# A Bayesian Dual Clustering Approach
# for Selecting Data and Parameter Granularities

Mingyung Kim, Eric T. Bradlow and Raghuram Iyengar[1]

## Abstract

While there are well-established *model* selection methods (e.g., BIC), they commonly condition on a priori selected data and parameter granularities. That is, researchers think they are doing model selection, but what they are really doing is model selection *conditional* on their chosen granularities. We propose a new method, *Bayesian dual clustering* (BDC), that infers data and parameter granularities by sampling over their posterior distribution. BDC entails representing data and parameters as two separate collections of nodes (e.g., SKUs) with each node being the unit of analysis. Then (a) each collection is clustered using a covariate-driven distance function that allows for a high degree of interpretability for the underlying drivers and (b) data and parameter granularity posteriors are inferred. Notably, BDC can (c) accommodate parameter restrictions using a split-merge sampler, (d) handle large collections, and (e) relate to other extant methods (e.g., latent-class analysis). We apply BDC to a frequently purchased grocery category. The results show that BDC inferred granularities, as compared to those from extant approaches, impact demand elasticities and optimal actions. We conclude by highlighting the generalizability of BDC to a broad array of marketing problems.

**Keywords:** Data granularity, parameter granularity, clustering, Bayesian non-parametrics

[1] * Mingyung Kim is an Assistant Professor at the Ohio State University (kim.9572@osu.edu). Eric T. Bradlow is the K.P. Chao Professor, Professor of Marketing, Economics, Education, Statistics and Data Science at the Wharton School of the University of Pennsylvania (ebradlow@wharton.upenn.edu). Raghuram Iyengar is Miers-Busch, W'1885 Professor, Professor of Marketing at the Wharton School of the University of Pennsylvania (riyengar@wharton.upenn.edu). Researcher(s) own analyses calculated (or derived) based in part on data from Nielsen Consumer LLC and marketing databases provided through the NielsenIQ Datasets at the Kilts Center for Marketing Data Center at The University of Chicago Booth School of Business. The conclusions drawn from the NielsenIQ data are those of the researcher(s) and do not reflect the views of NielsenIQ. NielsenIQ is not responsible for, had no role in, and was not involved in analyzing and preparing the results reported herein.

## 1. Introduction

One of the fundamental, and academically well researched, decisions that a marketing manager routinely makes is how to model the relationship between firm-controlled covariates (e.g., price) and an outcome of interest (e.g., sales) while controlling for macroeconomic drivers. This problem is commonly denoted as *model selection*, for which there have been many solutions proposed that compare model fit (e.g., BIC, marginal likelihood) across models to "select the winner". While the above is typically described as "model selection", we suggest that researchers are actually conducting "conditional" model selection, based on a priori chosen levels of data and parameter granularities. The former refers to decisions about whether to employ the most granular data available (e.g., weekly) or aggregate it to a coarser level (e.g., monthly). Similarly, the latter refers to decisions about whether to vary parameters across the most granular units (e.g., across individuals) or at a more aggregate level (e.g., across zip codes). In this paper, it is the inference for these two "big" decisions (aforementioned data aggregation[2] and parameter granularity) that we seek to address. While many past studies have demonstrated that empirical results (e.g., price elasticity) and the corresponding marketing decisions (e.g., optimal price) *vary* based on the chosen levels of data and/or parameter granularities (e.g., Bucklin and Gupta 1999; Christen et al. 1997), how to empirically *select* their levels (and the accompanying uncertainty in them) is an important gap in the literature that our research fills.

Marketing managers often select the levels of data and parameter granularities based on common practices. For example, SKU-level data are typically aggregated to the brand-size level (e.g., Dubé and Gupta 2008; Hoch et al. 1995) or to the brand level (e.g., Ataman, Van Heerde, and Mela 2010). However, relying on such heuristics may lead to problems. For data granularity, using overly coarse (aggregated) data may induce aggregation bias (e.g., Christen et al. 1997), whereas using overly granular data may consist of substantial noise, making it difficult to reliably uncover systematic relationships (e.g., Kim, Bradlow, and

[2] We utilize the terms data aggregation (which is standard) and data granularity interchangeably.

Iyengar 2022). For instance, when sales data are collected at a per-second frequency, much of the variation at that level may reflect customer behaviors that cannot be explained by observed variables (e.g., price, advertising), and this unexplained variation may get averaged out (in a favorable prediction sense) in coarser data. Similarly, for parameter granularity, using overly coarse parameters (e.g., a single price elasticity across SKUs) may ignore important heterogeneity, whereas overly granular parameters (e.g., one price elasticity per SKU) may lead to sparse data problems and provide imprecise parameter estimates (e.g., Fader and Hardie 1996). Despite the importance of these trade-offs, there is little guidance on how to select (and understand the uncertainty of) levels of data and parameter granularities that best fit the data. This motivates the need for a systematic approach that balances these trade-offs and selects (1) data granularity that reduces noise while avoiding aggregation bias and (2) parameter granularity that captures necessary heterogeneity while avoiding sparse data problems. Our research addresses this need by developing a method that allows researchers to infer both data and parameter granularities.

As a solution, we propose what-we-call *Bayesian dual clustering* (BDC). The proposed method, as the clustering name suggests, represents data and parameters as separate collections of nodes. Then, it probabilistically clusters the nodes in each collection to infer the levels of data and parameter granularities. One notable contribution is that by representing both data and parameters in a generalized manner (i.e., using nodes), BDC is *flexible* and can accommodate differing units of analysis with little modification (e.g., time, space, people or SKUs as nodes), making it a highly generalizable tool for marketing researchers. To cluster the dual (data and parameter) collections, we develop a novel extension of the Bayesian non-parametric clustering method, the distance-dependent Chinese Restaurant Process (ddCRP; Blei and Frazier 2011), originally used to cluster texts and pixels in a document and an image, respectively (e.g., Arfa, Yusof, and Shabanzadeh 2019; Ghosh et al. 2011). Unlike document and image segmentation, many marketing applications may need to link (restrict) the data and parameter clusterings – e.g., the data must be at least as granular as the parameters. To accomplish this restriction, BDC relates the data and parameter clusterings by introducing a split-merge sampler, which we explain in the Methodology section.

There are two other points about our method that are noteworthy. First, BDC offers a high degree of *interpretability* as to the underlying drivers of data and parameter clusters by relating distances between nodes in the respective collections to *observed* attributes of the unit of analysis (e.g., brand or package size for an SKU-level demand analysis). Second, BDC can handle large data sets that are becoming more prevalent in marketing, demonstrating that our method can be applied to a broad set of (large) problems.

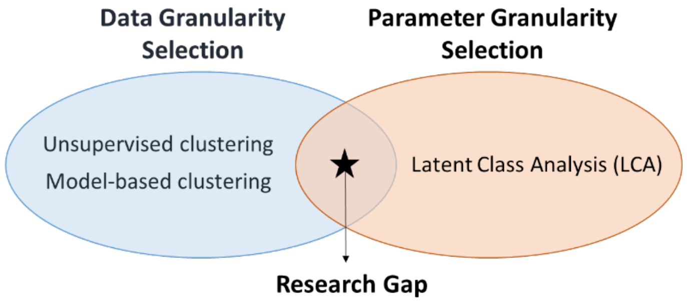
Notably, extant data and parameter clustering methods select *either* parameter or data granularity while fixing the level of the other (see Figure 1). Extant *parameter* clustering methods like latent-class analysis (LCA)[3] determine the level of parameter granularity of a demand model while fixing the level of data granularity (typically, at the most granular level). In this respect, BDC, when its level of data granularity is fixed, reduces to a version of LCA that imposes our ddCRP prior on parameter clustering. Thus, BDC nests this version of LCA. Some extant studies have applied versions of LCA that impose different Bayesian non-parametric priors on parameter clustering, and BDC relates to some of them. A notable example is the location-scale partition (LSP) prior proposed by Smith, Rossi, and Allenby (2019). While they imposed the LSP prior on SKU clustering to constrain cross-price elasticities at a chosen cluster level, it could be applied to constrain demand parameters (e.g., intercept, price elasticity, advertising elasticity), as in our paper. While our ddCRP prior looks different from the LSP prior on the surface, it in fact extends the LSP prior, which we explain in detail in the Methodology section and in Online Appendix C. Thus, our approach relates to other parameter clustering methods that are currently in the literature.

In contrast, extant *data* clustering methods are typically unsupervised and combine data points that have similar attributes without taking a final demand model into account. For instance, Morwitz and Schmittlein (1992) employed K-means to cluster households with respect to their characteristics (e.g., demographics and past purchase behaviors) and aggregated data for households in the same cluster. An

[3] We refer to LCA as a method that clusters units (e.g., SKUs, customers) into homogeneous groups and constrains parameters at the group level (e.g., Kamakura and Russell 1989), rather than as a method that reduces dimensions of attributes (such as PCA or factor analysis).

exception is the scaled power likelihood with multiple weights (SPLM), a Bayesian model-based data clustering method proposed by Kim, Bradlow, and Iyengar (2022). Unlike unsupervised data clustering methods, it accounts for a final demand model and determines the level of data granularity of a demand model while fixing the level of parameter granularity (e.g., at the coarsest level). This objective is what BDC would attain if the parameter granularity is fixed. Note that BDC, even when parameter granularity is held fixed, is more useful than SPLM in contexts where candidate data granularities are not clearly defined (e.g., how to aggregate SKUs or customers). We explain this in more detail in the Methodology section.

**Figure 1:** Research Gap


Data Granularity Selection
Parameter Granularity Selection
Unsupervised clustering
Model-based clustering
Latent Class Analysis (LCA)
★
Research Gap

*Notes*. Data granularity selection methods are either unsupervised (e.g., Morwitz and Schmittlein 1992) or model-based (e.g., Kim, Bradlow, and Iyengar 2022). Parameter granularity selection methods consist of LCA (e.g., Smith, Rossi, and Allenby 2019).

