Bayesian non-parametric inference with monotonicity restrictions for discrete choice experiments

Remi Daviet [*] and William McCausland[†]

December 31, 2016

Abstract

Insert abstract here

Keywords: Big Data, Discrete Choice experiments

[*] Corresponding author. Department of economics, University of Toronto; remi.daviet@mail.utoronto.ca

[†] Department of economics, Université de Montréal

1 Products attributes and products classes

Every product considered is represented by a point in the chosen attributes space. For instance, a mobile phone can be represented by a set of two attributes, the screen size and the color. Each attribute has a level. In our example, the screen size can have 3 levels $(Small, Medium$ or $Large)$ and the color can have 4 levels $(Black, White, Blue$ or $Other)$. Considering this attributes space, two possible phones are phone 1 $(Large, Blue)$ and phone 2 $(Medium, White)$.

Products can be regrouped into classes. Classes are set of attributes commons to every product belonging to the class. A class can be very simple, for instance the class $(\{Small\}, any)$ which regroups all the phones with a small screen size, no matter the color. Classes can also be more complicated such as $(\{Small, Large\}, \{White, Other\})$ which regroups phones of small or large screen size having a white or "other" color. Note that the $(\{Small\}, any)$ class can be written $(\{Small\}, \{Black, White, Blue, Other\})$ and that the keyword "any" is just used for convenience. Finally, a class can specify a unique product in the attribute space, such as the class $(\{Large\}, \{Blue\})$.

More formally, considering a product space with L attributes, a product x_m is represented by the vector of attributes $(x_{m1}, x_{m2}, \ldots, x_{mL})$. Each attribute i can have L_i possible levels. For instance, if the attribute (1) "screen size" has 3 possible levels $(Small, Medium$ or $Large)$, we write $L_1 = 3$. For the second attribute "Color" which has 4 possible levels, we write $L_2 = 4$. The attributes levels attributed to a product x_m are indicated by their corresponding number. For instance, if $Small$ and $Black$ are the first levels of their respective attributes, the phone will be represented by the vector $(x_{m1} = 1, x_{m2} = 1)$.

A class of product Λ_k is represented by a Cartesian product of L sets $(\Lambda_{k1}, \Lambda_{k2}, \ldots, \Lambda_{kL})$. Each Λ_{ki} represents the set of possible levels for the attribute i for a product in the class k. The levels belonging to Λ_{ki} are called active levels. For instance, a class regrouping $Black$ phones with either $Medium$ or $Large$ screen size will be represented by $(\Lambda_{k1} = \{2, 3\}, \Lambda_{k2} = \{1\})$. When all the levels of an attribute are active, it means that this particular attribute does not matter when defining the class, and the attribute is defined as inactive. On the contrary, when only some levels are active, it means that the attribute is used as a criterion to select if a product is a member of the class, and the attribute is defined as active.

A product x_m belongs to a class Λ_k if and only if $x_{m1} \in \Lambda_{k1}, x_{m2} \in \Lambda_{k2}, \ldots$ and $x_{mL} \in \Lambda_{kL}$. For convenience, we will use the notation $x_m \in \Lambda_k$ to indicate that the product x_m is a member of the class Λ_k.

2 Individual preferences

We have a population of individuals, each one choosing a product within a set at a time t. The choice of an individual n is done by maximizing a function $u(x, \psi)$ representing the preferences this individual over the set of available choices X_{nt}, where ψ is some set of parameters. The choice maximizing the individual preferences is denoted x^*. The preferences for each individual are built by attributing some value to various classes of products. The value of a class can be positive or negative. For instance, at a time t an individual n attributes the value θ_{nkt} to the class Λ_k. The value of a given product for an individual is obtained by summing the values of all the classes it belongs to. If the value of a class k is zero for an individual, we say that the individual does not possess the feature k. It means that whether a product belongs to the class k or not does not change the value of the product for the given individual.

