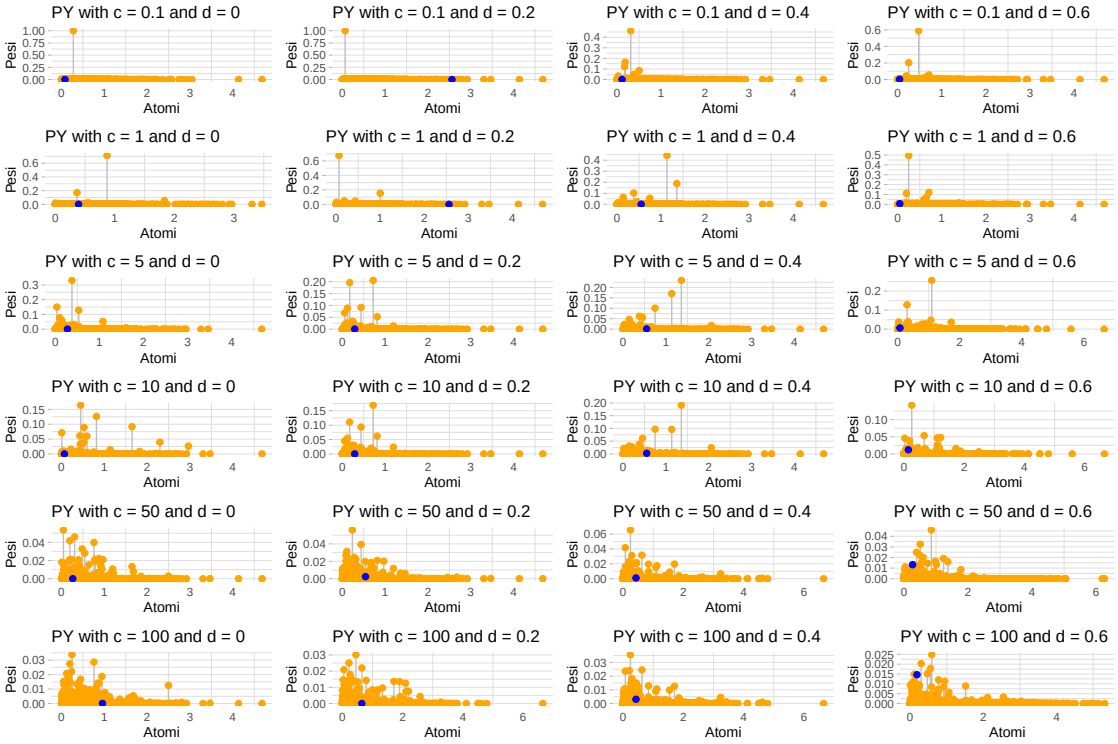

Figure 1: Scatterplot of the simulated weights

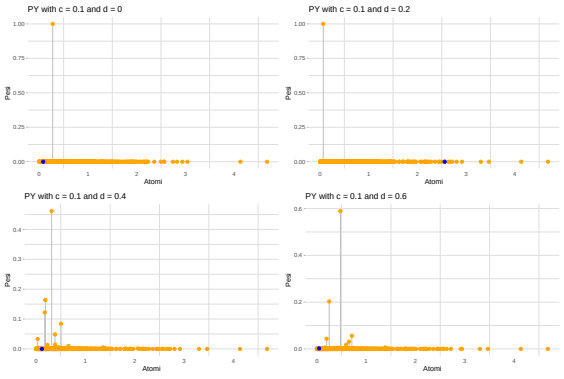

Figure 2: $c = 0.1$

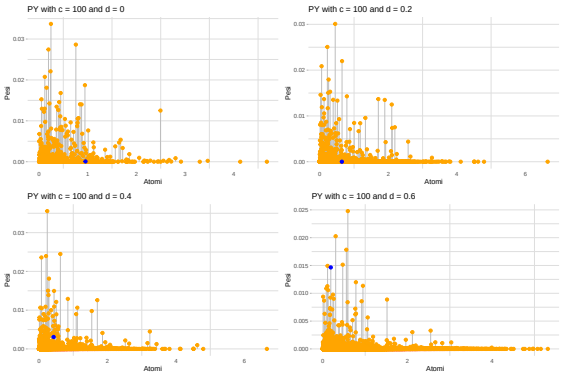

Figure 3: $c = 100$

Figure 4: Scatterplot of the simulated weights for fixed c

From the following graphs, it is possible to study the role of the parameters c and d within the process. The concentration parameter c defines the strength of the prior guess G_0, i.e., it establishes how confident we are that the realizations of the process will be distributed according to the base measure G_0. For this study, it was assumed a priori to use a Gamma($\alpha = 1, \beta = 2$) distribution, as it is more suitable for modeling asymmetric phenomena with heavier or lighter tails. The penalization parameter d allows modeling the tails

of the prior distribution. In fact, from the simulation, it emerges that increasing d penalizes the probability mass assigned to the more frequent atoms, redistributing it over a greater number of realizations x_j, making the tails of the distribution heavier.

From the analysis (Figure 4), it is possible to observe that the parameters c and d work in the same direction, meaning that for low values of these parameters, the probability mass of the realizations of $\tilde{P}$ is concentrated on few atoms. On the other hand, as these values increase, a greater number of atoms are assigned a significantly positive probability mass.

