Contents

Introduction

The Bayesian approach to inference treats the parameter as a random variable, as opposed to the frequentist approach in which a true value for the parameter is assumed to exist. For example, suppose we are given a sample of observations $\boldsymbol{X} = X_1, ..., X_n$ drawn from a sampling model $p(\boldsymbol{X}|\theta)$. By encoding the prior belief of the parameter taking value θ into a probability distribution $p(\theta)$ over the parameter space Θ, the Bayesian approach involves updating the information on the parameter by using the data $\boldsymbol{X}$. This is done by deriving the optimal posterior distribution $p(\theta|\,\boldsymbol{X})$, which is given by the Bayes rule:

$$p(\theta|\boldsymbol{X}) = \frac{p(\boldsymbol{X}|\,\theta)\,p(\theta)}{\int_\Theta p(\boldsymbol{X}|\tilde{\theta})\,p(\tilde{\theta})\,d\tilde{\theta}}.$$

The result is a whole probability distribution, that can be summarized by point estimates with associated credible intervals. To do so, the posterior calculation is included into a decision theoretic framework: a loss function $L(\hat{\theta}, \theta)$ quantifying the loss of estimating θ with $\hat{\theta}$ is specified and the point estimate is then derived by minimizing the expected posterior loss.

This work will focus on a particular subset of Bayesian models whose key parameter to be inferred is a partition π_n over the observations, where a partition is defined in its mathematical meaning as a grouping of observations into non-empty subsets such that every observation belongs to exactly one of these subsets. The task of learning such a partition is commonly known as clustering. After an introduction to the theoretical foundations of Random Partition Models, to which the first chapter is devoted to, the

second chapter will cover an application to experimental mouse-tracking data. The code to replicate this analysis can be found in the linked Github repository. Most of the code in the repository for both the descriptive and modelling part was written in R, except for the implementation of the adopted MCMC sampling scheme. That part requires an higher computational effort compared to the rest of the analysis and was written in C++ for performance reasons, by taking inspiration from the original implementation of the algorithm introduced in [14].

The work is the result of a 10-weeks stay at the University of California, Irvine and was developed under the joint supervision of Prof. Antonio Lijoi and Prof. Michele Guindani.

Chapter 1

Theoretical Background

This chapter will discuss the theoretical foundations of Random Partition Models. These are clustering models that specify a probability distribution over the space of possible partitions, whose development is mainly due to Bayesian nonparametric methods. In particular, in this chapter, I will motivate and introduce the Dirichlet process [4], the main nonparametric prior that can be used to specify such a model. I will present the main definition of the Dirichlet process, as well as alternative constructions that show up frequently in applications. This will lay the foundation to discuss the Location-Scale Partition Distribution [14]. It is a probability model on the space of partition that relies on a modification of the model induced by the Dirichlet process and allows for the specification of a centering partition. I will then conclude by discussing an algorithm that can be used to sample from the posterior of a model that uses the LSP distribution as a prior.

Before stating further theoretical results, a preliminary definition is needed. Indeed, we will see that the Dirichlet process is a way to specify a probability distribution over probability distributions. Formally we say that the Dirichlet process is a random probability measure.

Definition. (Random probability measure) Any random element $\tilde{p}$ from some probability space $(\Omega, \mathcal{F}, \mathbb{P})$ and with values in $(P_X, B(P_{\mathbb{X}})$ is a random probability measure.

1.1 Exchangeability.

Definition. Let $\mathbb{X}$ be a separable and complete metric space. A sequence of $\mathbb{X}$-valued random elements $(X_n)_{n\geq 1}$ is exchangeable if for any $n \geq 1$ and permutation σ of $(1, ..., n)$ one has

$$(X_1, ..., X_n) \stackrel{d}{=} (X_{\sigma(1)}, ..., X_{\sigma(n)}).$$

In other other words, $(X_1, ..., X_n)$ are exchangeable if the joint distribution of the random variables is invariant to permutations of the elements. The notion of exchangeability is at heart of a fundamental theorem in Bayesian statistics, the representation theorem by De Finetti.

Theorem (De Finetti). A sequence of random elements $X = (X_n)_{n\geq 1}$ taking values in a separable and complete metric space X are exchangeable if and only if there exists a probability measure Q on $(\mathcal{P}_{\mathbb{X}}, \mathcal{B}(\mathcal{P}_{\mathbb{X}}))$ such that

$$\mathcal{P}[X \in A] = \int_{\mathcal{P}_{\mathbb{X}}} \prod_{i=1}^{n} p(A_i) Q(dp) \quad \forall A \in \mathcal{B}(\mathbb{X}^{\infty}). \tag{1.1}$$

Therefore, De Finetti's theorem has a crucial role in justifying the Bayesian approach to inference. It states that any exchangeable sequence can be written as a mixture of independent and identically distributed sequences. Indeed, in virtue of de Finetti's theorem, the exchangeability assumption allows to characterize a sequence of random variables as

$$\begin{aligned} X_i|\tilde{p} &\stackrel{iid}{\sim} \tilde{p}, \qquad i = 1, \ldots, n, \\ \tilde{p} &\sim Q. \end{aligned} \tag{1.2}$$

In the case of a parametric model, Q is defined on a finite-dimensional subspace of $\mathcal{P}_{\mathbb{X}}$. On the other hand, the model is then called nonparametric if Q is defined on an infinite dimensional space.

1.2 Dirichlet process.

The Dirichlet process [4] is one of the most popular Bayesian nonparametric prior. It is characterized by two parameters: a base distribution G_0 and a concentration constant α.

Definition. Let $\alpha > 0$ and G_0 be a probability measure defined on S. A Dirichlet Process with parameters (α, G_0) is a random probability measure G defined on S which assigns probabilities to every measurable set B such that for every measurable finite partition $\{B_1, .., B_k\}$ of S,

$$(G(B_1), ..., (G(B_k)) \sim Dir(\alpha G_0(B_1), ..., \alpha G_0(B_k)), \tag{1.3}$$

where a random vector $\boldsymbol{X} = (X_1, ..., X_n) \sim Dir(\alpha_1, ..., \alpha_n)$ if it follows a Dirichlet distribution with parameters $(\alpha_1, ..., \alpha_n)$ and has joint density

$$f_{\boldsymbol{X}}(\boldsymbol{x}) = \frac{\Gamma(\sum_{i=1}^{n} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} x_1^{\alpha_1 - 1} ... x_{k-1}^{\alpha_{k-1} - 1} (1 - |\boldsymbol{x}|)^{\alpha_k - 1} \mathbb{1}_{\mathcal{S}_{k-1}}(\boldsymbol{x}), \tag{1.4}$$

with $\Gamma(z) = \int_0^\infty x^{z-1} e^{-x} dx$, the gamma function.

The Dirichlet distribution is the multivariate generalization of the Beta distribution and has support on the set $\mathcal{S}_{k-1}$, the $(k-1)$-dimensional simplex, that is:

$$\mathcal{S}_{k-1} = \{\boldsymbol{x} \in \mathbb{R}^k : x_0 + ... + x_{k-1} = 1,\ x_i \geq 0 \text{ for } i = 1, ..., k-1\}.$$

This means that realizations of the Dirichlet process are indeed discrete probability distributions with probability one.