Within the population, when two individuals have a non-zero θ_{nkt} for a class of products k, we say that they share the common feature k.The values come from a common distribution $f_k(\theta_{nkt})$. It is important to note however that the value of the class does not need to be the same for both individuals sharing the feature. One individual with feature k may attribute a positive value to the class k while the other attributes a negative value. Some features can be shared by many individuals, while some other features can be rare. Note that there is some redundancy: a same class can appear in two different features. The two classes of products Λ_k and $\Lambda_{k'}$ can be identical ($\Lambda_k = \Lambda_{k'}$), but to be respectively associated with a θ_{nk} and a $\theta_{nk'}$ following different distributions.

The total number of features in the population is potentially infinite. However, in a given sample only a finite number of features are realized. The features present in our sample are numbered from 1 to K_+.

Formally, the value of an object x for and individual n at time t can be written:

$$u(x, \psi) = \sum_{k=1}^{K_+} \theta_{nkt} \cdot \mathbb{1}(x \in \Lambda_k), \tag{2.1}$$

where ψ contains the θ_{nkt} and the Λ_k.

The value θ_{nk} may be decomposed into four parts as follow:

$$\theta_{nkt} = z_{nk} \cdot (-1)^{s_{nk}} \cdot (\theta_{nk} + \epsilon_{nkt})$$

where z_{nk} is a binary variable indicating whether individual n possesses feature k, s_{nk} is a binary variable that indicates whether the value of the class k for individual n is positive or negative, θ_{nk} is the stable part of the value in the consumer preferences, and ϵ_{nkt} is a tremble part.

3 Hierarchichal model

The Bayesian framework gives us a simple and intuitive way to model our uncertainty about the various parameters. We have to define how the various variables and parameters are distributed in order to be able to obtain a posterior distribution.

3.1 Features allocation

Our first object of interest is the binary matrix of features allocation $Z = \{z_{nk}\}$ where each row represents an individual and each column represents a feature. While the Z matrix has an infinite number of columns, we are only interested in the columns where there is at least a 1 for our sample. We use the infinite feature model defined by Griffiths and Ghahramani (2011) in their paper about the Indian Buffet Process (IBP). The IBP approach presents a convenient way to perform inference by simulation for models using latent features. Using this model, we assume that the probability that any individual possesses feature k is π_k^z and that the features are generated independently. The distribution of the infinite matrix

Z can be obtained by starting with a finite $N \times K$ matrix of features allocation Z_K and taking the limit when $K \rightarrow \infty$.

In the finite case, the probability of Z_K given $\pi^z = \{\pi_1^z, \ldots, \pi_K^z\}$ is:

$$P(Z_K|\pi) = \prod_{k=1}^{K} \prod_{n=1}^{N} P(z_{nk}|\pi_k^z) = \prod_{k=1}^{K} (\pi_k^z)^{m_k} (1 - \pi_k^z)^{N - m_k}$$

where $m_k = \sum_{n=1}^{N} z_{nk}$ is the number of individuals having feature k. We can then define a prior on π^z using a $Beta$ distribution:

$$\pi_k^z \sim Beta(\frac{\alpha^z \beta^z}{K}, \beta^z)$$

In order to take the limit as $K \rightarrow \infty$, we need to define a distribution over classes of equivalent matrices. Matrices are defined as equivalent if one can be obtained from the other by permutation of the columns. A class of equivalent matrices is denoted $[Z]$. In order to be able to compare matrices, we use permutations on their column in order to put them in left ordered form. Griffiths and Ghahramani (2011) define the left-ordered form as follow: "$lof(Z)$ is obtained by ordering the columns of the binary matrix Z from left to right by the magnitude of the binary number expressed by that column, taking the first row as the most significant bit". Two equivalent matrices have the same left-ordered form. Taking left-ordered forms let us define a distribution over equivalent classes of matrices. Matrices with left-ordered forms following that distribution can be generated using the IBP.

Using this process, the average number of feature per individual will be α^z. The β^z parameter lets us set the average overall number of features observed in a sample. The expected overall number of features present in a sample will be $\alpha^z \sum_{n=1}^{N} \frac{\beta^z}{\beta^z + n - 1}$. Depending on β, this number can range from α^z where everybody share the same features, to $N\alpha^z$ where no features are shared.