In the Pitman-Yor process, the sampled atoms have a higher multiplicity and lower density compared to those generated by a Dirichlet process. In fact, by analyzing more closely the stick-breaking construction of the two processes, it is possible to observe that in the Dirichlet process, the random variables $W_j \overset{\text{iid}}{\sim} \text{Beta}(1, c)$ are independent and identically distributed, with an expected value given by

$$E[W_j] = \frac{1}{c+1}.$$

On the other hand, in the Pitman-Yor process, the variables $W_j \overset{\text{ind}}{\sim} \text{Beta}(1-c, c+jd)$ are only independent, with an expected value

$$E[W_j] = \frac{1-d}{1-d+c+jd},$$

which decreases as j increases.

The formulas highlight that, on average, the weights generated by the Dirichlet process are greater than those of the Pitman-Yor process, with the difference increasing as j grows. This implies that, in order to exhaust the total probability, a larger number of weights and consequently a larger number of atoms must be generated. This consideration leads to a computational issue for high values of d, due to the algorithm's inability to fully exhaust the remaining probability $1 - \sum_{j=1}^{N} p_j$, and thus generate a valid probability measure. To mitigate this problem, one strategy is to set a priori a truncation by defining a certain threshold $\epsilon > 0$, which is the residual probability we are willing to accept. If $1 - \sum_{j=1}^{N} p_j > \epsilon$, the generation of new realizations continues. However, this technique can be computationally expensive, which is why in the analysis, different truncation values were considered for different combinations of c and d. Note that using different values of ϵ may introduce difficulties when comparing graphs, as a higher concentration of weights may be attributable to a greater multiplicity due to a different parameter value.
In Figures 1 and 4, the residuals obtained $1 - \sum_{j=1}^{N} p_j$ are identified by the blue dots.

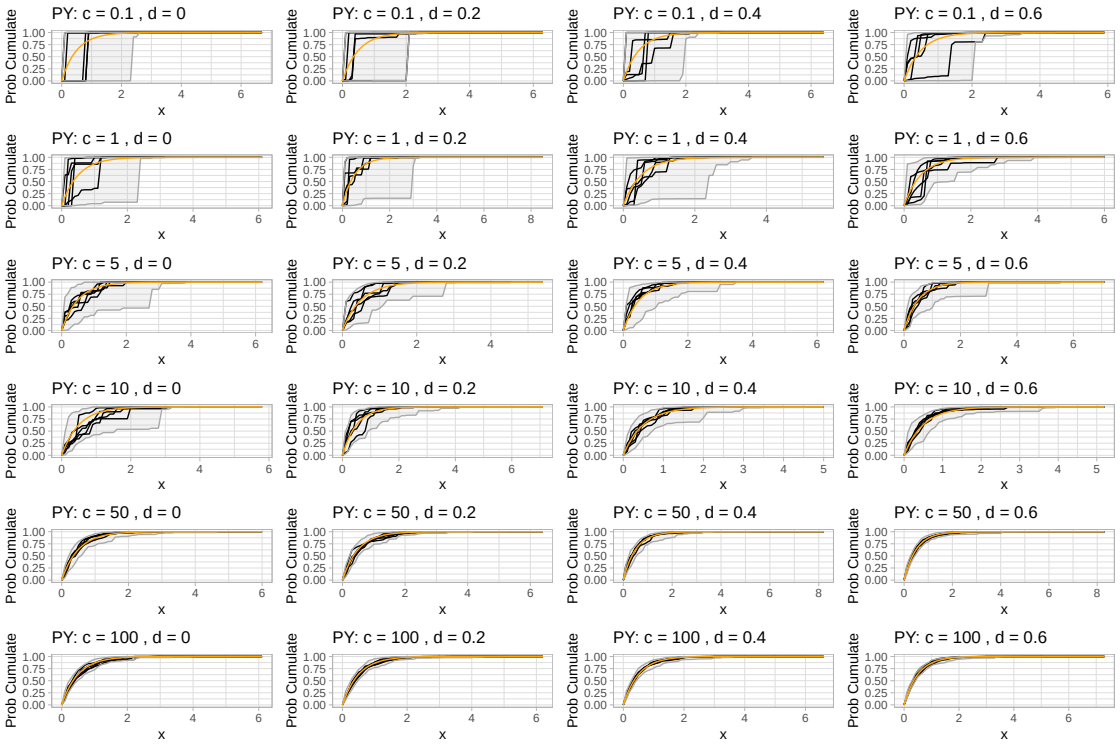

Figure 5: ECDF simulate

The ECDF plots (Figure 5) further illustrate the effect of the parameters. As c and d increase, the variability of the process decreases. In fact, the realizations of $\tilde{P}$ become concentrated around the base measure G_0.