1.2.1 Prior conjugacy.

The Dirichlet process can be used as nonparametric prior Q in the framework of described by (1.2). This choice is particularly convenient from a computational point of view, as the posterior distribution resulting from it has a closed form. Indeed, the Dirichlet process is a conjugate prior. In general, in Bayesian inference a prior is said to be conjugate to a specific likelihood function if the resulting posterior belongs to same family as the prior. In the case of model (1.2), where Q is chosen as a Dirichlet process with parameters (α, G_0), we have:

$$\tilde{p}|X_1, ..., X_n \sim DP\left(\alpha + n, \frac{\alpha}{\alpha + n} G_0 + \frac{n}{\alpha + n} \hat{G}_n\right), \tag{1.5}$$

where $\hat{G}_n = n^{-1} \sum_{i=1}^{n} \delta_{X_i}$ is the empirical distribution.

1.2.2 Dirichlet mixture models.

As a further steps, mixture models can also be defined. The resulting class of models are the so-called Dirichlet mixture models and they are particularly interesting for their applications to clustering problems. The general hierarchical representation of Dirichlet process mixture models is as follows [9]:

$$\begin{aligned} X_i|\theta_i &\overset{ind}{\sim} f_{\theta_i}, \\ \theta_i|Q &\overset{iid}{\sim} Q. \end{aligned} \tag{1.6}$$

This representation is a particular specification of model (1.2) in which latent variables θ_i specific to each experimental unit are introduced and the distribution f is an arbitrary distribution specified by θ_i. The discrete nature of the DP implies a positive probabilities of ties among the θ_i, thus inducing a partition of the sample. A common choice for f is

the normal distribution $\mathcal{N}(\mu_i, \sigma_i)$ with $\theta_i = (\mu_i, \sigma_i)$, although several other distributions have been used, depending on the application. For example, f can be chosen to be itself a Dirichlet process, defining what's commonly known as Hierarchical Dirichlet process. Such a model is also a generalization of the Latent Dirichlet allocation [3], that has extensive applications to clustering problems in various domains, particularly for topic modelling in Natural Language Processing.

1.3 Stick-breaking construction.

The stick-breaking construction [13] is a representation of the Dirichlet process that exposes the almost sure discreteness of its realizations, by representing it as the sum of discrete point masses. Indeed, if $\tilde{p} \sim DP(\alpha, G_0)$, then:

$$\tilde{p} = \sum_{i \geq 1} \pi_i \delta_{\xi_i}, \tag{1.7}$$

with $\pi_1 = V_1$ and $\pi_i = V_i \prod_{r=1}^{i-1} (1 - V_r)$, $\forall i \geq 2$, for a sequence $(V_i)_{i \geq 1}$ of i.i.d. random variables from a $Beta(1, \alpha)$ distribution and $\epsilon_i \overset{iid}{\sim} G_0$ independent from $(\pi_i)_{i \geq 1}$. In this construction, the i-th weight π_i can be thought as a fraction of $\{1 - \sum_{j \leq i} \pi_j\}$, that is, a fraction of what is left after the preceding point masses. This is where the name of the construction originates from, since this procedure of weight assignment can be viewed as sequentially breaking a stick of initial length equal to 1.

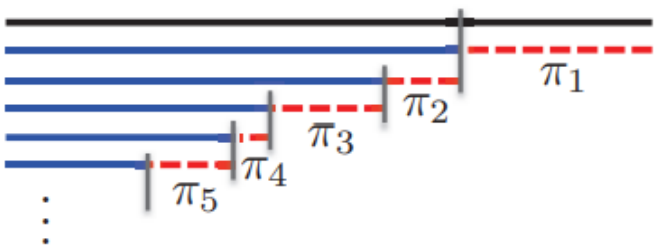


Figure 1.1: Visual representation of the stick breaking construction [5]

1.4 Polya urn model.

As we have seen in the formulation of model 1.6, the discrete nature of samples from a Dirichlet process induces a partition because it implies a positive probability of ties among the latent variables θ_i.

The model presented in this section is the following variant of the Polya urn model, also called the Blackwell-Macqueen urn: imagine drawing one random ball at a time with replacement from an urn initially containing α black balls. Moreover, at every draw if the ball is black, we insert a new ball whose color is new inside the urn and sampled from a probability distribution G_0. On the other hand, if the ball is of any other color, we add a new ball of the same color as the ball we just drew.

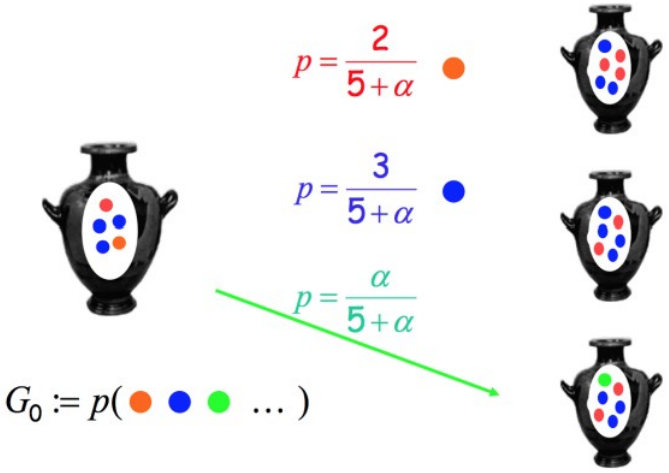


Figure 1.2: Polya urn visualization [1]

Therefore, representing colors with their order of first appearance inside the urn, the probability of draw i resulting in color j conditioned on the past draws is given by:

$$p(s_i = j | s_1, ..., s_{i-1}) = \begin{cases} \frac{n_{i-1,j}}{\alpha+i-1} & \text{for } j = 1, ..., k_{i-1} \\ \frac{\alpha}{\alpha+i-1} & \text{if } j = k_{i-1}+1 \end{cases}, \tag{1.8}$$

where s_i is the color of the i-th draw and k_{i-1} is the last new color drawn up to draw

$i-1$. This construction is particularly interesting for its link with the Dirichlet process. Indeed, Blackwell & MacQueen [2] proved the almost sure convergence of the Polya urn scheme to the Dirichlet process as $n \to \infty$. This means that a sequence of θ_i resulting from a Dirichlet process prior can be approximated by a sequence generated from a Polya urn scheme. As a consequence, this scheme is widely adopted as a sampling strategy for the Dirichlet Process in inference and clustering problem.

From this model, we can derive the predictive distribution for the θ_i, that will be useful to describe the LSP distribution in the following section. Let $\theta^*_{i,j}$ be the j-th unique value among $\{\theta_1, .., \theta_i\}$. Noting that $s_i = j$ implies $\theta_i = \theta^*_{i-1,j}$ and $s_i = k_{i-1} + 1$ implies that $\theta_i \sim G_0$, we can state the conditional distribution for θ_i as:

$$\theta_i|\theta_1, ..., \theta_{i-1} \sim w_0 G_0(\theta_i) + \sum_{k=1}^{K^{(i)}} w_k \delta_{\theta^*_k}(\theta_i), \tag{1.9}$$

with

$$w_0 = \frac{\alpha}{\alpha + i - 1} \quad \text{and} \quad w_k = \frac{n_{i-1,k}}{\alpha + i - 1},$$

$i = 1, \ldots, n$. To complete the analogy with the Polya urn scheme, here w_0 represents the probability of θ_i forming a new cluster while w_k is the probability of θ_i joining the pre-existing cluster k.