Here are three matrices produced using the described process using various parameter values:

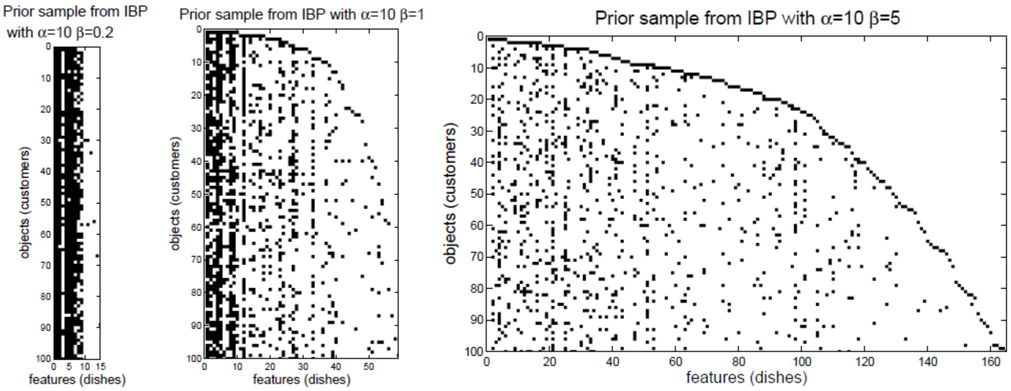

Figure 1: Three matrices produced with the IBP.(Griffiths and Ghahramani, 2011)

3.2 Attributes with monotone preferences and Λ_k

The second object of interest are the product classes Λ_k associated with each feature. We need to define a distribution over the space of Cartesian products of possible Λ_{ki} excluding the empty set. While there is no closed form for the distribution of Λ_k we can define a generating process in two steps. The first step generates the set of active attributes C_k with cardinality c_k and the second step determines the set of active levels Λ_{ki} for each active attribute.

It is possible that we want to restrict preferences such that they only increase in some attributes. For such attribute, the active levels are restricted to be of the form $\{l, l+1, \ldots, L\}$ if the the class Λ_k is associated with a $s_{nk} = 1$, and of the form $\{1, \ldots, l-1, l\}$ if the class is associated with a $s_{nk} = 0$. We denote the first case as a upper active levels, and the second as lower active levels. If a class Λ_k has at least one active monotone attribute, we denote it by a corresponding indicator $m_k = 1$, and $m_k = 0$ otherwise. Note that if several monotone attributes are active in the class, they should all be of the same type. If all the active monotone attribute are of the upper active levels type, we say that the class has a positive polarity and denote it $\rho_k = 1$. conversely, if the monotone attributes are of the lower active levels type, we say that the class has a negative polarity and denote it $\rho_k = 0$.

This leads us to the following generating process for drawing a Λ_k:

- Draw the number of active attributes $A \sim U(1, ..., L)$
- Draw polarity $\rho_k \sim Bernoulli(0.5)$
- Initialize the monotonicity indicator $m_k = 0$
- Draw A attributes represented by the set L^*:
- For each $i \in L^*$:
  - If attribute i is monotone :
    - Set $m_k = 1$
    - Draw one level $l \sim U(1, ..., L_i)$
    - If $\rho_k = 1$:
      - Set level l and all level above to active
    - If $\rho_k = 0$:
      - Set level l and all level below to active
  - Else :
    - Draw the number of active levels $a \sim U(1, ..., L_i - 1)$
    - Draw a level without replacement and set them to active
- For each $i \notin L^*$:
  - Set all levels of attribute i to active

A more general alternative could be used for the generating process. We first draw the number of active attributes c_k with a completely flexible distribution π^{Λ} is an n-vector giving the prior distribution of number of active attributes in utility term k. Given the number of active attributes, the set of active attributes is given by *conditional* Poisson sampling from the set $\{1, \ldots, L\}$, with sample size c_k and parameter vector $\lambda \in \mathbb{R}^L$. The vector λ gives the relative importance of the various attributes. A higher weight λ_i indicates that attribute i is more likely to be active in any given utility term. Conditional Poisson sampling guarantees that the random set C_k has cardinality c_k with probability one. The normalization factor associated with conditional Poisson sampling is not known in closed form, but there is an efficient algorithm for computing it. Given the set C_k of active attributes, subsets Λ_{ki}, $i \in C_k$, are mutually independent. For non monotone attributes, the set of active levels is obtained by Poisson sampling, truncated to assign probability zero to both the empty set and the the set containing all the possible levels. For monotone attributes, the set is drawn following the same process as in the simple case.