4 Properties

Let $\tilde{P} \sim PY(c, d, G_0)$ be defined on the space $\mathbb{X}$, and let $A \subseteq \mathbb{X}$ be a measurable subset of $\mathbb{X}$. Then:

$$E[\tilde{P}(A)] = G_0(A),$$

$$\text{Var}[\tilde{P}(A)] = G_0(A) \cdot (1 - G_0(A)) \cdot \frac{1-d}{c+1}.$$

The expected value of the Pitman-Yor process, as in the case of the Dirichlet process, coincides with the base measure evaluated on a measurable subset of $\mathbb{X}$. In contrast, the variability in the case of the Pitman-Yor process is controlled by two parameters: the concentration parameter c and the penalization parameter d. As observed previously, it can be stated that the variability of the process tends to decrease as c increases and/or for high values of d approaching one.

5 Predictive Distribution

Let $X_i \mid \tilde{P} \overset{\text{iid}}{\sim} \tilde{P}$ with $\tilde{P} \sim PY(c, d, G_0)$. The predictive distribution of X_{n+1} given the observations $X_1, \ldots, X_n$ is:

$$P(X_{n+1} \in A \mid X_1, \ldots, X_n) = \frac{c + k \cdot d}{n + c} \cdot G_0(A) + \sum_{j=1}^{k} \frac{n_j - d}{n + c} \cdot \delta_{X_j^*}(A),$$

for every $A \subseteq \mathbb{X}$, a measurable subset of $\mathbb{X}$, where $\{X_1^*, \ldots, X_k^*\}$, with $k \leq n$, denote the unique values in the set $\{X_1, \ldots, X_n\}$ with frequencies $\{n_1, \ldots, n_k\}$. From this expression, it follows that the joint distribution of $\{X_1, \ldots, X_n\}$ can be defined starting from the generalization of the Polya urn scheme, such that:

$$X_1 \text{ is sampled from } G_0, \quad \text{for} \quad i = 2, \ldots, n$$

$$X_i \mid X_1, \ldots, X_{i-1} \sim \begin{cases} \text{new value sampled from } G_0, & \text{with probability } \dfrac{c + k \cdot d}{i - 1 + c}, \\ \text{observed value } X_j^*, & \text{with probability } \dfrac{n_j - d}{i - 1 + c}. \end{cases}$$

In a Pitman-Yor process, the probability of observing a new observation depends not only on the concentration parameter c, but also on the penalization parameter d and the number of distinct observations generated previously. In fact, the penalization coefficient d affects both the probability of generating new clusters and the growth of those already existing. The higher the values of d and c, the greater the probability of generating new distinct clusters, while simultaneously reducing the probability of populating the already existing clusters. Therefore, unlike the Dirichlet process, where the generated clusters tend to have a similar size on average, in the Pitman-Yor process, there will be many sparse clusters and a few that are densely populated.

6 Pitman-Yor Mixture Models

The Pitman-Yor process is useful to apply when it is assumed that the distribution of the observations $X_1, \ldots, X_n$ is discrete. If the observations follow a continuous distribution, it cannot be applied directly for density estimation problems. For this reason, so-called mixture models are introduced.

A Pitman-Yor process mixture on $\mathbb{X}$ is a random PDF $\tilde{f}$ defined as:

$$\tilde{f}(x) = \int_\Theta K(\theta, x) d\tilde{P}(\theta)$$

where $\tilde{P} \sim PY(c, d, G_0)$ and G_0 is a probability measure on Θ. This model can also be rewritten in its hierarchical form:

$$\begin{aligned} X_i|\theta_i &\overset{\text{ind}}{\sim} K(X_i, \theta_i); \\ \theta_i|\tilde{P} &\overset{\text{iid}}{\sim} \tilde{P}; \\ \tilde{P} &\sim PY(c, d, G_0). \end{aligned}$$

The goal is to estimate the density of the distribution of the observations $\boldsymbol{X} = (X_1, \ldots, X_n)$ through:

$$\hat{f} = E[\tilde{f}(x)|\boldsymbol{X}].$$

This quantity can be estimated using the Monte Carlo method. Let $\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(M)}$ be a set of M realizations from the distribution $\boldsymbol{\theta}|\boldsymbol{X}$, where $\boldsymbol{\theta}^{(m)} = (\theta_1^{(m)}, \ldots, \theta_n^{(m)})$ for each $m = 1, \ldots, M$, then:

$$\hat{f}(x) \approx \frac{1}{M} \sum_{m=1}^{M} E\left[\tilde{f}(x) \mid \boldsymbol{X}, \boldsymbol{\theta}^{(m)}\right] \tag{1}$$

where, conditionally on the auxiliary variables $\boldsymbol{\theta}$:

$$E[\tilde{f}(x)|\boldsymbol{X}, \boldsymbol{\theta}] = \frac{c + d \cdot k}{c + n} \int_\Theta K(X, \theta) g_0(\theta)\, d\theta + \sum_{j=1}^{k} \frac{(n_j - d)}{c + n} K(X; \theta_j^*) \tag{2}$$