To assess the performance of our method (BDC) on several metrics of relevance to both theory and practice, and compare it to those from extant methods, we conduct an extensive simulation study. We find that the bias in the coefficients of estimated demand models (e.g., price elasticity) is smaller under BDC than under extant data or parameter clustering methods. An in-depth analysis reveals that a key driver of these results is that BDC performs better in recovering the underlying true levels of data and parameter granularities as compared to extant approaches. Unlike BDC, extant parameter clustering methods select overly granular parameters, because they select parameter granularity while ("erroneously" or by default) conditioning on the most granular data. In a similar vein, extant data clustering methods select overly coarse data granularity, because they select data granularity while (erroneously) conditioning on the coarsest parameters. Notably, this pattern of results holds for different true data generating processes and for both small and large collections of nodes (e.g., with thousands of SKUs), thus highlighting the generalizability of the BDC's superior performance.

We apply BDC to a large-scale Nielsen scanner dataset containing SKU-store-quarter-level sales and marketing actions (price and advertising) of the juice and drinks category and compare its performance on a few metrics (e.g., model fit) with that from extant methods. We provide three key findings. First, BDC provides a significantly better (in-sample and out-of-sample) model fit than extant methods do. This result implies that it is important to select the levels of *both* data and parameter granularities along with the model itself. Second, the levels of data and parameter granularities inferred from our method differ from those inferred from extant methods. It is because unlike our method, extant methods select either data or parameter granularity while *conditioning* on the other. Lastly, the inference of demand parameters (e.g., price elasticity) based on our method differ substantially from those obtained by employing alternative methods.

Although we apply the proposed method to demand analysis for SKUs, researchers make decisions regarding data and parameter granularities in many other contexts. One example is a temporal or spatial analysis of demand, for which researchers select the temporal or spatial unit of analysis and build a model conditional on their choices. In most cases, they decide whether to use the most granular data (e.g., daily data) or aggregate it to a coarser level (e.g., weekly level) for analysis. Another example is customer (or group) level analysis, for which researchers often cluster customers and then aggregate data (and/or parameters) based on the chosen clustering. Our framework is flexible (as explained above) and can be easily extended to such contexts.

The remainder of the paper is as follows. In Section 2, we propose BDC as a tool for data and parameter granularities and compare it with extant data or parameter clustering methods. Section 3 describes a large-scale simulation study that assesses the performance of BDC in selecting data and parameter granularities as compared to other extant approaches. In Section 4, we present an application of BDC to a large-scale data set containing SKU purchases. Section 5 concludes with how BDC contributes to conceptual and substantive issues in marketing, as well as limitations and future research directions.

## 2. Methodological Framework

We first lay out a general overview of our methodological framework (2.1) and the prior distribution chosen for inferring data and parameter granularities (2.2). We then discuss the proposed framework in detail (2.3), its computational efficiency (2.4), and how it compares to extant data or parameter clustering methods (2.5).

### 2.1. General Overview of the Methodological Framework

Figure 2 provides an overview of our framework. There are four key steps. First, for a given context, we assemble the most granular level of data available (e.g., SKU-level sales data). Let $N^1$ indicate the number of observations in this dataset with the superscript "1" denoting the most granular data. Similarly, suppose that the model employed in the analysis is also specified with parameters at the most granular level (e.g., observation-specific parameters corresponding to the most granular data with $N^1$ observations). To enhance model flexibility and its use across different units of analysis (e.g., time, space, people, or SKUs), we represent the most granular data and parameters as two collections of nodes – a data collection and a parameter collection, respectively – with each unit being a node.[4]

Second, we probabilistically cluster nodes in both the data and parameter collections. We sample the posterior distribution of the data clustering (denoted as $D$) and parameter clustering (denoted as $M$) which recognizes and utilizes their uncertainty. By doing so, we extend typical statistical modeling, which conditions on the choice of $D$ and $M$. We accomplish this dual clustering (as per the title of the paper) by

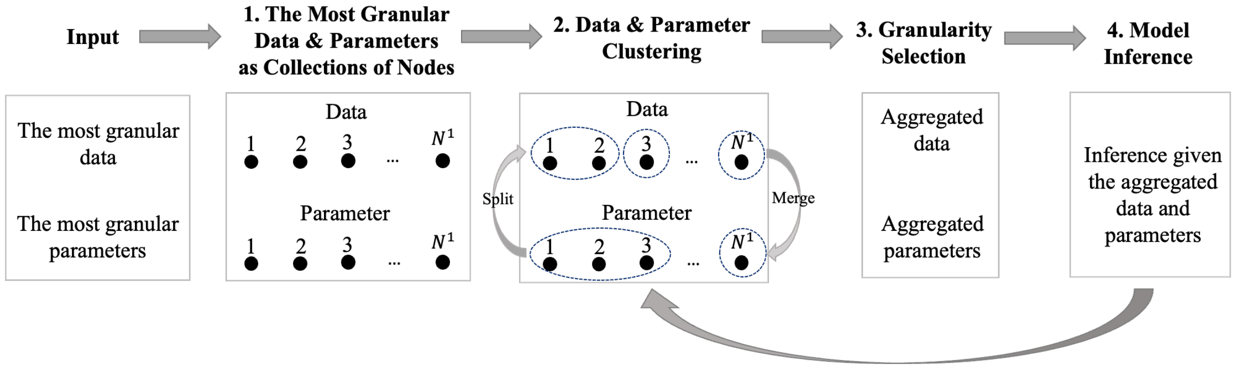
[4] While we focus on two collections for exposition, our method can easily handle multiple. For instance, we can allow different parameters (e.g., intercept, price elasticity, and advertising elasticity) to have different granularities by allowing three (instead of one) parameter collections.

building on an extant clustering method that imposes the distance-dependent Chinese restaurant process (ddCRP; Blei and Frazier 2011) prior on clusterings.

Third, we aggregate data and parameters based on the chosen clusters. Specifically, we aggregate data by summarizing (e.g., summing, averaging) the data for nodes in the same data cluster, whereas we aggregate parameters by setting the parameters equal for nodes in the same parameter cluster. In addition, while inference with parameters more heterogeneous than the data has been addressed in the literature (e.g., Chen and Yang 2007; Musalem, Bradlow, and Raju 2008), we focus here on cases where the data clustering $D$ nests the parameter clustering $M$. While our method can conceptually handle more general clusterings, exploring its properties in conjunction with the ddCRP is beyond the scope of this project. To accomplish this "restriction" (and other similar ones, whether managerial or theory based), we introduce a split-merge sampler. As the name suggests, the sampler is composed of split and merge steps (details in Section 2.3), which we follow (in order) to easily enforce this restriction at each iteration of the MCMC sampling.

Lastly, we conduct standard model and parameter inference conditional on the aggregated data and parameters. Notably, the last step indicates that our selection of data and parameter granularities (Step 3) is connected to the inference of the resulting model (Step 4). To infer data and parameter clusterings that sample from the posterior distribution of the resulting model, we iterate the steps 2–4 until convergence.

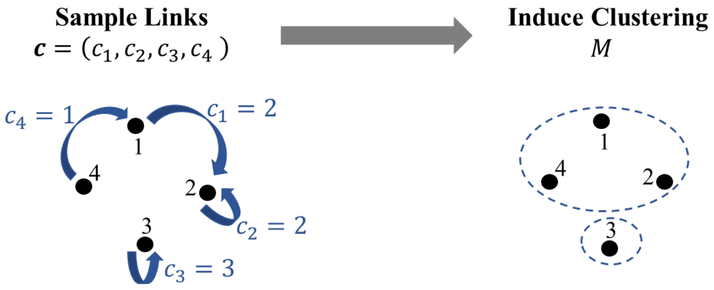
**Figure 2**: Methodological Framework


Input
1. The Most Granular Data & Parameters as Collections of Nodes
2. Data & Parameter Clustering
3. Granularity Selection
4. Model Inference
The most granular data
The most granular parameters
Data
1 2 3 ... $N^1$
Parameter
1 2 3 ... $N^1$
Split
Merge
Aggregated data
Aggregated parameters
Inference given the aggregated data and parameters

## 2.2. Distance-Dependent Chinese Restaurant Process (ddCRP)

The ddCRP is a non-parametric probability distribution over clusters. It represents the input of interest (e.g., an image) as a collection of nodes, with each unit (e.g., a pixel from the image) being a node. Then, as in Figure 3, it defines the probability distribution over clusterings (denoted as $M$) by iterating across nodes and, for each node, defining the probability of choosing its link (denoted as $c_i$).

**Figure 3**: Illustration of the Probability Distribution over Clustering $M$



Formally, the probability distribution of $M$ is denoted as:

$$\pi(M) = \prod_i \pi(c_i) \tag{1}$$

where the probability of linking node $i$ to itself (denoted as $c_i = i$) is proportional to a self-link parameter $\alpha$, and the probability of linking node $i$ to another node $i'$ (denoted as $c_i = i'$) depends on their distance. This leads to the following multinomial distribution conditional on the self-link parameter $\alpha$, the pairwise distance $dist_{ii'}$, and a decay function $f(.)$

$$\pi(c_i) \propto \begin{cases} \alpha & \text{if } c_i = i \\ f(dist_{ii'}) & \text{if } c_i = i' \end{cases} \tag{2}$$

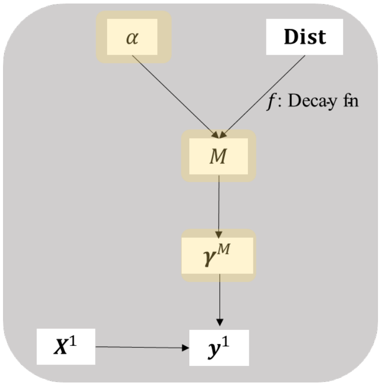
The decay function is non-negative, non-increasing, and $f(\infty) = 0$. Several decay functions (e.g., exponential decay) meet these conditions, and (without loss of generality) we assume the exponential decay function (e.g,, Arfa, Yusof, and Shabanzadeh 2019; Blei and Frazier 2011). Extant studies (outside of marketing) have employed the ddCRP distribution as a prior for latent class analysis (LCA). They impose the non-parametric prior on parameter clustering while using the most granular data available (e.g., Arfa, Yusof, and Shabanzadeh 2019; Blei and Frazier 2011; Ghosh et al. 2011).