3.3 Other parameters

The other parameters $(s_{nk}, \theta_{nk}, \epsilon_{nkt})$ are defined conditional on z_{nk}. When $z_{nk} = 0$, the values of all these parameters are arbitrarily set to 0. We have to define the prior distributions of $(s_{nk}, \theta_{nk}, \epsilon_{nkt})$ conditional on $z_{nk} = 1$:

$$s_{nk}|\pi_k^s, z_{ik} = 1 \sim \begin{cases} 0 & if\ m_k = 1 \text{ and } \rho_k = 1 \\ 1 & if\ m_k = 1 \text{ and } \rho_k = 0 \\ Bernoulli(\pi_k^s) & \text{otherwise} \end{cases}$$

$$\theta_{nk}|\sigma^\theta, \alpha_\theta, z_{ik} = 1 \sim Gamma(\sigma^\theta \alpha^\theta, 1)$$

$$\epsilon_{nkt}|\sigma^\theta, \alpha^\theta, z_{ik} = 1 \sim Gamma((1 - \sigma^\theta)\alpha^\theta, 1)$$

where (s_{nk}, θ_{nk}) are independent across n and k; and ϵ_{nkt} are independent across n,k and t.

Note that the distribution of the value size $(\theta_{nk} + \epsilon_{nkt})$ is also Gamma with parameters $Gamma(\alpha^\theta, 1)$ from the properties of a sum of Gamma distributed variables. The parameter α^θ is the shape parameter of the distribution of the value size, while σ^θ can be interpreted as a stability parameter. For σ^θ close to one, the stable component θ_{nk} is favored and the utility functions have high serial dependence; for σ^θ close to zero, the variable component ϵ_{nkt} is favored and the utility function has low serial dependence.

Finally, we have to define hyperpriors on some of the hyperparameters:

$$\alpha^\theta \sim Gamma(1, 1)$$

$$\sigma^\theta \sim Gamma(1, 1)$$

$$\pi_k^s \sim Beta(1, 1)$$

4 Inference

4.1 Likelihood and Posetrior

Conditional on the set of parameters ψ, there is no random term in the utility function. The likelihood becomes a degenerate function that is equal to 1 if all the observed choices are predicted correctly and 0 otherwise. The probability of observing the set of choices $\{x^*_{nt}\}$ is:

$$P(\{x^*_{nt}\}|\psi) = \mathbb{1}\{\arg\max_{x \in X_{nt}} u(x,\psi) = x^*_{nt}, \forall n, t\}$$

Likelihood-based inference consists in recovering the identified set of parameters such that:

$$P(\{x^*_{nt}\}|\psi) = 1$$

We can see that with the current likelihood function, if at least one choice is not maximizing utility according to the current value of ψ, the likelihood takes the value of 0.

In the Bayesian framework, the target distribution is the corresponding posterior:

$$P(\psi|\{x^*_{nt}\}) \propto P(\{x^*_{nt}\}|\psi) \cdot P(\psi),$$

where $P(\psi)$ denotes the prior density of our parameters.