To sample from the conditional distribution of $\boldsymbol{\theta}|\boldsymbol{X}$, the Gibbs sampling algorithm is used, where the full

conditional distributions for θ_i conditioned on $\boldsymbol{\theta}_{(-i)} = (\theta_1, \ldots, \theta_{i-1}, \theta_{i+1}, \ldots, \theta_n)$ and $\boldsymbol{X} = (X_1, \ldots, X_n)$ are defined as follows:

$$P(\theta_i \in d\theta | \boldsymbol{X}, \boldsymbol{\theta}_{(-i)}) \propto \frac{c + dk_{(-i)}}{c + n - 1} \int_{\Theta} K(X_i, \theta) g_0(\theta)\, d\theta \cdot \frac{K(X_i, \theta) \cdot g_0(\theta)}{\int_{\Theta} K(X_i, \theta) \cdot g_0(\theta)\, d\theta} \tag{3}$$
$$+ \sum_{j=1}^{k_{(-i)}} \frac{(n_{j(-i)} - d)}{c + n - 1} K(X_i; \theta^*_{j(-i)}) \delta_{\theta^*_{j(-i)}}(\theta).$$

where g_0 is the density function associated with the probability measure G_0, and $k_{(-i)}$ is the number of distinct values in $\boldsymbol{\theta}_{(-i)}$ with frequencies $(n_{1(-i)}, \ldots, n_{k_{(-i)}})$.

7 Location Scale Pitman-Yor Model Simulation

Consider the Location Scale Pitman-Yor model with a Gaussian kernel defined as follows:

$$\begin{aligned} X_i \mid \theta_i, \sigma^2 &\overset{\text{ind}}{\sim} N(\theta_i, \sigma^2) \qquad i = 1, \ldots, n \\ \theta_i \mid \tilde{P} &\overset{\text{iid}}{\sim} \tilde{P} \qquad i = 1, \ldots, n \\ \tilde{P} &\sim PY(c, d, G_0) \\ G_0 &\sim N(0, 1) \\ \sigma^2 &\sim Inv - Gamma(a, b) \end{aligned}$$

with $c > -d, d \in [0, 1], a > 0$ and $b > 0$.
In the following section, we report the results of applying this model to a simulated dataset of $n = 1000$ independent and identically distributed observations generated from the following mixture of Normal distributions:

$$0.3 \cdot N(y, 0, 1) + 0.6 \cdot N(y, 3, 1) + 0.1 \cdot N(y, 7, 1).$$

The posterior mean density $E[\hat{f}(x)|\boldsymbol{X}]$ was estimated using a marginal algorithm articulated in two steps:
STEP 1: Simulation of the vector $\boldsymbol{\theta}$ from $P(\theta_i \mid \boldsymbol{X}, \boldsymbol{\theta}_{-i})$, Eq.(3), the full conditional estimated using the Gibbs Sampling algorithm with 100 iterations and a burn-in of 10% of the total iterations.

STEP 2: Estimation of $E[\hat{f}(x)|\boldsymbol{X}]$ via Monte Carlo by combining equations (1) and (2).

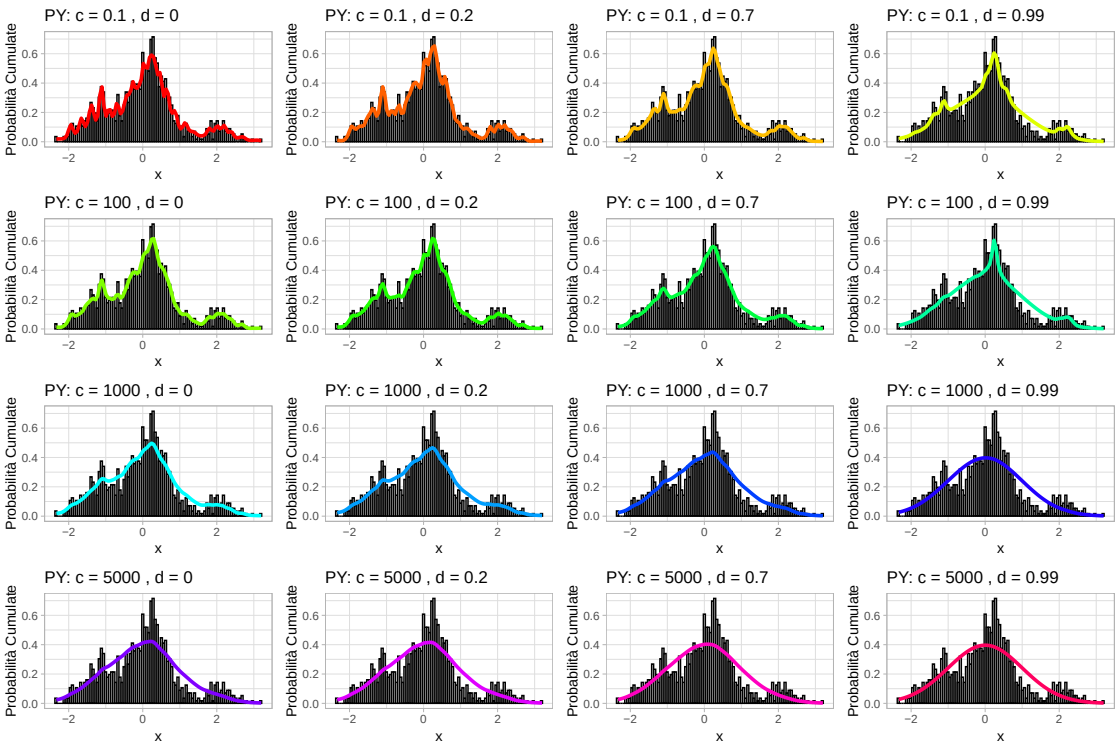

Figure 6: Mixture Models on Simulated Data Varying with Parameters

The goal of the analysis is to study how the posterior mean density changes as the parameters c and d vary. From the plots in Figure 6, it is evident that for low values of c and d, the estimated density function perfectly fits the data, capturing the peaks and fluctuations of the mixture itself. This arises because, for small values of the parameters, the impact of the prior guess is almost negligible, as shown in Equation (2). In fact, in this scenario, the posterior density is mainly sampled from the empirical component:

$$\sum_{j=1}^{k} \frac{(n_j - d)}{c + n} K(X; \theta_j^*)$$

thus obtaining a good approximation to the mixture.