The extant ddCRP-based latent class analysis (hereafter, extant ddCRP-LCA) is composed of three steps as in Figure 4 and iterates these steps until convergence. First, one samples a self-link probability parameter $\alpha$ and then parameter clustering ($M$). The posterior probability of $M$ is denoted as:

$$p\left(M|\boldsymbol{y}^{\mathbf{1}}, \boldsymbol{X}^{1}\right) \propto \pi(M) \cdot p\left(\boldsymbol{y}^{\mathbf{1}}|\boldsymbol{X}^{1}, M\right) \tag{3}$$

where $\pi(M)$ is the ddCRP prior for $M$ (see Equations (1)–(2)) and $p\left(\boldsymbol{y}^{\mathbf{1}}|\boldsymbol{X}^{1}, M\right)$ is the marginal likelihood given $M$ and $(\boldsymbol{y}^{1}, \boldsymbol{X}^{1})$, the most granular data (with superscript 1, as before). Next, one clusters the parameters given the sampled $M$. Specifically, one sets the parameters equal for nodes in the same cluster and denotes the clustered parameters as $\boldsymbol{\gamma}^{M}$. Finally, one conducts the standard parameter inference conditional on the observed data and the clustered parameters and computes the corresponding posterior probability.

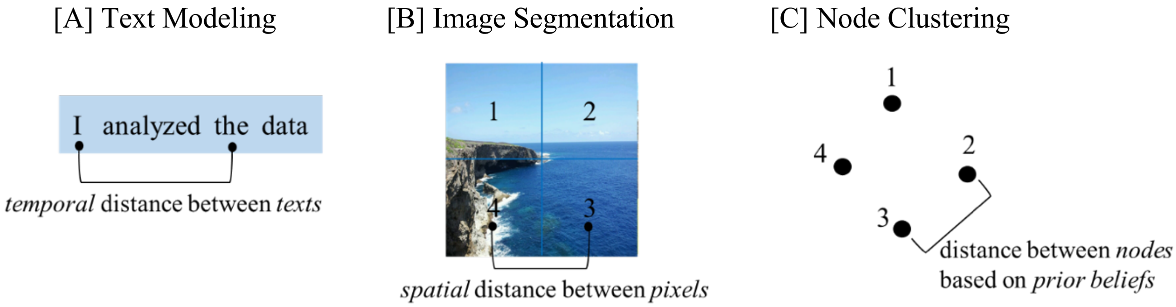
**Figure 4:** Directed Graph for the extant ddCRP-LCA



*Note.* We represent in yellow the parameters – $\alpha$ (self-link parameter), $M$ (parameter clustering), and $\boldsymbol{\gamma}^{M}$ (model parameters) – that are sampled in the extant ddCRP-LCA.

Note that the extant ddCRP-LCA has mainly been applied for segmenting a document or an image. A notable feature of the method is the ease with which prior beliefs relevant for segmentation can be accommodated via a distance function. For instance, extant studies use the distance between texts (pixels) as an input in a model for determining which texts (pixels) should be grouped together (Arfa, Yusof, and Shabanzadeh 2019; Blei and Frazier 2011; see Figures 5[A] and 5[B]). Similarly, one can employ the ddCRP to capture prior beliefs for how the nodes of a collection (e.g., parameter collection) would be aggregated based on their inter-node distance. Extant studies assume that the inter-node distance is known a priori, which makes sense for certain applications like document (image) segmentation, in which the actual distance between texts (pixels) is known (e.g., Arfa, Yusof, and Shabanzadeh 2019; Blei and Frazier 2011; Ghosh et al. 2011). However, this assumption makes less sense for many other problems in marketing like the one we study. For instance, for the problem of how best to segment SKUs, the nodes are SKUs and how to define the inter-node distance is less straightforward.

**Figure 5:** Distances in Text Modeling and Image Segmentation Versus Node Clustering



[A] Text Modeling

[B] Image Segmentation

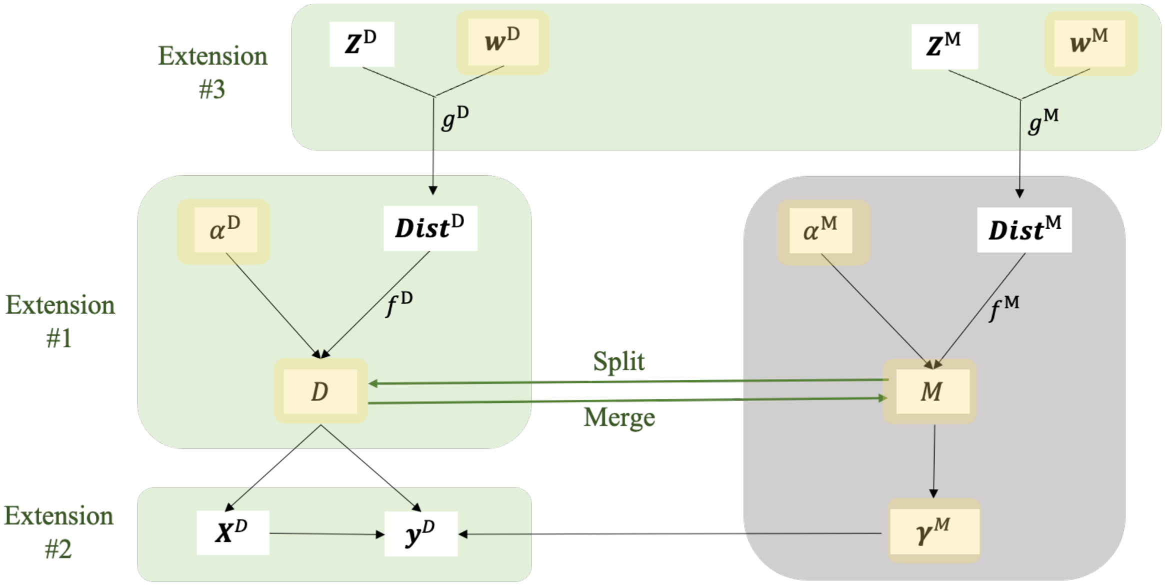
[C] Node Clustering

I analyzed the data
*temporal* distance between *texts*
1
2
4
3
*spatial* distance between *pixels*
1
4
2
3
distance between *nodes* based on *prior beliefs*

## 2.3. Bayesian Dual Clustering (BDC)

Our proposed method (BDC) extends the extant ddCRP-LCA (which is for parameters only) in three ways. Figure 6 illustrates our key extensions (shaded green) as compared to the extant ddCRP-LCA (shaded gray). We explain each key extension (and the corresponding vertices and edges in Figure 6) in the remaining part of this section. We provide BDC's pseudocodes and relate them to Figure 6 in Online Appendix A.

**Figure 6:** Directed Graph for the BDC



*Notes.* Figure 6 illustrates key extensions (in green) as compared to the extant ddCRP-LCA (in gray). We represent parameters sampled in our proposed method – e.g., $M$ (parameter clustering) – in yellow. We describe the role of each parameter in the main text of this section.

**(1) Allowing for dual (parameter and data) clustering**

Extant applications of the ddCRP-LCA cluster only a single collection, the parameter nodes. To cluster two collections (data and parameter) and capture any synergy across them, as illustrated in Figure 6, we allow for two sets of ddCRPs – one for parameter clustering (denoted as $M$) and the other for data clustering (denoted as $D$). Note, as before, we do not allow $M$ to be more granular than $D$. We accommodate this relationship (restriction) by linking $D$ and $M$ via a *split-merge* MCMC sampler as follows, which is an additional computational contribution of our research. The sampler is composed of the split and merge steps (details below), which we follow in order in each MCMC iteration.

*Split sampler.* To ensure that $D$ is at least as granular as $M$, we sample $D$ by *splitting* each parameter cluster in a previously sampled $M$. Specifically, the split sampler is composed of two steps as illustrated in Figure 7[A]. First, we treat each parameter cluster as an independent data sub-collection. For instance, in Figure 7[A], there are two parameter clusters, and we treat each cluster as a data sub-collection. As a next step, within each data sub-collection, we sample data clustering ($D$) by iterating over nodes with each node being sampled to form a link (denoted as $c_i^{\mathrm{D}}$). Thus, formally, the prior distribution of $D$ is denoted as:

$$\pi(D) = \prod_i \pi\left(c_i^{\mathrm{D}}\right) \tag{4}$$

where:

$$\pi\left(c_i^{\mathrm{D}}\right) \propto \begin{cases} \alpha^{\mathrm{D}} & \text{if } c_i^{\mathrm{D}} = i \\ f^{\mathrm{D}}\left(dist_{ii'}^{\mathrm{D}}\right) & \text{if } c_i^{\mathrm{D}} = i' \end{cases} \tag{5}$$

Note that Equations (4) and (5) are equivalent to the original ddCRP prior in Equations (1) and (2) except that they have the superscript D to denote that the ddCRP prior is imposed for data clustering.

Merge sampler. To ensure that $M$ is at most as granular as $D$, we sample $M$ by *merging* data clusters in a previously sampled $D$. The merge sampler is composed of two steps as illustrated in Figure 7[B]. First, we treat the clustered data ($D$) as a parameter collection and then treat each data cluster as a node in the parameter collection. For instance, in Figure 7[B], there are three data clusters, and we treat these three clusters as nodes in the parameter collection. As a next step, we sample parameter clustering ($M$) by iterating each data cluster $d$ and sampling a data cluster to link itself to (denoted as $c_d^{\mathrm{M}}$). Thus, formally, the prior distribution of $M$ is denoted as:

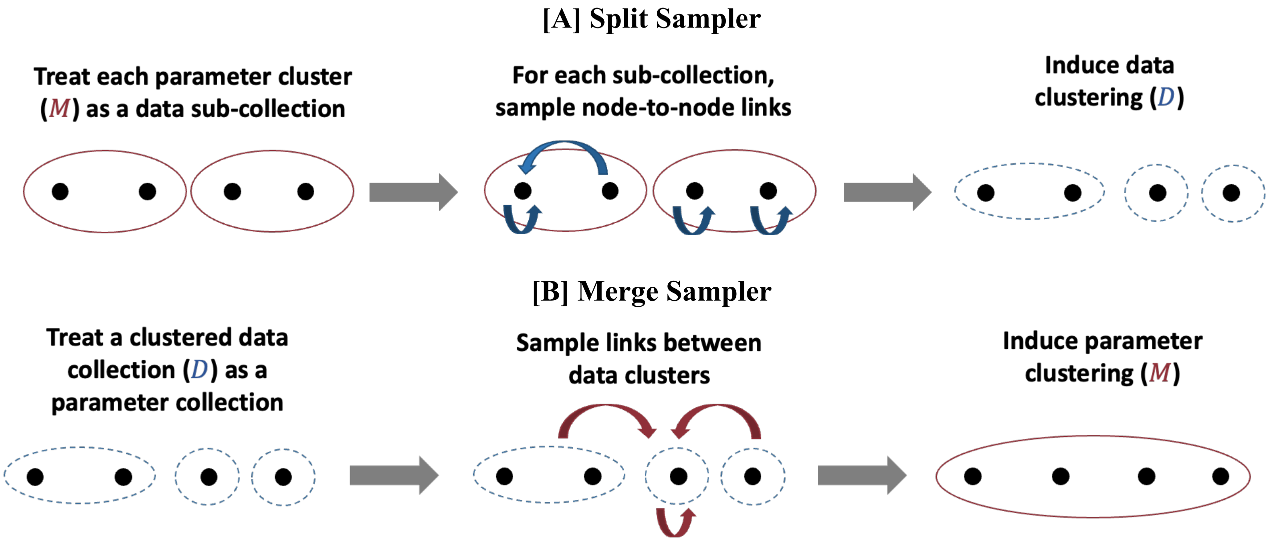
$$\pi(M) = \prod_d \pi\left(c_d^{\mathrm{M}}\right) \tag{6}$$

where:

$$\pi\left(c_d^{\mathrm{M}}\right) \propto \begin{cases} \alpha^{\mathrm{M}} & \text{if } c_d^{\mathrm{M}} = d \\ f^{\mathrm{M}}\left(dist_{dd'}^{\mathrm{M}}\right) & \text{if } c_d^{\mathrm{M}} = d' \end{cases} \tag{7}$$

Note that Equations (6) and (7) are equivalent to the original ddCRP prior in Equations (1) and (2) with two key differences. First, they have the superscript M to denote that the ddCRP prior is imposed for parameter clustering. Second, since they cluster data clusters instead of individual nodes, it uses $dist_{dd'}^{\mathrm{M}}$ (distance between data clusters $d$ and $d'$) instead. Note that $dist_{dd'}^{\mathrm{M}}$ is a summary (e.g., average is commonly used) of the distances between all the pairs of nodes in the clusters $d$ and $d'$.

**Figure 7:** Illustration of the Proposed Split-Merge Sampler


[A] Split Sampler
Treat each parameter cluster ($M$) as a data sub-collection
For each sub-collection, sample node-to-node links
Induce data clustering ($D$)
[B] Merge Sampler
Treat a clustered data collection ($D$) as a parameter collection
Sample links between data clusters
Induce parameter clustering ($M$)

### (2) Modeling distances between nodes and its interpretability

Extant studies on ddCRP assume that the pairwise distance between nodes is known a priori (see Section 2.2). This assumption is reasonable in applications of text and image segmentation, where the *actual distance* between words and between pixels, respectively, is known. However, in other problems relevant for marketers this assumption may not capture all the nuances of the context. For instance, consider the problem of how best to segment SKUs. In this context, there is no *literal distance* between two SKUs. Thus, we define the distance between nodes $i$ and $i'$ ($dist_{ii'}^{\mathrm{D}}$) as a function of the weighted average of the differences in their observed attributes (denoted as $\mathrm{diff}(\boldsymbol{z}_i^{\mathrm{D}}, \boldsymbol{z}_{i'}^{\mathrm{D}})$) and a vector of weight parameters ($\boldsymbol{w}^{\mathrm{D}}$) whose posterior distributions are inferred in conjunction with the clusterings.

$$dist_{ii'}^{\mathrm{D}} = g^{\mathrm{D}}\left(\boldsymbol{w}^{\mathrm{D}} \cdot \mathrm{diff}(\boldsymbol{z}_i^{\mathrm{D}}, \boldsymbol{z}_{i'}^{\mathrm{D}})\right) \quad (8)$$

Here, a link function $g^{\mathrm{D}}(.)$ translates the weighted differences into distances; hence, it is a non-negative and increasing function. Several types of link functions (e.g., exponential) meet these conditions, and (without loss of generality) we assume the exponential link function. Note that we define a distance between nodes $i$ and $i'$ in the parameter collection in a similar way. Specifically, we define the distance between nodes $i$ and $i'$ ($dist_{ii'}^{\mathrm{M}}$) as a function of the weighted average of the differences in their observed attributes (denoted as $\mathrm{diff}(\boldsymbol{z}_i^{\mathrm{M}}, \boldsymbol{z}_{i'}^{\mathrm{M}})$) and a vector of the corresponding weight parameters ($\boldsymbol{w}^{\mathrm{M}}$).

$$dist_{ii'}^{\mathrm{M}} = g^{\mathrm{M}}\left(\boldsymbol{w}^{\mathrm{M}} \cdot \mathrm{diff}(\boldsymbol{z}_i^{\mathrm{M}}, \boldsymbol{z}_{i'}^{\mathrm{M}})\right) \tag{9}$$

Note that the weight parameters ($\boldsymbol{w}^{\mathrm{D}}, \boldsymbol{w}^{\mathrm{M}}$) are identifiable even when the observed attributes for parameter clustering ($\boldsymbol{z}_i^{\mathrm{M}}$) are identical to those for data clustering ($\boldsymbol{z}_i^{\mathrm{D}}$). As before, we set $g^{\mathrm{M}}$ as the exponential function. Finally, a notable contribution of our method is that the latent weight parameters in the distance functions enhance the interpretability of the results as it helps to explain *why* certain levels of data and parameter granularities are chosen which will be emphasized in our data application.

### (3) Making the likelihood comparable across the levels of data aggregation

One (popular) extant method (as discussed), the extant ddCRP-LCA, fixes the data (e.g., document, image) at the most granular level and clusters parameters in a way that fits the data well. We denote the posterior probability of choosing $M$ (and aggregating the most granular parameters accordingly) in Equation (3). In contrast, our work proposes to cluster and aggregate both model parameters and the data. The latter allows for the possibility that the most granular data may not be the one best fitted for the focal analysis conditional on the parameter granularity ($M$). Hence, the posterior probability of choosing $D$ and $M$ together (and aggregating the most granular data and parameters, respectively) is denoted as:

$$p(D, M|\boldsymbol{y}^1, \boldsymbol{X}^1) \propto \pi(D, M) \cdot p(\boldsymbol{y}^1|\boldsymbol{X}^1, D, M) \tag{10}$$

where $\pi(D, M)$ is the ddCRP prior for $(D, M)$, and $p(\boldsymbol{y}^1|\boldsymbol{X}^1, D, M)$ is the marginal likelihood given $(D, M)$ and $(\boldsymbol{y}^1, \boldsymbol{X}^1)$.

It is important to note that the likelihood (and so the posterior probability) in Equation (10) is *not comparable* across different levels of data aggregation ($D$) and hence cannot be used directly. In particular, the total likelihood multiplies individual likelihood terms over the *number* of observations. Hence, the likelihood (and so the posterior probability) is higher for coarser data as it has fewer likelihood terms. To solve this issue, as originally addressed in Kim, Bradlow, and Iyengar (2022), we note that $p(\boldsymbol{y}^1|\boldsymbol{X}^1, D, M)$ in Equation (10) actually represents the marginal likelihood for aggregated data $(\boldsymbol{y}^D, \boldsymbol{X}^D)$, which aggregates $(\boldsymbol{y}^1, \boldsymbol{X}^1)$ based on $D$, and so can be denoted more formally as $p_D(\boldsymbol{y}^D|\boldsymbol{X}^D, M)$:

$$p(D, M|\boldsymbol{y}^1, \boldsymbol{X}^1) \propto \pi(D, M) \cdot p_D(\boldsymbol{y}^D|\boldsymbol{X}^D, M) \tag{11}$$

Then, to make the marginal likelihood (and so the posterior probability) comparable, we scale the marginal likelihood $p_D(\boldsymbol{y}^D|\boldsymbol{X}^D, M)$ in Equation (11) to the same data granularity (particularly, to the most granular data):

$$p_{1(D)}(D, M|\boldsymbol{y}^1, \boldsymbol{X}^1) \propto \pi(D, M) \cdot p_{1(D)}(\boldsymbol{y}^D|\boldsymbol{X}^D, M) \tag{12}$$

where $p_{1(D)}(\boldsymbol{y}^D|\boldsymbol{X}^D, M)$ and $p_{1(D)}(D, M|\boldsymbol{y}^1, \boldsymbol{X}^1)$ indicate the scaled marginal likelihood and the scaled posterior probability, respectively.

Then, how can we scale the likelihood to the finest data granularity? Kim, Bradlow, and Iyengar (2022) proposed to scale the likelihood at $D$ to the most granular level by following two steps: (1) scale the dependent variable $(\boldsymbol{y}^D)$ to the most granular level by dividing it with the number of nodes in each cluster and (2) replace $\boldsymbol{y}^D$ in each likelihood term with the scaled value. This divisible scaling is applicable only for distributions (e.g., normal, log–normal) that retain their functional forms after scaling.

However, this divisible scaling cannot be applied to several distributions commonly used in Marketing (e.g., Poisson, Binomial). To this end, we introduce probabilistic scaling that can be applied to any type of distribution. Online Appendix B provides more mathematical details on the probabilistic scaling, but we note here that it is composed of three steps: (1) given observed aggregated data, draw a set of possible most granular datasets, (2) compute the likelihood given each of the granular datasets, and (3) compute the scaled likelihood by averaging the likelihoods for these granular datasets.

## 2.4. Computational efficiency

Our proposed method (BDC) is both methodologically novel and computationally efficient and hence practical for real-world applications. Notably, the ddCRP sampler, which our method builds upon, exhibits rapid mixing and convergence. Since updating even a single link can substantially change the clustering, the ddCRP sampler can propose significant moves in each iteration (Blei and Frazier 2011). This feature has been demonstrated in various applications (e.g., Arfa, Yusof, and Shabanzadeh 2019; Ghosh et al. 2011). For instance, Arfa, Yusof, and Shabanzadeh (2019) clustered 100 pixels in a trajectory map and reported convergence within 10 iterations. Similarly, Ghosh et al. (2011) clustered 1,000 image pixels and achieved convergence within 50 iterations. Our simulations exhibit similar rapid mixing and convergence.