1.4.1 Expected number of clusters.

An interesting property that has to do with the number of clusters generated by a Dirichlet mixture models is derived by resorting to the Polya urn scheme. Indeed, this class of models differs from other clustering methods such as K-means in that it does not require the number of clusters to be fixed a priori and, as a consequence, such a number is free to grow as the sample size grows. Hence, one may wonder if an estimate for the expected number of clusters can be provided. The predictive density described by equation 1.9 is

used to derive the growth rate of the expected number of clusters in a mixture model with a Dirichlet process prior. Starting from such equation and considering the sequential generation of θ_i, $\forall i \in \{1, ..., n\}$, the event of forming a new cluster at draw i is described by a Bernoulli trial $W_i \sim Bern\left(\frac{\alpha}{\alpha+i-1}\right)$, where $W_i = 1$ if observation i formed a new cluster, $W_i = 0$ otherwise. Therefore, by letting $K_n = \sum_{i=1}^{n} W_i$ be the number of clusters after n draws, we have

$$\mathbb{E}K_n = \sum_{i=1}^{n} \mathbb{E}W_i = \sum_{i=1}^{n} \frac{\alpha}{\alpha + i - 1},$$

from which we deduce that as $n \to \infty$

$$\mathbb{E}K_n \approx \alpha \log\left(1 + \frac{n}{\alpha}\right), \tag{1.10}$$

which shows that as sample size tends to infinity, the expected number of cluster diverges and is approximately logarithmic with the sample size.

1.5 Chinese restaurant process.

A stochastic process closely related to the Polya urn scheme and the Dirichlet process is the Chinese restaurant process. This process is a way to derive the model on the space of partitions induced by the Dirichlet process and takes the name from a metaphor that describes its evolution as a sequence of customers entering a chinese restaurant. Upon entering, the first customer sits at an emptpy table. Following that, customer i $\forall i \geq 2$, either joins an existing table with probability proportional to its size or starts a new table, with probability proportional to a parameter α. In particular, the evolution of the process is described by the following probabilities:

$$\begin{aligned} \mathbb{P}(\text{i joins table c}|\pi_{i-1}) &= \frac{|c|_{i-1}}{\alpha + i - 1} \\ \mathbb{P}(\text{i starts a new table}|\pi_{i-1}) &= \frac{\alpha}{\alpha + i - 1} \end{aligned} \tag{1.11}$$

Denoting the customers as experimental units and the tables as clusters, we can see that this process induces a model over the partitions such that the probability of observing a partition with k cluster is equal to:

$$\frac{\alpha^k}{(\alpha)_n},$$

where $(\alpha)_n = \alpha(\alpha+1)...(\alpha+n-1)$. Each of these partitions have the same probability, equal to:

$$\frac{1}{k!}\binom{n}{n_1...n_k}\frac{\alpha^k}{(\alpha)_n}\prod_{j=1}^{k}(n_j-1)!$$

From this, we can see that the probability of a given partition $\pi_n = (g_1, ..., g_k)$, where $g_i = j$ if customer i belongs to table j, k is the number of clusters is equal to:

$$p(\pi_n) = \frac{\alpha^k}{(\alpha)_n}\prod_{j=1}^{k}(n_j-1)! \tag{1.12}$$

Expression (1.12) is also known as Exchangeable Partition Probability Function (EPPF). It can be shown that the partition model induced by the Dirichlet process prior (model 1.6) by letting two experimental units belong to the same cluster C_l if $\theta_i = \theta_j$, is exactly equal to the EPPF. Moreover, this model is exchangeable, as the probability is only affected by the sizes of the cluster and not by order in which they are grown.

1.6 LSP Distribution.

In this next section, I am going to move on to an alternative random partition model. The approach is going to be different, as the random partition model $p(\pi_n)$ will be specified directly, without resorting to a nonparameteric prior. This will allow the introduction of a centering partition, around which the distribution will be concentrated. we now move to consider a random partition model of the form:

$$\begin{aligned} X_i|\theta,\pi_n &\overset{iid}{\sim} p(X_i|\theta,\pi_n), \qquad i=1,\ldots,n, \\ \theta|\pi_n &\sim p(\theta_{\pi_n}), \\ \pi_n &\sim p(\pi_n). \end{aligned} \tag{1.13}$$

This framework requires direct specification of a prior on the space of partitions. A possibile probability distribution is given by [14], that described the Location-Scale partition (LSP) distribution, as part of an application in the social science. In that particular case, the distribution is introduced as a prior to formulate a demand regression model based on random partitions. The LSP Distribution is constructed as a variation of the Polya urn scheme. Its main difference with the DP-induced distribution is the adoption of a location partition $\rho_n = (s_1, ..., s_n) \in_n$ around which the distribution is centered and a scale parameter $\tau > 0$. The weights of the Polya urn are modified to include the information in these two parameters:

$$\theta_i|\theta_{<i},\rho_n,\tau \sim \; w_0(\rho_n,\tau)G_0(\theta_i) + \sum_{k=1}^{K^{(i)}} w_k(\rho_n,\tau)\delta_{\theta_k^*}(\theta_i), \tag{1.14}$$

$i = 1, \ldots, n$. The partition $\pi_n = (g_1, ..., g_n)$ is then formed by letting $g_i = k$ if $\theta_i = \theta_k^*$.
In this kind of models, the choice for $w_0(\cdot)$ and $w_k(\cdot)$ is usually dictated by ease of computation. In [14], an auxiliary probability model is defined for the elements of ρ_n such that $w_0(\cdot)$ and $w_k(\cdot)$ has closed-form expressions, by following the choices of [10] and [12]:

$$\begin{aligned} w_0(\rho_n,\tau) \equiv w_0(s_i,\tau) &= c_i \int p(s_i|\boldsymbol{\xi}) f_0(\boldsymbol{\xi}|\tau) d\boldsymbol{\xi}, \\ w_k(\rho_n,\tau) \equiv w_k(\{s_i,S_k\},\tau) &= c_i \int p(s_i|\boldsymbol{\xi}) f_0(\boldsymbol{\xi}|\tau,S_k) d\boldsymbol{\xi}, \end{aligned}$$

where s_i is the cluster associated to observation i and $S_k = \{s_j : g_j = k \text{ and } j < i\}$, that is the set of items associated to cluster k up to observation $i-1$. Being s_i a categorical

variable, a convenient choice is a Dirichlet-Categorical model:

$$\begin{aligned} p(s_i|\boldsymbol{\xi}) &= Cat(\xi_0, ..., \xi_{C^{(i)}+1}), \\ f_0(\boldsymbol{\xi}|\tau) &= Dir(\tau_1, ..., \tau_{C^{(i)}+1}), \\ f_k(\boldsymbol{\xi}|\tau, S_k) &= Dir(\tau_1^*, ..., \tau_{C^{(i)}+1}^*), \end{aligned} \tag{1.15}$$

which, exploiting conjugacy of this model, yields the closed-form expressions:

$$w_0(s_i, \tau) \propto \int Cat(\xi_1, ..., \xi_{C^{(i)}+1}) Dir(\tau_1, ..., \tau_{C^{(i)}+1}) d\boldsymbol{\xi},$$

$$w_k(\{s_i, S_k\}, \tau) \propto \int Cat(\xi_1, ..., \xi_{C^{(i)}+1}) Dir(\tau_1^*, ..., \tau_{C^{(i)}+1}^*) d\boldsymbol{\xi},$$

$$w_0(s_i, \tau) = \frac{\tau + 1(s_i = C^{(i)} + 1)}{\tau C^{(i)} + \tau + 1}, \quad w_k(\{s_i, S_k\}, \tau) = \frac{\tau + n_{S_k}^{s_i}}{\tau C^{(i)} + \tau + n_k}$$