On the other hand, as c and d increase, the posterior mean density becomes progressively smoother. When the prior guess component becomes more dominant than the empirical one, the posterior mean density is mainly sampled from the prior component:

$$\frac{c + d \cdot k}{c + n} \int_{\Theta} K(X, \theta) g_0(\theta)\, d\theta$$

resulting in a posterior density that closely approximates the base measure.

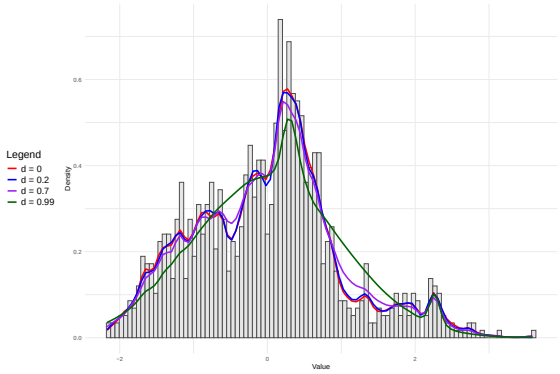

Figure 7: $c = 100$

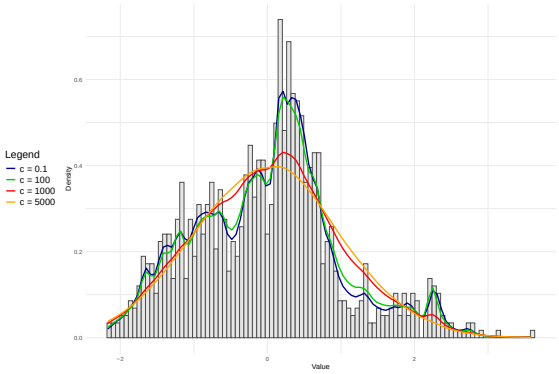

Figure 8: $d = 0.7$

Figure 9: Mixture Models on Simulated Data Using the Marginal Algorithm

In conclusion, Figure 9 further illustrates in more detail the role of the parameters c and d in modeling the level of smoothness of the estimated density function.

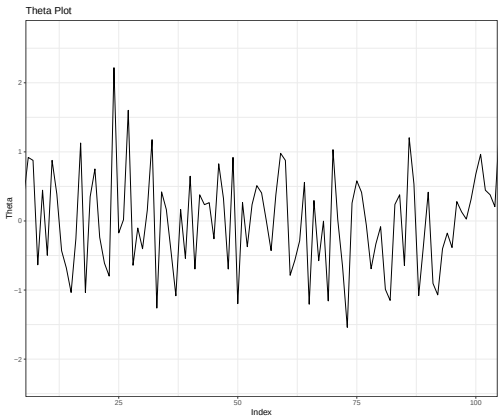

Figure 10: Trace plot parameters θ

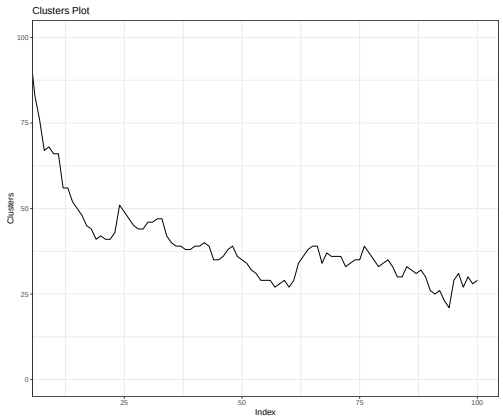

Figure 11: Trace plot number of clusters

Figure 12: Trace plot

Figure 12 allows monitoring the convergence of the generation algorithm. It reveals some issues with the convergence of the cluster number estimate, as the trace plot does not stabilize around a constant mean value but instead continues to decrease. Regarding the parameter vector $\boldsymbol{\theta}$, the algorithm's performance is discrete. In conclusion, the density estimates obtained are acceptable, but not fully satisfactory. One solution for achieving more accurate estimates could be to increase the number of iterations and the burn-in period. Using the manually implemented Gibbs Sampling algorithm, increasing the number of iterations leads to excessive computational time. For this reason, the *PYdensity* function from the *BNPmix* package was used to estimate the mixture density, considering 1000 iterations and a burn-in of 100. The *BNPmix* package offers three different methods for updating the θ values in the MCMC chain. Specifically, the method used is the MAR (Marginal Sampler) method.