4.2 Tempering and Sequence of Distributions

Simulating the posterior can be quite difficult, and we simulate instead a sequence of target distributions converging to this Posterior. The functions used in the sequence are increasingly difficult to simulate. This process is known as tempering, and we achieve it by combining two methods. First, we use Data Tempering and introduce the observations one by one in the likelihood. We start with the first observation of the first individual $x^*_{1,1}$, then add the subsequent observations for this individual until all the observations are included in the likelihood. We then add the observations of the following individual, and continue until every observation of every individual are included. For the second tempering approach, we use an instrumental function that does not take a value of zero if the last observed choice is not utility maximizing with the current value of ψ. We call this approach ζ-tempering and substitute the likelihood with the following function:

$$Q(\{x^*_{nt}\}|\psi;\eta,\tau,\zeta) = P(\{\{x^*_{nt}\}_{t=1}^{T}\}_{n=1}^{\eta-1}|\psi) \cdot P(\{x^*_{\eta t}\}_{t=1}^{\tau-1}|\psi) \cdot \zeta^{u(x^*_{\eta\tau},\psi) - \max_{x \in X_{\eta\tau}} u(x,\psi)},$$

where $\zeta > 1$ is a constant chosen by the researcher. We can see that this instrumental function is the product of 3 terms. The first one is the likelihood of observing the choices of individuals 1 to $\eta - 1$. The second is the likelihood of observing the $\tau - 1$ first choices of individual η. The last term requires further attention. If the last observed choice is utility maximizing, this term is equal to 1 as for the likelihood. However, it is equal to a value between 0 and 1 if another choice would be utility maximizing. The smaller the difference between the observed choice and the utility maximizing one, the closer the term gets to 1. It is important to note that $Q(\{x^*_{nt}\}|\psi;\eta,\tau,\zeta) \rightarrow P(\{x^*_{\eta t}\}_{t=1}^{\tau} \cup \{\{x^*_{nt}\}_{t=1}^{T}\}_{n=1}^{\eta-1}|\psi)$ as $\zeta \rightarrow \infty$.

The corresponding quasi-posterior distribution is defined as

$$Q(\psi|\{x^*_{nt}\};\eta,\tau,\zeta) \propto Q(\{x^*_{nt}\}|\psi;\eta,\tau,\zeta)\cdot P(\psi).$$

Using these tempering approaches, we can define a sequence of distributions by specifying the values (η,τ,ζ) for each distribution in the sequence. We start by the distribution

$$Q(\psi|\{x^*_{nt}\};\eta=1,\tau=1,\zeta=2) \propto 2^{u(x^*_{1,1},\psi)-\max_{x\in X_{1,1}} u(x,\psi)}\cdot P(\psi)$$

and progressively increase ζ in each subsequent distribution until $\zeta\to\infty$ and the target distribution becomes

$$Q(\psi|\{x^*_{nt}\};\eta=1,\tau=1,\zeta\to\infty) \propto P(\psi|\{x^*_{1,1}\}).$$

We then alternatively increase the value of τ by 1 and progressively increase ζ from 2 to ∞ until we reach the last observation T for the current individual. For our application, we choose to increase ζ by a factor of 2 with each step in the sequence. We then increase η by one and start again alternatively increasing the value of τ by 1 and progressively increasing ζ from 2 to ∞ until we reach the last observation of the last individual N.

The ratio of two consecutive target distributions when increasing ζ by a factor of a is equal to

$$\frac{P(\{\{x^*_{nt}\}_{t=1}^{T}\}_{n=1}^{\eta-1}|\psi)\cdot P(\{x^*_{\eta t}\}_{t=1}^{\tau-1}|\psi)\cdot (a\zeta)^{u(x^*_{\eta\tau},\psi)-\max_{x\in X_{\eta\tau}} u(x,\psi)}}{P(\{\{x^*_{nt}\}_{t=1}^{T}\}_{n=1}^{\eta-1}|\psi)\cdot P(\{x^*_{\eta t}\}_{t=1}^{\tau-1}|\psi)\cdot \zeta^{u(x^*_{\eta\tau},\psi)-\max_{x\in X_{\eta\tau}} u(x,\psi)}} = a^{u(x^*_{\eta\tau},\psi)-\max_{x\in X_{\eta\tau}} u(x,\psi)}.$$