Thus, the probability distribution of paritition a π_n, conditional to the hyperparameters ρ_n and τ, can be factored into

$$p(\pi_n|\rho_n, \tau) = \prod_{i=1}^{n} p(g_i|g_{<i}, \rho_n, \tau),$$

where

$$p(g_1) = 1 \text{ and } p(g_i|g_{<i}, \rho_n, \tau) = \begin{cases} w_k(\rho_n, \tau) \text{ if i in cluster k} \\ w_0(\rho_n, \tau) \text{ if i starts a new cluster} \end{cases}$$

1.6.1 Posterior inference via Markov Chain Monte Carlo.

Deriving the posterior distribution for the partition $p(\pi_n, \boldsymbol{\theta}|\boldsymbol{x})$ in the model specified by equations 1.13 and 1.14 is analytically intractable. However, such a distribution can be approximated by sampling from it through Markov Chain Monte Carlo methods. These methods are extensively used for posterior sampling in the context of Bayesian models and are based on the idea of generating a Markov chain that has the posterior distribution

of interest as its stationary distribution. Consequently, by running the resulting Markov chain for long enough, we will be effectively sampling from the distribution of interest. To guarantee the existence of a stationary distribution, the Markov chain must satisfy 3 properties, that are irreducibility, positive recurrency and aperiodicity. The most general sampling algorithm to generate such a Markov chain is the Metropolis-Hastings algorithm. Let $\Pi(\cdot)$ be the target distribution and $p(y|x)$ be a proposal distribution from which it is easy to sample. The Metropolis-Hastings algorithm proceeds as follows:

Algorithm 1. Metropolis-Hastings

1. Initialize X^0 randomly and iteratively repeat the following procedure:
2. Let X^t be the last sample and generate a proposal $Y^{t+1} \sim p(\cdot|X^t)$
3. Compute the acceptance probability $\rho(X^t, Y^{t+1})$ as:

$$\rho(X^t, Y^{t+1}) = min\left\{\frac{\Pi(Y^{t+1})p(X^t|Y^{t+1})}{\Pi(X^t)p(Y^{t+1}|X^t)}, 1\right\}$$

4. Set $X^{t+1} = Y^{t+1}$ with probability $\rho(X^t, Y^{t+1})$. Otherwise, set $X^{t+1} = X^t$

In the special case in which multiple parameters are to be estimated and their marginal conditional distributions have a closed form, a sampling scheme called Gibbs sampling can be applied by using the conditional distribution $p(\theta_i|\theta_{-i}, y)$ as proposal distribution and setting the acceptance probability equal to 1. For example, suppose that we want to sample from the joint posterior of (θ_1, θ_2) The Gibbs sampler generates a sequence of dependent samples as follows:

Algorithm 2. Gibbs sampler

1. Initialize (θ_1^0, θ_2^0) and iteratively repeat the following:
2. Let (θ_1^t, θ_2^t) be the last sample 3. Sample $\theta_1^{t+1} \sim p(\theta_1|\theta_2^t, \mathbf{y})$ and $\theta_2 \sim p(\theta_2|\theta_1^{t+1}, \mathbf{y})$
4. Let $\left(\theta_1^{t+1}, \theta_2^{t+1}\right)$ be a new sample from the joint posterior

Gibbs sampling and Metropolis-Hastings can also be combined if such a conditional distribution is only available for some of the parameters to estimate. This is the case in the algorithm proposed by [14] to sample from the posterior of the model specified by equations 1.13 and 1.14 if the prior on the parameter θ is specified as a conjugate prior with respect to the likelihood. A step of the Markov chain is built by two steps: a Metropolis-Hastings update for the partition π_n and a Gibbs update for the parameters θ.

Algorithm 3. Single LSP Proposal

1. Generate $\pi_n^* \sim q\left(\pi_n \mid \pi_n^{(r)}, v\right) = \text{LSP}\left(\pi_n^{(r)}, v\right)$. Set $\pi_n^{(r+1)} = \pi_n^*$ with probability

$$\mathcal{A}\left(\pi_n^*, \pi_n^{(r)}\right) = \min\left\{1, \frac{p\left(y \mid \boldsymbol{\theta}^{(r)}, \pi_n^*\right) p\left(\pi_n^*\right)}{p\left(y \mid \boldsymbol{\theta}^{(r)}, \pi_n^{(r)}\right) p\left(\pi_n^{(r)}\right)} \times \frac{q\left(\pi_n^{(r)} \mid \pi_n^*, v\right)}{q\left(\pi_n^* \mid \pi_n^{(r)}, v\right)}\right\}$$

Otherwise set $\pi_n^{(r+1)} = \pi_n^{(r)}$.

2. Draw $\boldsymbol{\theta}^{(r+1)} \mid \boldsymbol{y}, \pi_n^{(r+1)}$ using a Gibbs update.

Chapter 2

Application to mouse-tracking data

This chapter will explore a practical application of a Demand Partition model based on the LSP distribution to mouse-tracking data. Mouse-tracking has been rising in popularity as a technique to study the cognitive processes underlying decision making in several contexts such as language processing and memory functions. Moreover, these data can be supplement analysis of other types of data from the domain of neurosciences, such as EEG data. In this chapter, I will first provide an overview of mouse-tracking data and why such data is suitable for clustering analysis. Afterwards, I will describe the details of the specific dataset and model used for this application, concluding with the results of the Markov Chain Monte Carlo sampling scheme for the posterior distribution of the parameters of the model.

2.1 Mouse-tracking experiments.

The general idea of mouse-tracking experiments is to have the participants perform a pre-specified computer task and record its mouse activity. Statistical analysis can then be performed on the data regarding the trajectory and shape drawn by individuals in different tasks.[17] In particular, in one of the first experiments in this domain carried out by Spivey et al. (2005) [15], participants would see two objects on-screen and they

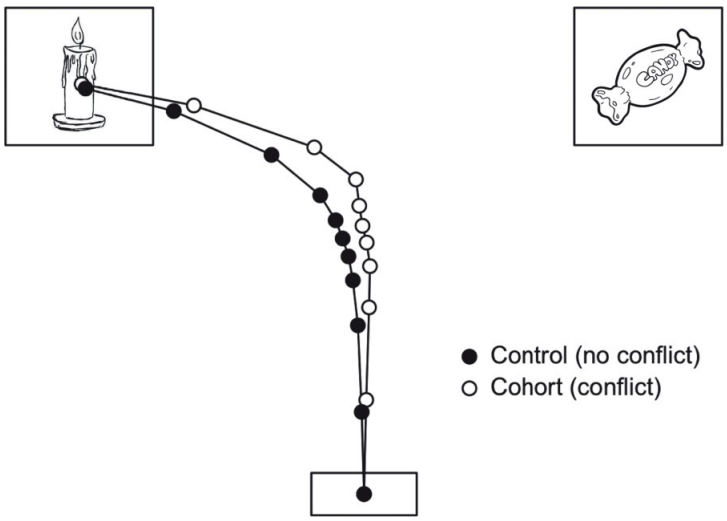


Figure 2.1: "Candle vs. Candy" trial from Spivey et al. (2005)

would see a delayed on-screen stimulus that told them the name of the object they had to click on. The trials were divided among a Cohort group and a Control group. In the cohort group, the two objects were chosen to have names with similar pronounciation (e.g. "Candle vs. Candy" trial shown in Figure 2.1) while in the Control group this was not the case. The hypotesis supporting this study was that the trials in the Cohort group would create conflict in choosing the correct answer. As a consequence, the resulting mouse trajectories would lean more heavily towards the wrong answer than the observations in the Control group.