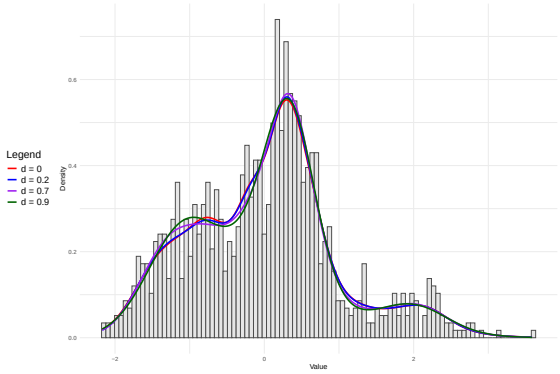

Figure 13: $c = 100$

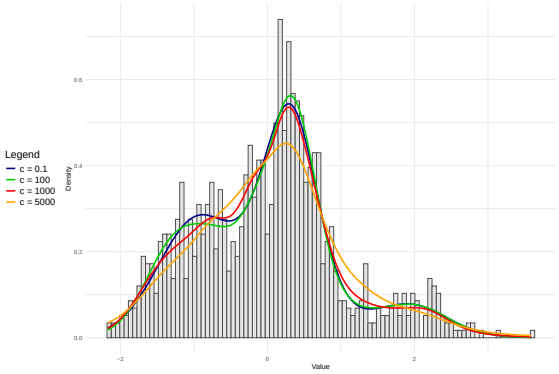

Figure 14: $d = 0.7$

Figure 15: Density simulated with *BNPmix*

By comparing the results in Figure 9 with those in Figure 15, it can be observed that the densities estimated using the *BNPmix* package are, in general, much smoother. Additionally, for computational reasons, the maximum value for d was chosen to be 0.9 rather than 0.99 as before. In general, the trend of the density estimates using *BNPmix* as d changes is much more uniform and less sensitive to changes in the parameter. In fact, the curves with $d = 0$, shown in red, and $d = 0.2$, shown in blue, in Figure 13, are practically indistinguishable.

7.1 Estimation of the Number of Clusters

The *BNPmix* package, after estimating the parameters θ, uses three different similarity-based methods to identify the number of clusters. The first two methods employ an hclust algorithm with average or complete linkage, while the third calculates the similarity matrix $(\hat{P})$ to find the optimal partition from which the number of clusters is extracted, as defined below:

$$\hat{P} = (\hat{p}_{ij}), \quad i, j = 1, \ldots, n$$

$$\hat{p}_{ij} = \frac{1}{R - R_0} \sum_{r=R_0+1}^{R} I_{\{\theta_i^r = \theta_j^r\}}$$

where R is the number of iterations and R_0 is the burn-in number.

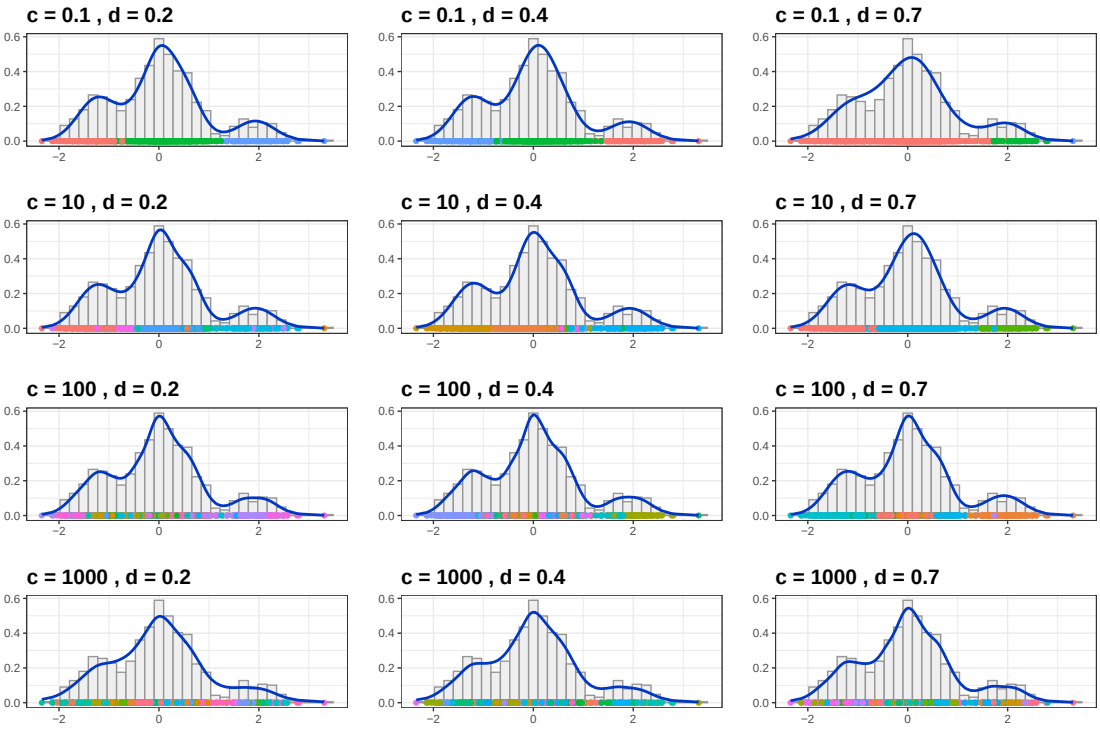

Figure 16: Number of Clusters as a Function of Parameters c and d

Given $n = 1000$, for values of c equal to 0.1 and 10, the effect of the prior on the density estimate, and thus on the estimate of the clusters, is almost negligible, regardless of the value of d. The parameter d primarily influences the empirical part

$$\sum_{j=1}^{k} \frac{(n_j - d)}{c + n} K(X; \theta_j^*)$$

reducing the probability of populating already existing clusters with small cardinality. This is reflected in the graph (Figure 16) as, with increasing d, the number of clusters seems to decrease. On the other hand, when c starts to approach n, the number of clusters increases, an effect that tends to amplify as d increases.