### 2.5. Related Methods

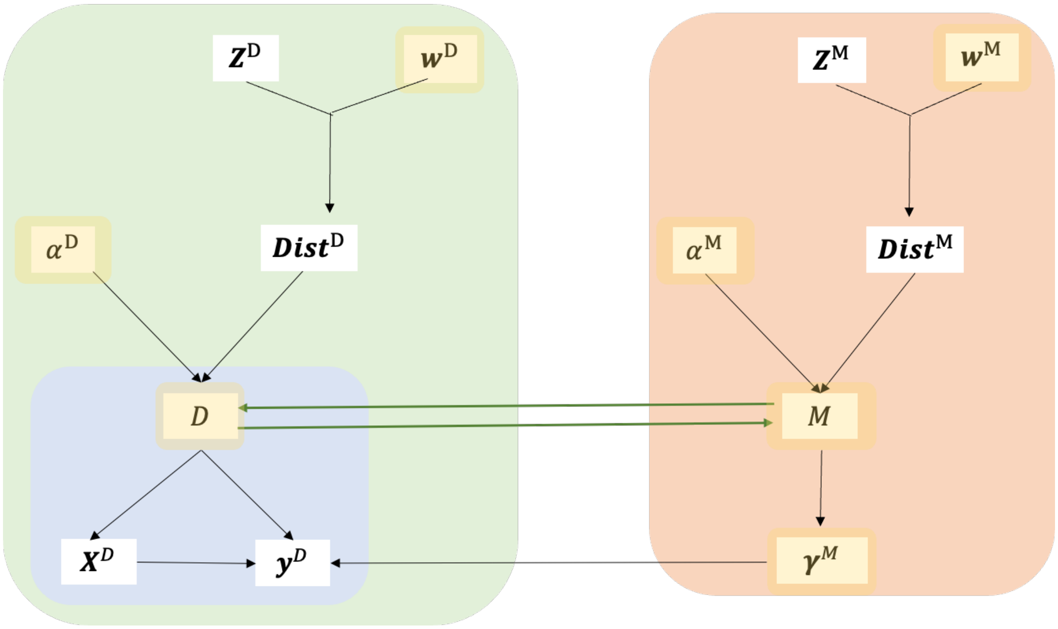
Extant clustering methods select *either* data or parameter clustering while fixing the other. Figure 8 shows that our proposed method (BDC), which selects *both*, relates to some of those extant clustering methods. Extant *parameter* clustering methods like LCA determine parameter granularity ($M$) of a demand model while fixing data at the most granular level. In this respect, BDC – when its level of data granularity is fixed – reduces to a version of LCA that imposes our extended ddCRP prior on parameter clustering (hereafter, ddCRP-LCA; in orange in Figure 8). Thus, BDC nests this version of LCA.

Some studies have applied versions of LCA with different Bayesian non-parametric priors on parameter clustering, and our method relates to a few of them. A notable example (as mentioned earlier) is the location-scale partition (LSP) prior proposed by Smith, Rossi, and Allenby (2019). While they imposed the LSP prior on SKU clustering to constrain cross-price elasticities at a chosen cluster level, it could be applied to constrain demand parameters (e.g., intercept, price elasticity, advertising elasticity). The LSP distribution is governed by two elements: a prior clustering and a dispersion parameter. As implemented in Smith, Rossi, and Allenby (2019), it is centered around a single prior clustering with low dispersion, thus assuming a strong prior belief toward this clustering. In contrast, our extended ddCRP prior specifies a prior belief on clustering via distances between nodes. It models a pairwise distance as a function of a set of pairwise differences and a set of their associated prior beliefs as in equation (9). While the extended ddCRP looks different from the LSP prior on surface, it in fact extends the LSP prior. Specifically, the extended ddCRP prior allows researchers to relate clustering to a large set of pairwise differences and to specify their belief toward each difference, whereas the LSP prior focuses on a *single* difference, which is based on the prior clustering of interest, and typically assumes a *strong* belief toward this difference. (This statement is mathematically demonstrated in Online Appendix C). Therefore, BDC relates to a version of LCA that imposes the LSP prior on parameter clustering (hereafter, LSP-LCA).

In contrast, most existing *data* clustering methods are unsupervised and cluster data without considering a final demand model. One exception (as mentioned earlier) is the scaled power likelihood with multiple weights (SPLM), a model-based data clustering method proposed by Kim, Bradlow, and Iyengar (2022). This method selects data granularity ($D$) of a demand model while fixing its parameter granularity at the coarsest level. This objective is what BDC – when its level of parameter granularity is held fixed – would attain (hereafter, ddCRP-data; in green in Figure 8).

However, unlike ddCRP-data (and thus BDC), SPLM requires researchers to *pre-specify* a set of candidate data granularities (e.g., weekly, monthly, quarterly) and then selects the one with the highest in-sample posterior probability. This approach works well when candidate granularities are clear and limited. However, this is not the case in many applications – such as aggregating SKUs or customers. Furthermore, if one were to consider all possible data granularities as candidates, SPLM could become computationally infeasible. For example, clustering just 10 SKUs results in 115,975 possible clusterings, with the number increasing exponentially as the number of SKUs grows (Bell 1934). In contrast, ddCRP-data (and thus BDC) does not require a pre-defined set of granularities. Instead, it probabilistically determines the level of data granularity using attribute-based distances (as explained in Section 2.3). Since our simulations and applications focus on such settings where candidate granularities are not obvious, we use ddCRP-data, rather than SPLM, as a benchmark throughout the remainder of the paper.

**Figure 8:** Directed Graph for the Proposed Framework Versus the Related Methods



*Notes.* Our proposed framework (BDC) relates to ddCRP-LCA (in orange), ddCRP-data (in green), and SPLM (in blue). We represent in yellow the parameters sampled in our proposed method – e.g., *D* (data clustering) and *M* (parameter clustering). We explained the role of each parameter in Section 2.3.

## 3. Simulation Study

We conducted an extensive simulation study to assess the performance of our proposed method (BDC) along four dimensions: (1) recovery of the simulated pairs of data and parameter granularities; (2) the reduction in the parameter bias in contrast to extant methods (ddCRP-LCA, LSP-LCA, and ddCRP-data). As ddCRP-LCA and LSP-LCA fix data at the most granular level, the second objective highlights that it is not always beneficial to use the most granular data; (3) the generalizability (e.g., whether BDC recovers data and parameter granularities even when they are generated from the extant methods); and (4) the computational speed (e.g., the time required for BDC to converge) and computational scalability (e.g., how well BDC scales with an increasing amount of data).

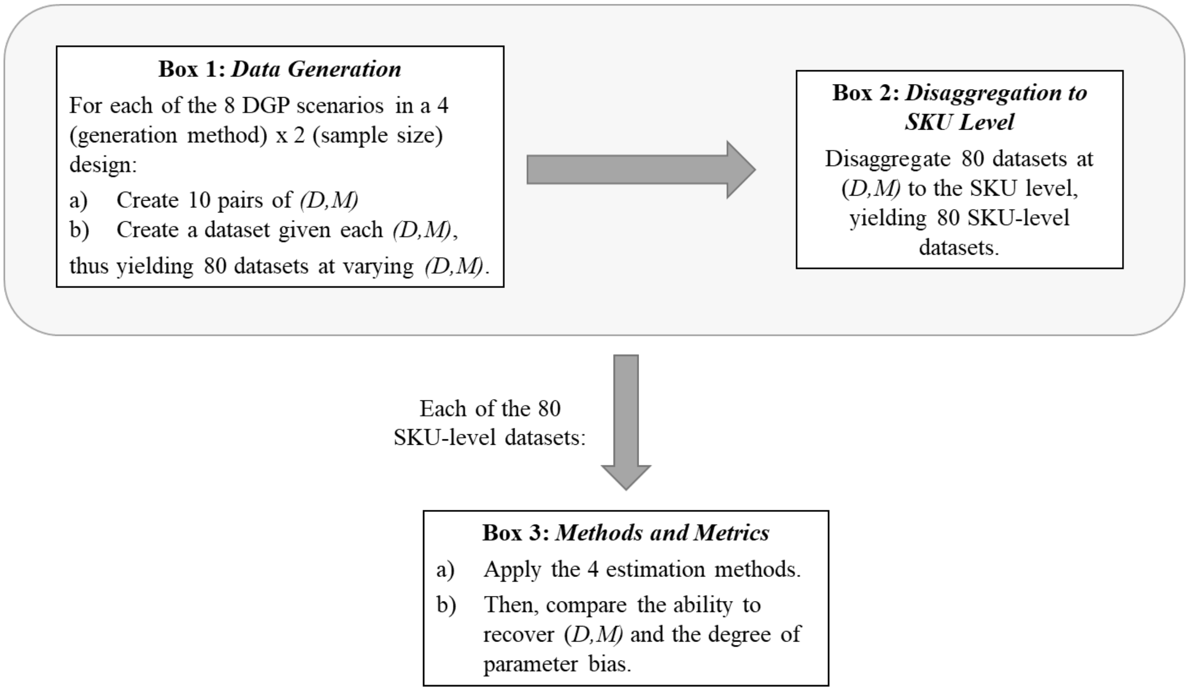
For ease of exposition, we continue to consider the context of a researcher who has access to SKU-level sales data for a CPG category (e.g., the juice and drinks category) and wishes to assess the relationship between sales and prices.

### 3.1. Simulation Design

Figure 9 overviews the design that we pursue to support our objectives of the simulation. As in Box 1, we generate sales data while varying two factors – true granularity generation method and sample size – to support the third and fourth objectives, respectively. We vary the first factor at 4 levels: our proposed method (BDC) and our comparison benchmarks (ddCRP-LCA, LSP-LCA, and ddCRP-data). We vary the latter at 2 levels: small (100 SKUs) and large (1,000 SKUs) to assess the viability of BDC for large collections. This manipulation results in 8 DGP scenarios in a 4 (true granularity generation method) x 2 (sample size) design. In each of the 8 DGP scenarios, we simulate 10 pairs of data and parameter granularities from the corresponding generation method and then generate a dataset from a demand model conditional on each simulated $(D, M)$, thus yielding a total of 80 datasets at varying $(D, M)$. We explain the granularity generation methods as well as the demand model in the *Data Generation* section.