Further studies on mouse-tracking data have also proposed clustering analysis, by using techniques such as k-means and prototype matching. On the other hand, random partition models allow for more flexibility as the number of clusters is not required to be pre-specified a priori.

2.2 Dataset Description.

The dataset used for this application is the result of an experiment carried out by Kieslich Henninger (2017) and is freely available as part of the R package mousetrap [7]. In this

experiment, participants have to assign animal exemplars to one of two categories, by clicking on the button corresponding to the correct category. The trajectories drawn by the mouse of the participants are then recorded. A trajectory is recorded as a sequence of (x, y) pairs describing the evolution of the mouse position over the course of the experiment. In total, 60 participants took part in 19 different trials, which means that the raw dataset is composed by 1140 sequences of coordinates. An important remark about trials that will be useful when defining the location partition for the LSP prior is that 6 of them are "atypical". This means that the animal is not a typical exemplar of the correct category (e.g. whale in the mammal category). These trials are meant to be more challenging and the difference should be evident from the comparison of trajectories among the two groups, as in figure 2.2, where letting the starting position be $(0, 0)$ and the correct answer be at the top-left of the chart, the average trajectory by condition shows a greater skews towards the alternative for "Atypical" trials. A broader overview of the trials that were part of experiment is given in figure 2.3, where "Exemplar" is the name of exemplar shown to the participants, "CategoryLeft" and "CategoryRight" are respectively the options that were presented at the left and right of the screen and "CategoryCorrect" is the correct among these two options.

2.3 Preprocessing.

Some preprocessing was needed to make the dataset suitable for the intended clustering on a trial level. First of all, not all trajectories have the same length in terms of datapoints collected: since the position of the mouse was recorded at regular intervals, if an individual took a longer time to answer a specific trial, the related trajectories was inevitably composed by a larger number of points. To take into account this heterogeneity, all the trajectories were space-normalized. As a results of the normalization, each trajectory was described by 101 timesteps.

Then, a common simplification in the literature was adopted to reduce the dimensionality

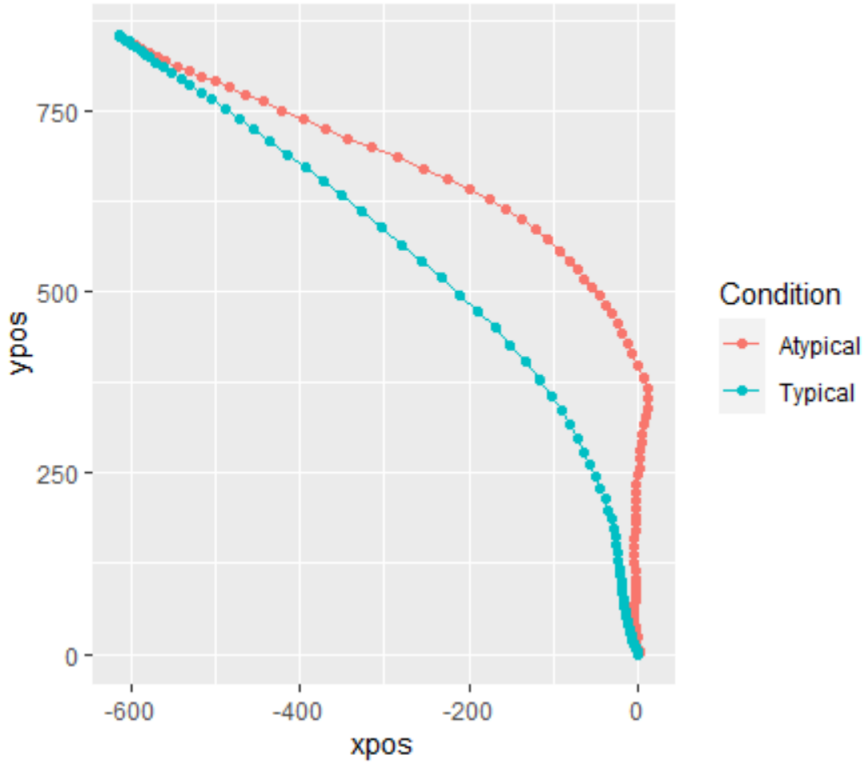
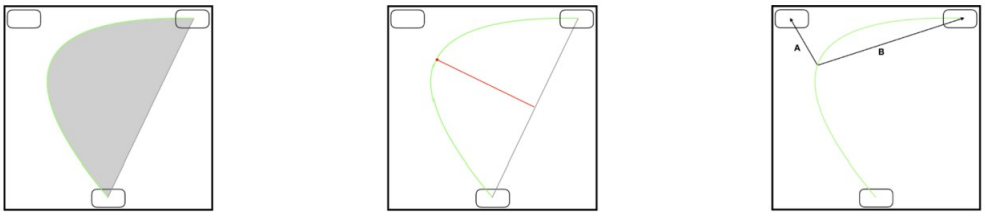


Figure 2.2: Average trajectory by Condition

Condition	Exemplar	CategoryLeft	CategoryRight	CategoryCorrect
Typical	Katze	Saeugetier	Reptil	Saeugetier
Atypical	Pinguin	Vogel	Fisch	Vogel
Atypical	Schmetterling	Insekt	Vogel	Insekt
Typical	Spatz	Saeugetier	Vogel	Vogel
Typical	Falke	Vogel	Reptil	Vogel
Atypical	Aal	Reptil	Fisch	Fisch
Typical	Hund	Saeugetier	Insekt	Saeugetier
Typical	Klapperschlange	Reptil	Amphibie	Reptil
Atypical	Fledermaus	Saeugetier	Vogel	Saeugetier
Typical	Goldfisch	Amphibie	Fisch	Fisch
Typical	Loewe	Saeugetier	Fisch	Saeugetier
Typical	Hai	Fisch	Saeugetier	Fisch
Typical	Kaninchen	Saeugetier	Reptil	Saeugetier
Typical	Lachs	Fisch	Saeugetier	Fisch
Typical	Pferd	Saeugetier	Vogel	Saeugetier
Atypical	Wal	Fisch	Saeugetier	Saeugetier
Typical	Chamaeleon	Insekt	Reptil	Reptil
Atypical	Seeloewe	Fisch	Saeugetier	Saeugetier
Typical	Alligator	Reptil	Saeugetier	Reptil

Figure 2.3: Trials in the dataset



(a) Area Under the Trajectory Curve (b) Maximal Deviation (c) Maximal Log Ratio

Figure 2.4: Commonly used metrics to condense trajectories [8]

of the problem: every trajectory was condensed into a single metric by using its Maximal Deviation. It is worth noting that there is actually a wide variety of metrics that can be used for this purpose. Some of the most effective in distinguishing different decision patterns in the trajectories were shown to be the Area Under the Trajectory Curve, the Maximal Deviation and the Maximal Log Ratio [8]. The approach of these measures in slightly different in that the Area under the curve and the maximal deviation consider an ideal straight line trajectory going from the starting point to the correct answer and capture the distance between the observed trajectory and the ideal one. In particular the first one does so by measuring the area between the two trajectories while the second one measures the maximal distance achieved by a point in the observed trajectory to the ideal one. On the other hand, the Maximal Log Ratio is calculated by keeping into account the ratio of the distance from the correct answer to the distance from the wrong answer (Fig. 2.4).