The ratio of two consecutive target distributions when increasing τ by 1 and reseting ζ to 2 is equal to:

$$\frac{P(\{\{x^*_{nt}\}_{t=1}^{T}\}_{n=1}^{\eta-1}|\psi)\cdot P(\{x^*_{\eta t}\}_{t=1}^{\tau-1}|\psi)\cdot 2^{u(x^*_{\eta\tau},\psi)-\max_{x\in X_{\eta\tau}} u(x,\psi)}}{P(\{\{x^*_{nt}\}_{t=1}^{T}\}_{n=1}^{\eta-1}|\psi)\cdot P(\{x^*_{\eta t}\}_{t=1}^{\tau-1}|\psi)} = 2^{u(x^*_{\eta\tau},\psi)-\max_{x\in X_{\eta\tau}} u(x,\psi)}$$

4.3 Quasi-Posterior Simulation

We propose the adaptive Sequential Monte Carlo (SMC) approach (Durham and Geweke, 2013) as a posterior simulation strategy, for two main reasons. First, many utility functions of the form in (2.1) give identical predictions. We cannot expect these functions to be close, in the sense of a practical MCMC chain passing from one to the other in a small number of steps with reasonable probability. Having many particles exploring the parameter space solves that problem. Second, a sequential inference method is required for our tempering approach.

The sequence of target distributions is obtained by setting $\zeta = 2$ for the first distribution, and by increasing ζ by a factor of 2 at each iteration:

$$Q(\psi|\{x^*_{nt}\},\zeta),\quad \text{for } \zeta = 2,4,8,\ldots$$

4.3.1 Algorithm

The algorithm uses a population of J groups of P particles, each of them containing a set of realizations of all our parameters and some simulated hyperparameters that do not have a closed form posterior:

$$\psi_{jp} = \left\{\alpha^{\theta}, \sigma^{\theta}, K_{+}, \left\{\Lambda_k, \pi^{z}_{k}, \left\{z_{nk}, s_{nk}, \theta_{nk}, \{\epsilon_{nkt}\}_{t=1}^{T}\right\}_{n=1}^{N}\right\}_{k=1}^{K_{+}}\right\}.$$

The algorithm starts with an Initialization (I) phase and then iterates over Correction (C), Selection (S) and Mutation (M) phases.

The first group of particles will be used to determine when to move from the C phase to the S phase based on Effective Sample Size (ESS). The ESS for the first group of particles will be denoted ESS_1. The simulated values of our first group of particles will be discarded in our analysis of the simulated posterior.

The general structure of the algorithm is as follow:

- Initialization phase: Initialize $J \times P$ particles $\{\{\psi_{jp}\}_{p=1}^{P}\}_{j=1}^{J}$
- For $\eta = 1$ to N *(Iterate over individuals)*
    - Draw $\{z_{\eta k}, s_{\eta k}, \theta_{\eta k}\}$ for the newly added individual (*see section 4.3.3*)
    - For $\tau = 1$ to T *(Iterate over observations for each individual)*
        - Draw $\epsilon_{\eta k \tau}$ for the newly added observation (*see section 4.3.3*)
        - Set $\zeta = 1$
        - Repeat until $\zeta = \infty$:
            - Correction phase: increase ζ until $ESS_1/P < 0.5$ or $\zeta = \infty$ (*see section 4.3.4*)
            - Selection phase: re-sample particles within each group based on weight
            - Mutation phase: move each particle with an Metropolis-Hastings within Gibbs step using the target density $Q(\psi_{jp}|\{x^{*}_{nt}\}; \eta, \tau, \zeta)$

The various parts of the algorithm are detailed in the following sections.

4.3.2 Initialization phase

Applied to every particle:

- Draw $\alpha^{\theta} \sim Gamma(1, 1)$
- Draw $\sigma^{\theta} \sim Gamma(1, 1)$
- Set $K_{+} = 0$

4.3.3 Drawing $\{z_{\eta k}, s_{\eta k}, \theta_{\eta k}\}$ and $\{\epsilon_{\eta k \tau}\}$

To draw the $z_{\eta k}$ we follow the IBP approach. As detailed in section 3.2, when Λ_k has an active monotone attribute, we set $m_k = 1$. When the monotone active attributes have upper active levels (all levels above a specified level are active), we set $\rho_k = 1$. When the monotone active attributes have lower active levels (all levels below a specified level are active), we set $\rho_k = 0$.