7.1.1 Deep Dive into Cluster Number Prediction

Given the number of clusters in the reference population of size n, it is possible to estimate the number of clusters a posteriori present in a new population of size m.

In the case of a Pitman-Yor process, the form of the Exchangeable Partition Probability Function (EPPF) is as follows:

$$\Pi_n(n_1, ..., n_k) = V_{n,k} \prod_{j=1}^{k} (1 - \sigma)_{n_j - 1},$$

where the non-negative weights $\{V_{n,k} : n \geq 1, 1 \leq k \leq n\}$ satisfy the following recursive relation $V_{n,k} = (n - \sigma k)V_{n+1,k} + V_{n+1,k+1}$, and are of the form:

$$V_{n,k} = \frac{\prod_{i=1}^{k-1}(c + id)}{(c + 1)_{n-1}},$$

with $d \in [0, 1)$ and $c > -d$.

The notation $(1-\sigma)_{n_j-1}$ is a Pochhammer symbol, which is equivalent to the following ratio: $\Gamma(n_j - \sigma)/\Gamma(1-\sigma)$.

Based on the EPPF, an explicit expression for the distribution of the number of new distinct clusters observed in a new additional sample, $K_m^{(n)}$, conditionally on the information provided by $X_1, \ldots, X_n$, is given by:

$$P(K_m^{(n)} = j | X_1, \ldots, X_n) = \frac{V_{n+m,k+j}}{V_{n,k}} \frac{\mathcal{C}(m, j; d, -n + kd)}{d^j},$$

where $X_1, \ldots, X_n$ are partitioned into $K_n = k$ clusters with frequencies $n_1, \ldots, n_k$, and $\mathcal{C}(m, j; d, -n + kd)$ is the generalized non-central factorial coefficient:

$$\mathcal{C}(m, j; d, -n + kd) = (j!)^{-1} \sum_{r=0}^{j} (-1)^r \binom{j}{r} (n - d(r+k))_m.$$

All of this is used to determine the non-parametric Bayesian estimator, considering a quadratic loss function, which is given by:

$$\hat{K}_m^{(n)} = E[K_m^{(n)} | K_n = k, N_n = \mathbf{n}],$$

with $\mathbf{n} = (n_1, \ldots, n_k)$. In the case of the PY process, with parameters (c, d), it becomes:

$$P(K_m^{(n)} = j | K_n = k, N_n = \mathbf{n}) = \frac{(c/d + k)_j}{(c+n)_m} \mathcal{C}(m, j; d, -n + kd).$$

The estimator is then given by:

$$\hat{K}_m^{(n)} = \left(k + \frac{c}{d}\right) \left(\frac{(c+n+d)_m}{(c+n)_m} - 1 \right).$$

8 Application to a Real Dataset

In conclusion, the location-scale Pitman-Yor model has been applied to a real dataset. The dataset analyzed contains data on concrete, specifically its composition and actual compressive strength. Indeed, within the mix, there are several ingredients including cement, water, fine aggregate (such as sand), and coarse aggregate (rock fragments). The dataset consists of 1030 observations with no missing values. The variable of interest was considered to be the amount of coarse aggregate in kg per cubic meter of concrete. In order to estimate the target density, both the manually implemented marginal algorithm, the results of which are shown in Figure 19, and the *BNPmix* package were used, as seen in the results of Figure 20. For the marginal procedure, 100 iterations were considered, while for the *PYdensity* function, 1000 iterations were examined. In both cases, the burn-in period was set to 10% of the generated realizations.

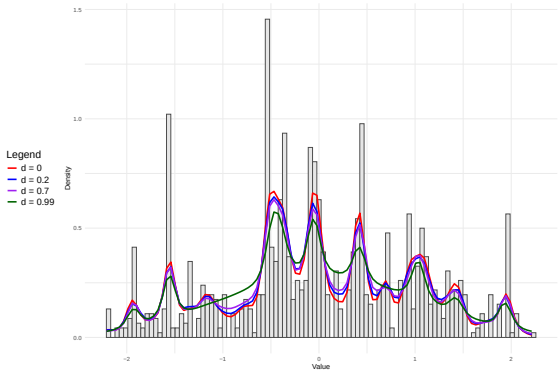

Figure 17: $c = 100$

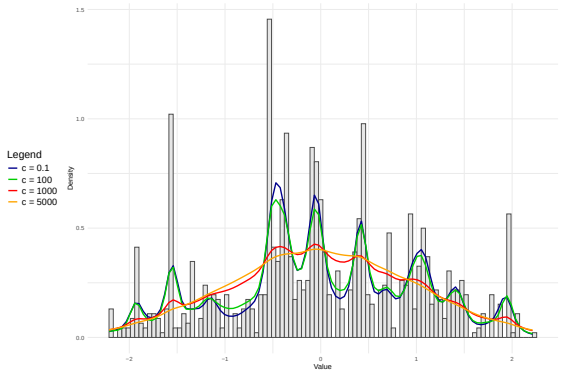

Figure 18: $d = 0.7$

Figure 19: Mixture Models of PY Estimated with the Marginal Algorithm as the Parameters Vary