Then, as in Box 2, we create the most granular (SKU-level) data via disaggregating each of the 80 datasets generated at $(D, M)$. We explain the disaggregation process in the *Disaggregation* section. Finally, as in Box 3a, for each of the 80 most granular datasets, we apply the same four estimation methods (BDC, ddCRP-LCA, LSP-LCA, and ddCRP-data). As in Box 3b, we then compare results across the estimation methods to demonstrate the BDC's superior ability to recover the simulated data and parameter granularity pairs and to reduce bias in parameter inferences (price elasticity in this case), supporting the first and second objectives. We explain the extensive set of clustering recovery metrics used to assess the performance in recovering the simulated granularity pair in the *Metrics* section.

**Figure 9:** Our 4 (generation method) x 2 (sample size) x 4 (estimation method) Design



**Box 1:** ***Data Generation***

For each of the 8 DGP scenarios in a 4 (generation method) x 2 (sample size) design:

a) Create 10 pairs of *(D,M)*

b) Create a dataset given each *(D,M)*, thus yielding 80 datasets at varying *(D,M)*.

**Box 2:** ***Disaggregation to SKU Level***

Disaggregate 80 datasets at *(D,M)* to the SKU level, yielding 80 SKU-level datasets.

Each of the 80 SKU-level datasets:

**Box 3:** ***Methods and Metrics***

a) Apply the 4 estimation methods.

b) Then, compare the ability to recover *(D,M)* and the degree of parameter bias.

***Data Generation* (Box 1, Figure 9).** Each of the 8 DGP scenarios consists of two parts. In the first part (Box 1a), we generate $(D, M)$ from each true granularity generation method. We explain each generation method in Online Appendix D. In the second part (Box 1b), we generate data containing price and sales at each simulated $(D, M)$ as described below:

(1) *Price*. We generate SKU-level prices and then aggregate (if necessary) the SKU-level prices to the data granularity $D$. Formally, for each SKU $i$ in data cluster $d$, we draw the price from $\mathrm{UNIF}(5,15)$ and denote it by $Price_{i(d)}^{D}$ with the superscript $D$ indicating the simulated data granularity.[5] We then aggregate the SKU-level prices by (without loss of generality) taking the average of the prices for SKUs in the same data cluster, and denote it by $Price_{d}^{D}$.

(2) *Sales*. For each data cluster $d$, we generate sales (denoted by $y_{d}^{D}$) from a demand model in Equation (13). As we have generated $Price_{d}^{D}$, we generate $y_{d}^{D}$ by drawing the parameters – the intercept ($\gamma_{0,m}^{M}$)

[5] We draw the SKU-level price from $UNIF(5,15)$ to keep the generated prices similar to prices observed in our application (Section 4).

and price elasticity ($\gamma_{1,m}^{M}$) – with the superscript $M$ indicating the simulated parameter granularity and plugging them into Equation (13).[6]

$$\begin{aligned} \log\left(y_d^D\right) &= \mu_d^D + \varepsilon_d^D \\ &= \gamma_{0,m}^M + \gamma_{1,m}^M \cdot log\left(Price_d^D\right) + \varepsilon_d^D \end{aligned} \tag{13}$$

where $\varepsilon_d^D$ is normally distributed with mean 0 and variance $V\left[\varepsilon_d^D\right]$. To allow enough signal in the generating process, we set the signal-to-noise ratio (SNR) = $\frac{V[\mu_d^D]}{V[\varepsilon_d^D]}$ high (i.e., at 9).[7]

***Disaggregation*** **(Box 2, Figure 9).** The most granular (SKU-level) data is the input to each estimation method we will apply, and we generate it by disaggregating the data we have generated at $D$. Among many possible disaggregation processes, we select a process that adds random noise without inducing any bias to the most granular data. Specifically, our disaggregation process adds stochastic noise to the SKU-level data, while retaining the systematic relationship between sales and price at the aggregated level. We describe our disaggregation process formally in Equations (14)–(17).

For each data cluster $d$, we disaggregate sales ($y_d^D$) to the SKU level and denote sales for SKU $i$ in cluster $d$ by $y_{i(d)}^D$:

$$y_{i(d)}^D = y_d^D \cdot share_{i(d)} \tag{14}$$

where for each cluster $d$, $\sum_i share_{i(d)} = 1$, and the vector of $share_{i(d)}$ follows a Dirichlet distribution with a vector of concentration parameters (each denoted by $r_{i(d)}$).

If the disaggregation process retains the systematic (nonlinear) relationship between sales and price in Equation (13), the log-transformed SKU-level sales must be in expectation proportional to the log-transformed price with price elasticity $\gamma_{1,m}^M$:

[6] For each parameter cluster $m$, we draw the intercept ($\gamma_{0,m}^M$) from $UNIF(4,8)$ and the price effect ($\gamma_{1,m}^M$) from $UNIF(-3,-1)$ following the related literature on price elasticities (e.g., Bijmolt, Van Heerde, and Pieters 2005; Christen et al. 1997) and the results from our application.

[7] We also generate data under low SNR (i.e., at 1/9) and assess how well each estimation method recovers the true granularities in such setting. We find that the proposed method (BDC) as well as the alternatives do not recover the true granularities well, due to lack of signal in the data generating process.

$$E\left[\log\left(y_{i(d)}^{D}\right)\right] \propto \gamma_{1,m}^{M} \cdot log\left(Price_{i(d)}^{D}\right) \tag{15}$$

The disaggregation process can maintain this relationship by setting each concentration parameter proportional to the exponential of the right-hand side of Equation (15):

$$r_{i(d)} \propto {Price_{i(d)}^{D}}^{\gamma_{1,m}^{M}} \tag{16}$$

We add moderate random noise to the disaggregation process (and so the SKU-level data) by setting the sum of the concentration parameters small (here, 10). Thus, for each data cluster $d$, we disaggregate its sales to the SKU level, in proportion to a vector of shares drawn from the Dirichlet distribution with each concentration parameter $r_{i(d)}$ being:

$$r_{i(d)} = \frac{{Price_{i(d)}^{D}}^{\gamma_{1,m}^{M}}}{\sum_i \ {Price_{i(d)}^{D}}^{\gamma_{1,m}^{M}}} \tag{17}$$

***Methods*** **(Box 3a, Figure 9).** We apply the four aforementioned estimation methods to the most granular (SKU-level) data. Unlike the proposed method (BDC), the alternatives (ddCRP-LCA, LSP-LCA, and ddCRP-data) as mentioned pre-specify either data or parameter granularity and infer the other in the log-log demand model in Equation (13).

For the ddCRP-based methods (BDC, ddCRP-LCA, ddCRP-data), we impose weakly informative priors on their parameters: $Normal(0,10)$ on the logit of the self-link probability parameters ($\alpha^{\mathrm{D}}, \alpha^{\mathrm{M}}$), $Normal(0,10)$ on the log of the weight parameters ($w^{\mathrm{D}}, w^{\mathrm{M}}$), $Normal(0, 10^2)$ on the demand model parameters ($\gamma_{0,m}^{M}, \gamma_{1,m}^{M}$), and $InvGamma(1,1)$ on the error variance ($V\left[\varepsilon_d^D\right]$).

For the LSP-LCA, the LSP prior is centered around a prior clustering ($\rho$) with dispersion ($\tau$). We set $\rho$ at the brand level (i.e., SKUs with the same brand have the same parameters) and $\tau$ low – e.g., $\tau = \frac{1}{N \cdot \log(N)}$ where $N$ is the number of SKUs – following Smith, Rossi, and Allenby (2019). Like the ddCRP-based methods, we impose $Normal(0, 10^2)$ on the model parameters and $InvGamma(1,1)$ on the error variance.

***Metrics* (Box 3b, Figure 9).** We use four clustering evaluation metrics – Rand Index (RI), Recall-Positive (RP), Recall-Negative (RN), and Normalized Mutual Information (NMI) – to assess the performance of each method in recovering the true underlying data and parameter clusters (e.g., Kvalseth 1987; Rand 1971; Vinh, Epps, and Bailey 2009). The first three metrics assess the recovery of pairwise clustering while the last one is based on mutual information. All four metrics range from 0 (perfect disagreement) to 1 (perfect agreement). Online Appendix E contains more details on these metrics, but we note that it is important to consider all of them as they reflect true positives, true negatives, and a combination thereof.

### 3.2. Simulation Results

We conduct inferences with the four previously described methods, using MCMC sampling implemented in R on a Mac Studio with an Apple M2 Ultra CPU with 24 cores and 192 GB of RAM. For the ddCRP-based methods, we run it for 200 iterations and discard the first 25% of draws as burn-in.[8] For LSP-LCA, we run each Markov chain for 50,000 iterations and discard the first 50% of draws as burn-in (e.g., Smith, Rossi, and Allenby 2019).[9] We use the post-burn-in samples to compare the performance of our method with that of the alternatives particularly along three dimensions: (1) the recovery of the simulated granularity pair, (2) the recovery of the price elasticity, and (3) the computational speed and scalability.

#### (1) Recovery of the simulated granularity pair

Table 1 reports the posterior means of the granularity recovery metrics (RI, RP, RN, and NMI) across the generation and estimation methods for both small and large sample sizes. We report the posterior distributions – specifically, 95% credible intervals – of these recovery metrics are provided in Online

[8] As discussed in Section 2.4, rapid mixing and convergence is one notable feature of the ddCRP sampler. For each parameter, we use a Gaussian random walk proposal distribution with its mean centered on the value from the previous iteration. The variance of each proposal distribution is adjusted to attain an acceptance rate close to 50% (Gelman et al. 2013).

[9] We use the LSP random walk proposal distribution with its mean centered on the parameter clustering from the previous iteration. Similar to the ddCRP-based methods, we adjust the step size of each proposal distribution to attain an acceptance rate close to 50% (Gelman et al. 2013).

Appendix F. We discuss key findings based on the small sample size only (Table 1[A]), as these findings generally hold for the large sample size (Table 1[B]).