- For $k = 1$ to K_+
  - If $\sum_{n=1}^{\eta-1} z_{nk} > 0$:
    - Draw $z_{nk} \sim Bernoulli(\frac{\sum_{n=1}^{\eta-1} z_{nk}}{\beta^z+\eta-1})$
    - If $z_{nk} = 1$:
      - Draw $s_{\eta k} \sim \begin{cases} 0 & if\ m_k = 1 \text{ and } \rho_k = 1 \\ 1 & if\ m_k = 1 \text{ and } \rho_k = 0 \\ Bernoulli(\frac{1+\sum_{n=1}^{\eta-1} s_{nk}}{2+\eta-1}) & \text{otherwise} \end{cases}$
      - Draw $\theta_{\eta k} \sim Gamma(\sigma^\theta \alpha^\theta, 1)$
  - Else :
    - Set $z_{nk} = 0$
- If $K_+ = 0$:
  - While $k^* < 1$: Draw $k^* \sim Poisson(\frac{\alpha^z \beta^z}{\beta^z+\eta-1})$
- Else:
  - Draw $k^* \sim Poisson(\frac{\alpha^z \beta^z}{\beta^z+\eta-1})$
- Set $K_+ = K_+ + k^*$
- For $k = K_+ - k^*$ to K_+
  - Det $z_{\eta k} = 1$
  - Draw Λ_k
  - Draw $s_{\eta k} \sim \begin{cases} 0 & if\ m_k = 1 \text{ and } \rho_k = 1 \\ 1 & if\ m_k = 1 \text{ and } \rho_k = 0 \\ Bernoulli(\frac{1+\sum_{n=1}^{\eta-1} s_{nk}}{2+\eta-1}) & \text{otherwise} \end{cases}$
  - Draw $\theta_{\eta k} \sim Gamma(\sigma^\theta \alpha^\theta, 1)$

In order to draw the $\{\epsilon_{\eta k \tau}\}$ when a new observation is added, we follow the following simple algorithm:

- For $k = 1$ to K_+
  - If $z_{\eta k} = 1$: Draw $\epsilon_{\eta k \tau} \sim Gamma((1 - \sigma^\theta)\alpha^\theta, 1)$

4.3.4 Correction phase

Applied to every particle:

- Repeat until $ESS_1/P < 0.5$ or $\zeta = \infty$:
  - compute utility differences $du_{jp} = u(x^*_{\eta\tau}, \psi_{jp}) - \max_{x \in X_{\eta\tau}} u(x, \psi_{jp})$
  - If $\sum_p du_{1p} = 0$:
    - Set $\zeta = \infty$
    - compute weight $w_{jp} = \mathbb{1}\{du_{jp} = 0\}$
  - Else:
    - Set $\zeta = 2 \cdot \zeta$
    - compute weight $w_{jp} = w_{jp} \cdot 2^{du_{jp}}$
- compute $ESS_1/P = \frac{(\sum_p w_{1p})^2}{\sum_p w_{1p}^2}$

4.3.5 Selection phase

The selection phase draws a new set of P within group j particles with replacement from the set of old particles using weights w_{jp}. For each group j of particle:

- For each group j of particle:
  - For $p' = 1$ to P
    - Draw a particle ψ^* from set $\{\psi_{jp}\}_{p=1}^P$ using weights $\{w_{jp}\}_{p=1}^P$
    - Set $\psi_{jp'} = \psi^*$

4.3.6 Mutation phase

The mutation phase do an MCMC step using Gibbs sampling and Metropolis-Hastings (MH) accept-reject algorithm.