The resulting dataset is a 19x60 matrix, which contains an entry per trajectory recorded.

2.4 Model.

The goal of the proposed model is to cluster the $n = 19$ trials by using the results of $d = 60$ participants as features. Let $X_{i,j}$ denote the maximal deviation of the trajectory drawn by individual i in trial j. The hierarchical representation of the generative model

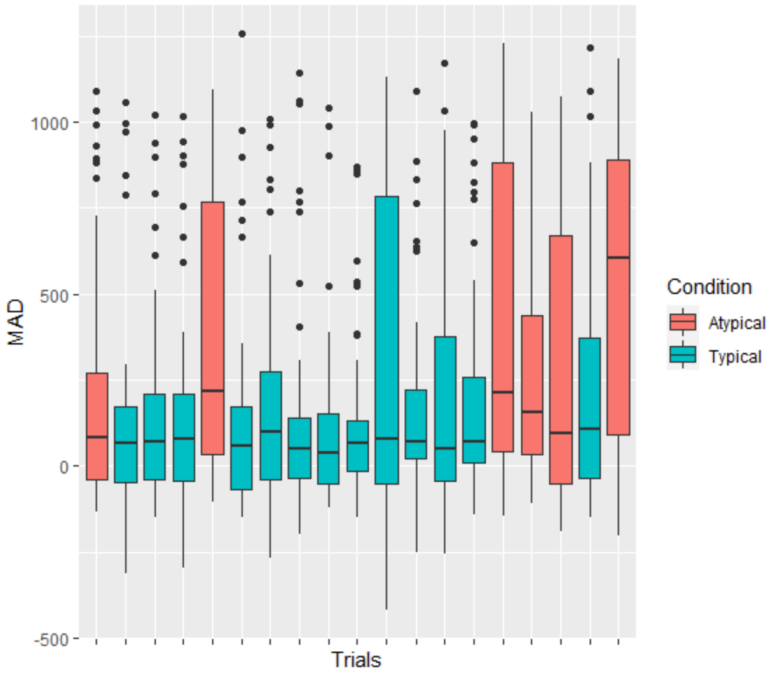


Figure 2.5: Maximal Deviation per trial

is as follows:

$$X_{ij}|\theta_{m_i}, \pi_n \overset{ind.}{\sim} \mathcal{N}(\theta_{m_i}, \sigma^2), \qquad i = 1, \ldots, n,\ j = 1, \ldots, d, \tag{2.1}$$

where θ_{m_i} indicates the cluster-specific mean of the cluster where trial i has been assigned to. Let $\boldsymbol{\theta}_{\pi_n} = (\theta_1, \ldots, \theta_{K_n})^T$ denote the vector grouping the cluster-specific means as defined by the latent partition π_n. Then, we assume

$$\boldsymbol{\theta}_{\pi_n}|\pi_n \sim \mathcal{N}(\mathbf{0}_{\pi_n}, \gamma^2\, I_{\pi_n}), \tag{2.2}$$

where $\mathbf{0}_{\pi_n}$ and I_{π_n} denote, respectively, a K_n-dimensional vector of zeros and a $K_n \times K_n$ identity matrix defined by the partition π_n with K_n denoting the number of clusters in π_n. The prior on the partition π_n over the set of trials is then assumed to be the LSP prior defined in section 1.6,

$$\pi_n \sim LSP(\rho_n, \tau), \tag{2.3}$$

where

$$\rho_n = (g_1, ..., g_n) \text{ with } g_i = \begin{cases} 1 & \text{if trial i is Typical,} \\ 2 & \text{if trial i is Atypical.} \end{cases}$$

An important choice for this model regards the hyperparameter τ of the LSP distribution. As mentioned in the Theory chapter, τ is a scale parameter. Roughly speaking, this means that it dictates how spread out is the probability mass across the partition space. In particular, for small values of τ, the mass of the LSP distribution will be mostly concentrated around ρ_n, while as τ grows, the distribution will assign more mass to partition that are significantly different from the location partitions ρ_n. The behaviour of the LSP distribution for different choices of τ is illustrated in figure 2.6. In this plot, 5 values were considered ($\tau \in \{0.05, 0.5, 1.5, 5, 10\}$) and for each of them, $50,000$ samples were drawn from an LSP distribution centered at ρ_n. The distribution of the recorded distance from ρ_n was plotted for each of the alternative. Here the distance metric between an arbitrary partition $\mathbf{c}$ and $\rho_\mathbf{n}$ used is the Variation of Information [11]. This metric uses the entropy $H(X) = \sum_{x \in K} p(x) \log p(x)$ to measure the uncertainty in the value of a random variable X with sample space K, and is computed as

$$\begin{aligned} VI(\pi, \rho_\mathbf{n}) &= -H(\pi) - H(\rho_\mathbf{n}) + 2H(\pi \wedge \rho_\mathbf{n}) \\ &= \sum_{j=1}^{K} \frac{\lambda_j}{N} \log_2 \left(\frac{\lambda_j}{N} \right) + \sum_{l=1}^{K'} \frac{\lambda'_l}{N} \log_2 \left(\frac{\lambda'_l}{N} \right) - 2 \sum_{j=1}^{K} \sum_{l=1}^{K'} \frac{\lambda^{\wedge}_{jl}}{N} \log_2 \left(\frac{\lambda_{jl}}{N} \right), \end{aligned} \tag{2.4}$$

where K denotes the number of clusters in π, K' the number of clusters in $\rho_\mathbf{n}$, λ_i and λ'_l the sizes of clusters in the corresponding partitions and $\lambda^{\wedge}_{jl}$ the sizes of all possible intersections between a cluster of π and a cluster of ρ_n. The figure shows that as τ increases, the empirical distribution is more skewed towards samples with higher distance from the location partition. On the other hand, generally for values lower than 0.1, the probability density is highly concentrated on the location partition. In particular, in the model defined by equations (2.1)–(2.3) the choice of the scale parameter has two distinct roles. Firstly,