Table 1[A-1] compares the posterior means of the *data* granularity recovery metrics across the four aforementioned true DGPs (BDC, ddCRP-LCA, LSP-LCA, ddCRP-data) and the two estimation methods (BDC, ddCRP-data). We focus on these two estimation methods (BDC and ddCRP-data), which infer *data* granularity rather than fixing it. Table 1[A-1] provides two key findings. First, when BDC is used for estimation, it achieves consistently high recovery metrics across all true DGPs, highlighting BDC's ability to recover the true data granularity. In contrast, ddCRP-data performs well only when the true DGP is also ddCRP-data. Its recovery metrics drop considerably when the true DGP is BDC, ddCRP-LCA, or LSP-LCA. On further inspection, we find that ddCRP-data tends to choose overly coarse data as it *fixes* parameters at the coarsest level and infers data granularity conditional on the coarsest parameters.

Similarly, Table 1[A-2] compares the posterior means of the *parameter* granularity recovery metrics across the four generation methods and the three estimation methods that estimate parameter granularity (BDC, ddCRP-LCA, and LSP-LCA). The table provides two key additional findings. First, BDC provides consistently high recovery metrics regardless of the true DGP, highlighting BDC's ability to recover the true parameter granularity. In contrast, the LCA methods perform well only when the true DGP matches the corresponding LCA method. Their RP values drop substantially when the true DGP is BDC or ddCRP-data. We find that the LCA methods choose overly granular parameters, as they *erroneously* fix data at the most granular level.

**Table 1**: Posterior Means of the Data and Parameter Clustering Recovery Metrics (RI, RP, RN, NMI) Across the Generation and Estimation Methods

[A] Small Sample Size

[A-1] Data Clustering

| | | Estimation methods | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | BDC | | | | ddCRP-data | | | |
| | | RI | RP | RN | NMI | RI | RP | RN | NMI |
| True DGPs | BDC | 1.00 | 0.91 | 1.00 | 0.98 | 0.11 | 0.92 | 0.09 | 0.02 |
| | ddCRP-LCA | 0.99 | --- | 0.99 | 0.91 | 0.25 | --- | 0.25 | 0.20 |
| | LSP-LCA | 0.99 | --- | 0.99 | 0.90 | 0.38 | --- | 0.38 | 0.31 |
| | ddCRP-data | 0.98 | 0.92 | 1.00 | 0.99 | 1.00 | 0.93 | 1.00 | 0.99 |

[A-2] Parameter Clustering

| | | Estimation methods | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | BDC | | | | ddCRP-LCA | | | | LSP-LCA | | | |
| | | RI | RP | RN | NMI | RI | RP | RN | NMI | RI | RP | RN | NMI |
| True DGPs | BDC | 1.00 | 1.00 | 1.00 | 1.00 | 0.82 | 0.24 | 1.00 | 0.62 | 0.83 | 0.06 | 1.00 | 0.59 |
| | ddCRP-LCA | 0.99 | 0.98 | 1.00 | 0.97 | 0.99 | 0.97 | 1.00 | 0.98 | 0.86 | 0.88 | 0.86 | 0.78 |
| | LSP-LCA | 1.00 | 1.00 | 1.00 | 1.00 | 0.96 | 0.84 | 1.00 | 0.91 | 0.83 | 0.86 | 0.83 | 0.81 |
| | ddCRP-data | 1.00 | 1.00 | --- | --- | 0.02 | 0.02 | --- | --- | 0.01 | 0.01 | --- | --- |

[B] Large Sample Size

[B-1] Data Clustering

| | | Estimation methods | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | BDC | | | | ddCRP-data | | | |
| | | RI | RP | RN | NMI | RI | RP | RN | NMI |
| True DGPs | BDC | 0.99 | 0.77 | 1.00 | 0.91 | 0.58 | 0.45 | 0.59 | 0.08 |
| | ddCRP-LCA | 0.98 | --- | 0.98 | 0.80 | 0.39 | --- | 0.39 | 0.27 |
| | LSP-LCA | 0.97 | --- | 0.97 | 0.75 | 0.57 | --- | 0.57 | 0.41 |
| | ddCRP-data | 0.99 | 0.77 | 0.99 | 0.91 | 0.99 | 0.80 | 0.99 | 0.91 |

[B-2] Parameter Clustering

| | | Estimation methods | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | BDC | | | | ddCRP-LCA | | | | LSP-LCA | | | |
| | | RI | RP | RN | NMI | RI | RP | RN | NMI | RI | RP | RN | NMI |
| True DGPs | BDC | 0.98 | 0.91 | 1.00 | 0.95 | 0.75 | 0.12 | 1.00 | 0.24 | 0.75 | 0.01 | 1.00 | 0.37 |
| | ddCRP-LCA | 0.97 | 0.93 | 0.99 | 0.91 | 0.96 | 0.96 | 0.97 | 0.88 | 0.99 | 0.97 | 1.00 | 0.87 |
| | LSP-LCA | 1.00 | 1.00 | 1.00 | 1.00 | 0.86 | 0.87 | 0.88 | 0.83 | 0.99 | 0.96 | 1.00 | 0.92 |
| | ddCRP-data | 1.00 | 1.00 | --- | --- | 0.01 | 0.01 | --- | --- | 0.01 | 0.01 | --- | --- |

*Notes*.

1) In panels [A-1] and [B-1], RP cannot be computed when ddCRP-LCA or LSP-LCA is the true generating process. It is because the denominator of RP is proportional to the number of SKU pairs that are clustered together in the true data clustering (*D*), and this number is 0 under the LCA methods that fix the true *D* at the most granular level.

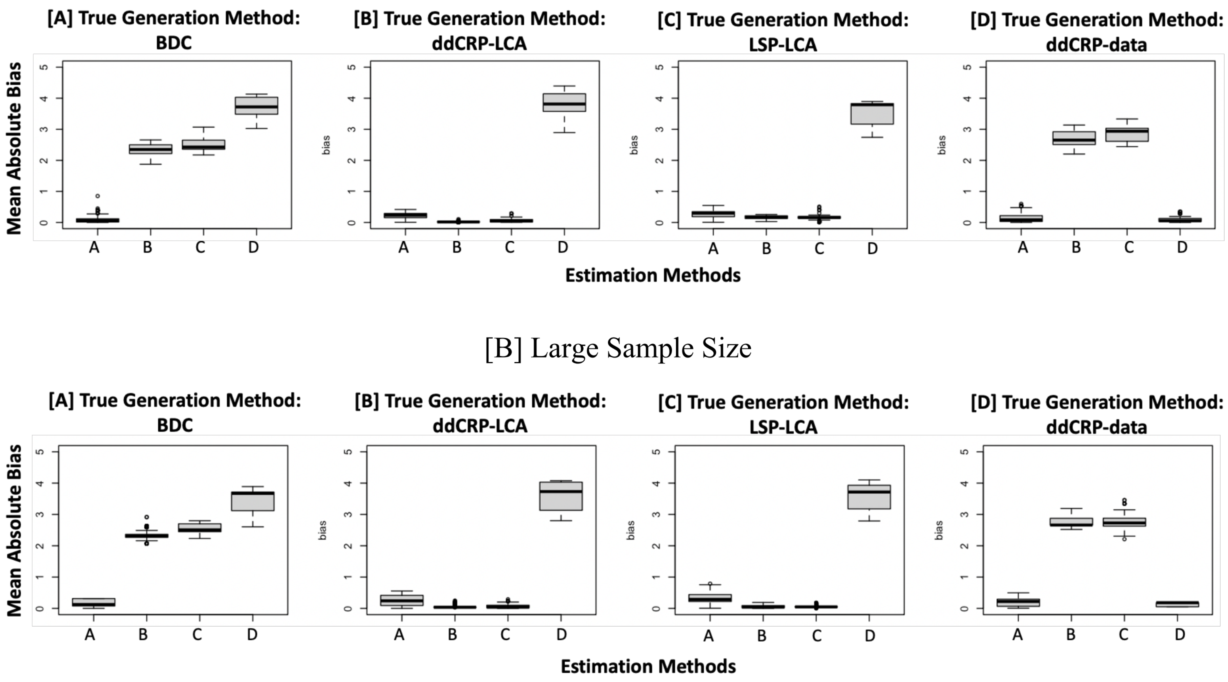
2) In panels [A-2] and [B-2], RN and NMI cannot be computed when ddCRP-data is the true generating process. It is because their denominators are proportional to the number of SKU pairs that are not clustered together in the true parameter clustering (*M*), and this number is 0 under ddCRP-data that fixes the true *M* at the coarsest level.

### (2) Bias in the price elasticity

Figure 10 compares the posterior distributions of the mean absolute bias in the price elasticity across the four DGPs and the same four estimation methods, under both the small and large sample sizes. It provides two key findings for both the sample sizes. First, the bias is negligible when the generation and estimation methods match, supporting the validity of our simulation study. Next, notably, the bias is small for the proposed method (BDC) regardless of the true DGP. It is because our method recovers the true data and parameter granularities well regardless of their generating processes (as shown in Table 1).

**Figure 10:** Posterior Distributions of Mean Absolute Bias in Price Elasticities Across the Generation and Estimation Methods

[A] Small Sample Size



[B] Large Sample Size

*Note*. The estimation methods (A, B, C, and D) are BDC, ddCRP-LCA, LSP-LCA, and ddCRP-data, respectively.

### (3) Computational speed and scalability

We report how computational time for our method scales with the number of SKUs in this setting. Specifically, for the small sample size (100 SKUs), our method converges within 20 iterations (0.60 minutes) on average. For the large sample size (1,000 SKUs), it again converges within 20 iterations (12.28 minutes) on average. Even for larger samples – 3,000 and 10,000 SKUs – the number of iterations remains similar: 20 iterations (50.42 minutes) for 3,000 SKUs and 25 iterations (6.94 hours) for 10,000 SKUs. Detailed convergence diagnostics – including trace plots, Gelman-Rubin statistics, and autocorrelated functions (ACFs) – are provided in Online Appendix G.

## 4. Application

### 4.1. Data

Our data come from the Nielsen database at the Kilts Center at the University of Chicago. The original data contains unit sales, price, the incidence of in-store feature advertising, and the incidence of in-store display advertising information at the SKU-store-week level. To illustrate the benefits of our proposed method, we use data from the Juice and Drinks category. SKUs in this category can be categorized based on multiple features (e.g., Nielsen-defined product module, brand, package size, and package material), and hence it is well suited to assess our proposed method. The category consists of 17 product modules defined by Nielsen. We use data from the top eight product modules, which constitute 93% of total unit sales for the entire Juice and Drinks category.