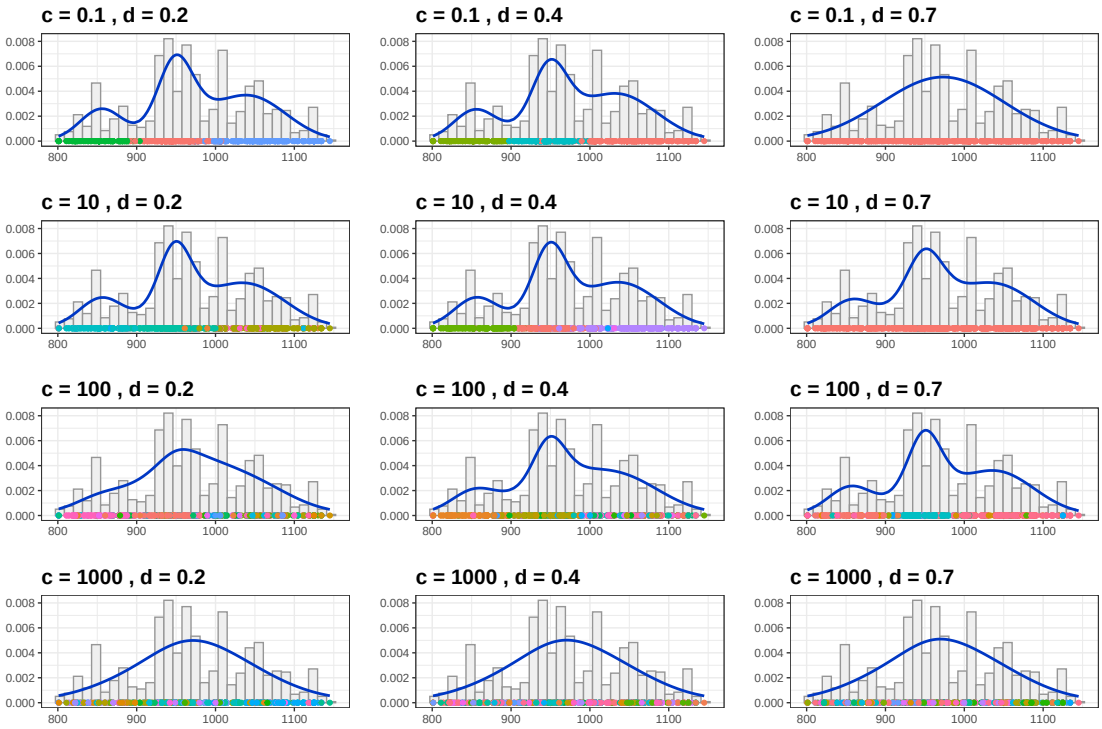

Figure 20: Number of Clusters as the Parameters c and d Vary with the BNPmix Package

Again, as shown in the figures, we can draw the same conclusions as before regarding the smoothness of the estimated densities as the parameters c and d vary. In fact, it seems that for low values of concentration and penalization, the data were generated from three mixtures, which represent the adoption of three different types of concrete aggregate.

Bibliography

Canale, A., Corradin, R. & Nipoti, B. (2022a). Importance conditional sampling for Pitman-Yor mixtures. *Statistics and Computing* **32(3)**, 40.

Canale, A., Corradin, R. & Nipoti, B. (2022b). Package 'BNPmix'. `https://cran.r-project.org/web/packages/BNPmix/BNPmix.pdf`.

Dassios, A. & Zhang, J. (2023). Exact simulation of Poisson-Dirichlet Distribution and Generalised Gamma process. *Methodology and Computing in Applied Probability* **25**.

De Blasi, P., Favaro, S., Lijoi, A., Mena, R., Prünster, I. & Ruggiero, M. (2015). Are Gibbs-type priors the natural generalization of the Dirichlet process? *IEEE Transactions Pattern Analysis and Machine Intelligence* **37(2)**, 212–229.

Ferguson, T. S. (1973). A Bayesian analysis of some Nonparametric problems. *The Annals of Statistics* **1(2)**, 209 – 230.

Ishwaran, H. & James, L. F. (2001). Gibbs Sampling methods for Stick-Breaking priors. *Journal of the American Statistical Association* **96(453)**, 161–173.

Orbanz, P. (2014). Lecture Notes on Bayesian Nonparametrics .

Pitman, J. & Yor, M. (1997). The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *The Annals of Probability* **25(2)**, 855 – 900.

Shanawad, V. (2021). Concrete compressive strength data. `https://www.kaggle.com/datasets/vinayakshanawad/cement-manufacturing-concrete-dataset`.