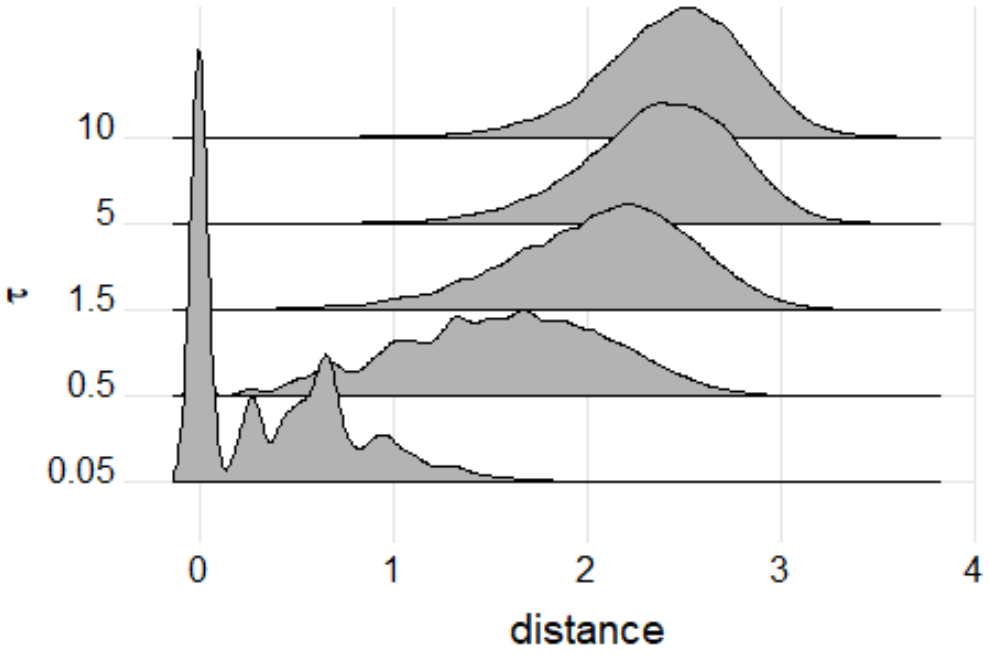


Figure 2.6: Comparison of different choices for τ in the LSP distribution

it control the informativeness of the prior, as the higher τ, the less informative is the prior distribution about π_n. Secondly, since the LSP distribution is also used as proposal in the sampling algorithm, its scale parameter controls how different the proposed partitions are from the current. Hence, it can be see as the step size of the random walk followed by Algorithm 3 in the sampling process. Following the choices of [14], the prior will be kept rather informative by setting $\tau = 0.01$, while the step size of the MCMC proposal will be fine-tuned to achieve good posterior mixing.

2.4.1 Conjugacy of the normal model

As mentioned in Chapter 1, posterior sampling from this model was performed basing on Algorithm 3. Let θ_m be the cluster-specific mean for an arbitrary cluster $m \in [1, ..., K]$. By assuming the conditional distribution of $\boldsymbol{\theta}$ to have a closed form, step 2 of that algorithm involves performing a Gibbs update through direct sampling from the posterior distribution of θ_m for every cluster m in the current partition. Because of the conjugacy of the normal model, such posterior distribution has indeed a closed form, which is derived

as follows:

$$
\begin{aligned}
p(\theta_m|\pi_n, X_1, ..., X_n) \propto p(X_1, ..., X_n|\theta_m, \pi_n)p(\theta_m|\pi_n) &= \prod_{i \in C_m} \mathcal{N}(\theta_m \mathbf{1}, \sigma^2 I)\mathcal{N}(0, \tau^2) \\
&\propto \exp\left\{-\frac{1}{2\sigma^2}\sum_{i \in C_m}(\mathbf{X_i} - \theta_m \mathbf{1})^T(\mathbf{X_i} - \theta_m \mathbf{1})\right\} \cdot \exp\left\{-\frac{\theta_m^2}{2\tau^2}\right\} \\
&= \exp\left\{-\frac{dn\theta_m^2}{2\sigma^2} + \frac{\theta_m}{\sigma^2}\sum_{i \in C_m}\sum_{j=1}^{d} X_{i,j}\right\} \cdot \exp\left\{-\frac{\theta_m^2}{2\tau^2}\right\} \\
&= \exp\left\{-\frac{1}{2}\left(\frac{dn}{\sigma^2} + \frac{1}{\tau^2}\right)\theta_m^2 + \frac{\theta_m}{\sigma^2}\sum_{i \in C_m}\sum_{j=1}^{d} X_{i,j}\right\}.
\end{aligned}
$$

The right-hand side of the last equation is the kernel of the normal distribution [6] with mean equal to $\frac{\sum_{i \in C_m}\sum_{j=1}^{d} X_{i,j}}{dn + \frac{\sigma^2}{\tau^2}}$ and variance $\frac{1}{\frac{dn}{\sigma^2} + \frac{1}{\tau^2}}$. Therefore,

$$
\theta_m|\pi_n, X_1, ..., X_n \sim \mathcal{N}\left(\frac{1}{dn + \frac{\sigma^2}{\tau^2}}\sum_{i \in C_m}\sum_{j=1}^{d} X_{i,j}, \frac{1}{\frac{dn}{\sigma^2} + \frac{1}{\tau^2}}\right), \tag{2.5}
$$

where $n = 19$ is the number of trials, $d = 60$ is the number of participants and C_m is the set of indices of elements in cluster m. This expression allows for straightforward sampling from the distribution of θ_m conditional on the data and the partition π_n in step 2 of the proposed sampling algorithm.

2.5 Results.

First of all, a k-means algorithm with $k = 3$ was run and the result π_0 was used as initial partition for the MCMC sampling scheme. The remaining parameters of the MCMC was set as in table 2.1, where v is the scale parameter of the LSP distribution used as proposal

Steps	Burn-in	Thin	v
300,000	0.5	10	9e-6

Table 2.1: MCMC parameters

and controls the step size of the random walk, the burn-in parameter controls the fraction of samples that are discarded to account for convergence time of the Markov chain and the thinning parameter controls the rate at which new steps are kept. In this case, a value of 10 means that every tenth step of the chain was kept as a sample from the posterior distribution. This allows for better posterior mixing, by avoiding multicollinearity that can be present due to the small step size. The resulting similarity matrix based on co-clustering probability is plotted in figure 2.7, where atypical trials are numbered $1 - 6$.