- For $n = 1$ to η
  - For $k = 1$ to K_+
    - If $\sum_{n' \neq n} z_{n'k} > 0$:
      - Set $\psi^* = \psi_{jp}$
      - Draw $z^*_{nk} \sim Bernoulli(\frac{\sum_{n' \neq n} z_{n'k}}{\beta^z + \eta - 1})$
      - If $z_{nk} = 1$: Draw $\{s^*_{nk}, \theta^*_{nk}, \{\epsilon^*_{nkt}\}\}$ (*see below*)
      - Store new values into ψ^*

        - Set $\psi_{jp} = \psi^*$ with probability $\min\{1, \frac{Q(\{x^*_{nt}\}|\psi^*;\eta,\tau,\zeta)}{Q(\{x^*_{nt}\}|\psi_{jp};\eta,\tau,\zeta)}\}$
      - Else :
        - Set $z_{nk} = 0$
  - Draw $k^* \sim Poisson(\frac{\alpha^z \beta^z}{\beta^z+\eta-1})$
  - Set $K_+ = K_+ + k^*$
  - For $k = K_+ - k^*$ to K_+
    - Set $\psi^* = \psi_{jp}$
    - Set $z^*_{nk} = 1$
    - Draw Λ^*_k
    - Draw $s^*_{nk} \sim \begin{cases} 0 & if\ \Lambda^*_k \text{ has active upper monotone atributes} \\ 1 & if\ \Lambda^*_k \text{ has active lower monotone atributes} \\ Bernoulli(\frac{1+\sum_{n=1}^{\eta-1} s_{nk}}{2+\eta-1}) & \text{otherwise} \end{cases}$
    - Draw $\theta^*_{nk} \sim Gamma(\sigma^\theta \alpha^\theta, 1)$
    - Store new values into ψ^*
    - Set $\psi_{jp} = \psi^*$ with probability $\min\{1, \frac{Q(\{x^*_{nt}\}|\psi^*;\eta,\tau,\zeta)}{Q(\{x^*_{nt}\}|\psi_{jp};\eta,\tau,\zeta)}\}$

- Draw $\alpha^* \sim Gamma(1, 1)$
- Draw $\sigma^* \sim Gamma(1, 1)$
- Set $(\alpha^\theta, \sigma^\theta) = (\alpha^*, \sigma^*)$ with probability

  $$\min\left\{1, \prod_k^{K_+} \prod_n^{\eta} \left(\frac{P_\Gamma(\theta_{nk}|\alpha^*\sigma^*, 1) \prod_t^{T_n} P_\Gamma(\epsilon_{nkt}|\alpha^*(1-\sigma^*), 1)}{P_\Gamma(\theta_{nk}|\alpha^\theta\sigma^\theta, 1) \prod_t^{T_n} P_\Gamma(\epsilon_{nkt}|\alpha^\theta(1-\sigma^\theta), 1)} \right)^{z_{nk}} \right\}$$

  where $P_\Gamma(\cdot|\alpha, \beta)$ is the PDF of the $Gamma(\alpha, \beta)$ distribution, and $T_n = \tau$ if $n = \eta$ and $T_n = T$ otherwise.

For this phase, in order to draw $\{s^*_{nk}, \theta^*_{nk}, \{\epsilon^*_{nkt}\}\}$, we propose the following process:

- Draw $s^*_{nk} \sim \begin{cases} 0 & if\ m^*_k = 1 \text{ and } \rho^*_k = 1 \\ 1 & if\ m^*_k = 1 \text{ and } \rho^*_k = 0 \\ Bernoulli(\frac{1+\sum_{n' \neq n}^{\eta} s_{nk}}{2+\eta-1}) & \text{otherwise} \end{cases}$
- Draw $\theta^*_{nk} \sim Gamma(\sigma^\theta \alpha^\theta, 1)$
- If $n < \eta$:
  - For $t = 1$ to T:
    - $\epsilon^*_{nkt} \sim Gamma((1-\sigma^\theta)\alpha^\theta, 1)$
- Else:
  - For $t = 1$ to τ:
    - $\epsilon^*_{nkt} \sim Gamma((1-\sigma^\theta)\alpha^\theta, 1)$

References

V. Chernozhukov and H. Hong. An MCMC approach to classical estimation. *Journal of Econometrics*, 115(2):293–346, 2003.

P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(3):411–436, 2006.

G. Durham and J. Geweke. Adaptive sequential posterior simulators for massively parallel computing environments. *Available at SSRN 2251635*, 2013.

T. L. Griffiths and Z. Ghahramani. The indian buffet process: An introduction and review. *The Journal of Machine Learning Research*, 12:1185–1224, 2011.