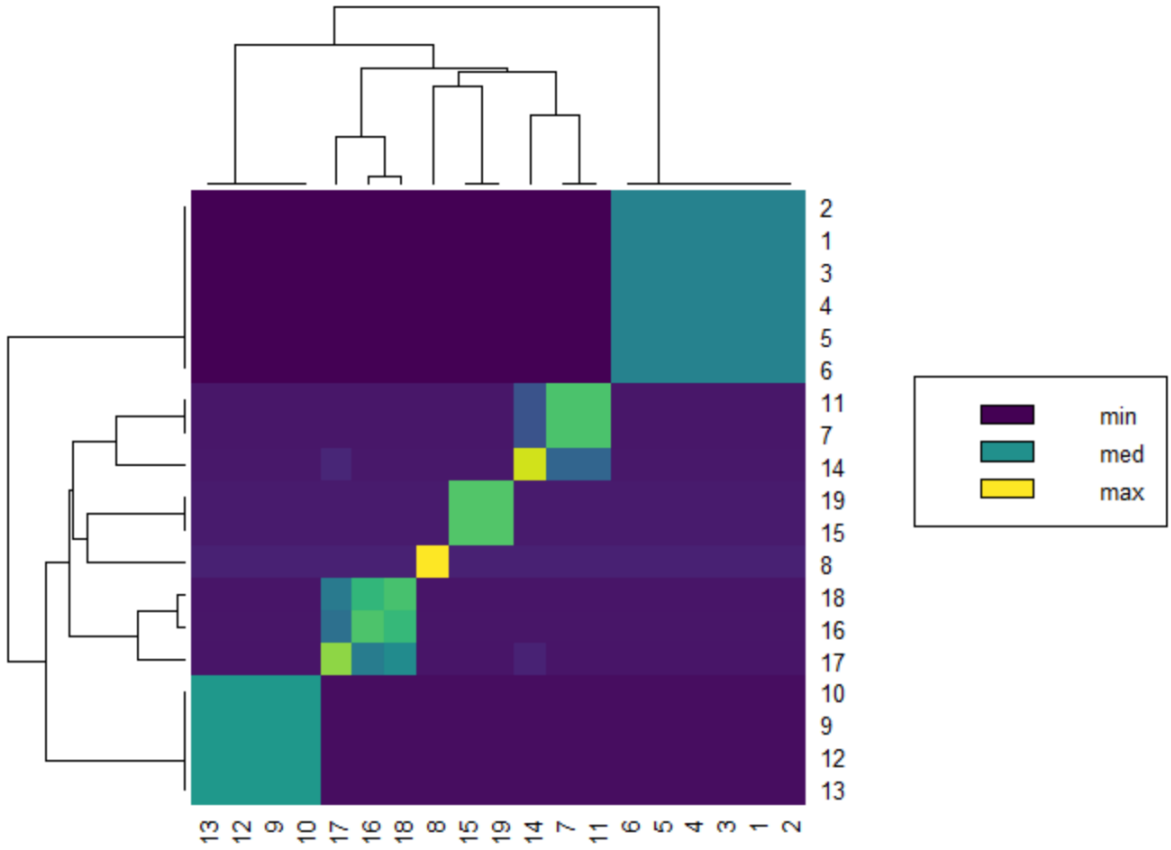


Figure 2.7: Co-occurence matrix

2.5.1 Point estimation.

To obtain point estimates one needs to define a loss function, which is not a straightforward task in clustering problems with large partition space. [16] propose to use the variation of information (eq. 2.4) as loss function. Therefore, the point estimate $\boldsymbol{\pi}^*$ would be the

partition that minimizes the expected variation of information between the partitions and $\boldsymbol{\pi}^*$, that is

$$\boldsymbol{\pi}^* = \underset{\hat{\pi}}{\operatorname{argmin}} \, \mathbb{E}\left[VI(\pi, \hat{\pi}) | \boldsymbol{X}\right].$$

Moreover, to make the calculation computationally feasibile, it is also suggested to estimate the argmin of such function in a large partition space by restricting the search to a subset of partitions. For example, such a subset could include only the partitions drawn during the MCMC sampling. The resulting point estimate $\hat{\pi}$, as compared with the location partition ρ_n is

$$\hat{\pi} = (1,\, 1,\, 1,\, 1,\, 1,\, 1,\, 2,\, 3,\, 4,\, 4,\, 2,\, 4,\, 4,\, 5,\, 6,\, 7,\, 7,\, 7,\, 6)\,,$$
$$\rho_n = (1,\, 1,\, 1,\, 1,\, 1,\, 1,\, 2,\, 2,\, 2,\, 2,\, 2,\, 2,\, 2,\, 2,\, 2,\, 2,\, 2,\, 2,\, 2)\,.$$

2.6 Limitations and future work

The model shows some limitations that could be addressed in future works.

1. In the specific dataset under consideration, some trials show higher variance than other, as highlighted by figure 2.5. Currently, this variability is not captured by the model, where the variance for the results of trials across participants is assumed to be constant and equal to σ^2 in equation 2.1. Therefore, it may be worth enriching the model by defining cluster-specific variances σ_m^2, allowing observations from separate clusters to have different variance.

2. Even after extensive hyperparameter tuning, posterior mixing is still not ideal. Although K-means initialization improved on random initialization for what concerns the exploration of the partition space, the chain is extremely sensible to the step size parameter. Moreover, the acceptance rate is not representative of actual mixing because it is influenced by the usage of LSP as proposal distribution, which is very

likely to propose an identical partition to current one for low step sizes. This could be addressed by adopting a different proposal approach that modifies a subset of the partition at each step, such as the Merge-Split proposal or the Block LSP proposal from [14].

3. This study focused on clustering on a trials level. However, interesting insights from similar mouse tracking datasets could also be drawn by clustering at the participants level. This approach would require more sophisticated nonparametric models, for example based on the hierarchical Dirichlet Process.

Bibliography

[1] Lecture 23: Bayesian nonparametrics: Dirichlet processes. URL https://www.cs.cmu.edu/~epxing/Class/10708-20/scribe/lec23_scribe.pdf.

[2] D. Blackwell and J. B. MacQueen. Ferguson distributions via polya urn schemes. *The Annals of Statistics*, 1(2):353–355, 1973. ISSN 00905364. URL http://www.jstor.org/stable/2958020.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003. ISSN 1532-4435. doi: http://dx.doi.org/10.1162/jmlr.2003.3.4-5.993. URL http://portal.acm.org/citation.cfm?id=944937.

[4] T. S. Ferguson. A bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230, 1973. ISSN 00905364. URL http://www.jstor.org/stable/2958008.

[5] E. Fox and M. Jordan. Mixed membership models for time series. *Handbook of Mixed Membership Models and Their Applications*, 09 2013.

[6] P. D. Hoff. *A First Course in Bayesian Statistical Methods*. Springer Publishing Company, Incorporated, 1st edition, 2009. ISBN 0387922997.

[7] P. Kieslich and F. Henninger. Mousetrap: An integrated, open-source mouse-tracking package. *Behavior research methods*, 49, 06 2017. doi: 10.3758/s13428-017-0900-z.

[8] M. Maldonado, E. Dunbar, and E. Chemla. Mouse tracking as a window into decision

making. *Behavior Research Methods*, 51(3):1085–1101, June 2019. doi: 10.3758/s13428-018-01194-x. URL https://hal.archives-ouvertes.fr/hal-02274523.

[9] P. Muller, F. A. Quintana, A. Jara, and T. Hanson. *Bayesian Nonparametric Data Analysis*. Springer Publishing Company, Incorporated, 2015. ISBN 3319189670.

[10] P. Müller, F. Quintana, and G. Rosner. A product partition model with regression on covariates. *Journal of computational and graphical statistics : a joint publication of American Statistical Association, Institute of Mathematical Statistics, Interface Foundation of North America*, 20:260–278, 03 2011. doi: 10.1198/jcgs.2011.09066.

[11] S. Paganin, A. H. Herring, A. F. Olshan, and D. B. Dunson. Centered Partition Processes: Informative Priors for Clustering (with Discussion). *Bayesian Analysis*, 16(1):301 – 670, 2021. doi: 10.1214/20-BA1197. URL https://doi.org/10.1214/20-BA1197.

[12] J.-H. Park and D. B. Dunson. Bayesian generalized product partition model. *Statistica Sinica*, 20(3):1203–1226, 2010. ISSN 10170405, 19968507. URL http://www.jstor.org/stable/24309487.

[13] J. Sethuraman. A constructive definition of the dirichlet prior. *Statistica Sinica*, 4: 639–650, 01 1994.

[14] A. N. Smith and G. M. Allenby. Demand models with random partitions. *Journal of the American Statistical Association*, 115(529):47–65, 2020. doi: 10.1080/01621459.2019.1604360. URL https://doi.org/10.1080/01621459.2019.1604360.

[15] M. J. Spivey, M. Grosjean, and G. Knoblich. Continuous attraction toward phonological competitors. *Proceedings of the National Academy of Sciences*, 102(29):10393–10398, 2005. doi: 10.1073/pnas.0503903102. URL https://www.pnas.org/doi/abs/10.1073/pnas.0503903102.

[16] S. Wade and Z. Ghahramani. Bayesian Cluster Analysis: Point Estimation and Credible Balls (with Discussion). *Bayesian Analysis*, 13(2):559 – 626, 2018. doi: 10.1214/17-BA1073. URL `https://doi.org/10.1214/17-BA1073`.

[17] D. Wulff, J. Haslbeck, P. Kieslich, F. Henninger, and M. Schulte-Mecklenbeck. *Mouse-tracking: Detecting types in movement trajectories*, pages 131–145. 06 2019.