To remove other confounding effects, we use data from 19 stores within a single grocery chain in Chicago, Illinois, for which in-store feature and display advertising information was available. We also use data for four years from January 2016 to December 2019. We use the first three years of data for in-sample estimation and the last one year of data for out-of-sample prediction.

Nielsen tracks price and advertising information for SKUs only when they were purchased. To avoid potential missing data issues, we retain SKU-store combinations that were purchased at least once in every quarter.[10] The final SKU-store-quarterly data contain 332 SKUs and 61,040 observations. While conceptually our BDGC approach can handle SKU, store, and time aggregation simultaneously, we focus here on SKU aggregation.

Table 2 summarizes the distributions of the variables in the final in-sample SKU-store-quarterly data.[11] Figure 11 shows that prices differ across brands and package sizes but not so much across package materials. Given this observed variation, some studies on the juice and drinks category allow price elasticity to vary across brands and package sizes (e.g., 11.5 oz versus 128 oz) but not across other features (e.g., Wedel and Zhang 2004). Other studies also noted that they aggregated data to the brand-size level for the same reason (e.g., Dubé and Gupta 2008; Hoch et al. 1995). In Section 4.3, we assess the performance of common practice for researchers (e.g., aggregating data and parameters to the brand-size level) and practitioners (e.g., aggregating data to the Nielsen-defined product module level).

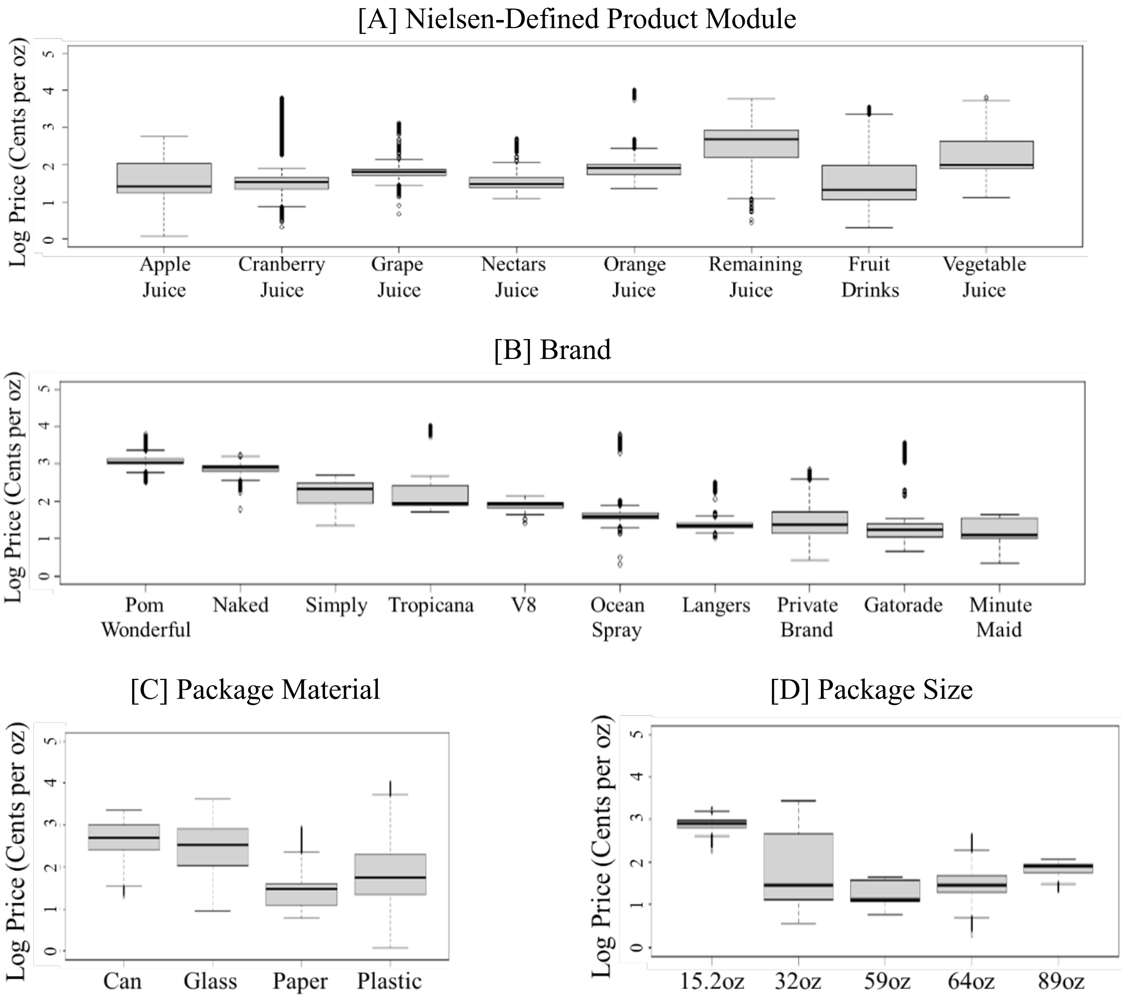
**Table 2:** Descriptive Statistics of the Final SKU-Store-Quarterly Juice and Drinks Data

| | Mean | SD | Min | Max |
|---|---|---|---|---|
| Unit sales | 95.18 | 178.07 | 1 | 7398 |
| Price (cents per oz) | 8.93 | 7.34 | 1.07 | 55 |
| Number of display advertisements | 0.04 | 0.12 | 0 | 1 |
| Number of feature advertisements | 0.13 | 0.17 | 0 | 1 |

[10] We set time granularity at the quarterly level to include a large number of SKUs in the analysis. Smaller numbers of SKUs (96 and 266 SKUs) were retained if we set time granularity at more granular levels (weekly and monthly levels), respectively.

[11] We aggregate the SKU-store-weekly data provided by Nielsen to the SKU-store-quarter level. Specifically, for each SKU-store combination, we sum up unit sales and take the sales-weighted averages of price, display advertising, and feature advertising across all weeks in each quarter.

**Figure 11:** Pricing Patterns of SKUs in the Juice and Drinks Category across SKU Features



[A] Nielsen-Defined Product Module

[B] Brand

[C] Package Material

[D] Package Size

*Notes.* In panel [B], we draw boxplots only with 10 brands (among 53 brands) with the highest total unit sales in the final in-sample data. In panel [D], we draw boxplots only with five sizes (among 35 size types) with the highest total unit sales in the final in-sample data.

### 4.2. Analysis

**4.2.1. Data and parameter clustering methods.** We choose data and/or parameter clusterings using the same four methods used in the simulation:

(1) **Proposed method (BDC).** BDC chooses both data and parameter clusterings. To understand (and interpret) the effects of the four observed SKU features on the choice of data/parameter clustering, we set the distance between SKUs as a function of the weighted average of four feature-based variables.

The first three features (module, brand, and package material) are categorical, whereas the last one (package size) is numerical. Hence, we define the first three feature-based variables as indicators – 1) whether their product modules differ, 2) whether their brands differ, and 3) whether their package materials differ – whereas the last one as a standardized difference in size between SKUs. Specifically, we compute the last variable by taking an absolute difference between their sizes and then standardizing the absolute difference by its maximum value. This way, we can ensure that the variable ranges between 0 and 1 like the other feature-based variables.

**(2) ddCRP-data.** It fixes parameters at the coarsest level and infers data clustering conditional on the coarsest parameters. Thus, we expect that data clustering chosen by this method would differ from that chosen by BDC. Specifically, we expect that data clustering chosen by this method would be less granular than data clustering chosen by BDC, as previously discussed.

**(3) ddCRP-LCA.** It fixes data at the most granular (here, SKU-store-quarterly) level and infers parameter clustering conditional on the most granular data. Thus, we expect that parameter clustering chosen by this method would differ from that chosen by BDC. Specifically, we expect that parameter clustering chosen by this method would be more granular than parameter clustering chosen by BDC, as previously discussed.

**(4) LSP-LCA**. Like ddCRP-LCA, it infers parameter clustering while fixing data at the most granular level. However, it differs from ddCRP-LCA since it imposes the LSP prior instead of the ddCRP prior on the parameter clustering. The LSP prior is centered around a prior clustering ($\rho$) with dispersion ($\tau$). We follow the extant literature (e.g., Wedel and Zhang 2004) and set $\rho$ at the brand-size level (i.e., SKUs with the same brand and the same size have the same parameters). We set $\tau$ low – e.g., $\tau = \frac{1}{N \cdot \log(N)}$ where $N$ is the number of SKUs – following Smith, Rossi, and Allenby (2019).

**4.2.2. Data and parameter aggregation.** We apply a demand model conditional on the chosen data and parameter clusterings. First, we aggregate the most granular data based on $D$. We denote the aggregated variables as $\text{sales}_{dst}^{D}$, $\text{price}_{dst}^{D}$, $\text{display}_{dst}^{D}$, and $\text{feature}_{dst}^{D}$ with the superscript $D$ indicating the sampled data clustering. Specifically, $\text{sales}_{dst}^{D}$ is the total number of SKUs in data cluster $d$ that were sold at store $s$ in quarter $t$. We use the total unit sales as a dependent variable to estimate price elasticity (i.e., the impact of price change on quantity demand) (e.g., Hoch et al. 1995; Smith, Rossi, and Allenby 2019). The variables $\text{price}_{dst}^{D}$, $\text{display}_{dst}^{D}$, and $\text{feature}_{dst}^{D}$ refer to the simple-averaged price, display advertisement, and feature advertisement, respectively, averaged across SKUs in data cluster $d$ at store $s$ in quarter $t$.[12] Then, we apply the following log-log model to the aggregated data.

$$\log\left(\text{sales}_{dst}^{D}\right) = \gamma_{0,m}^{M} + \gamma_{1,m}^{M} \log\left(\text{price}_{dst}^{D}\right) + \gamma_{2,m}^{M} \log\left(\text{display}_{dst}^{D} + 1\right) + \gamma_{3,m}^{M} \log\left(\text{feature}_{dst}^{D} + 1\right) + \varepsilon_{dst}^{D} \tag{18}$$

In the model, we aggregate model parameters – intercept, price, feature advertising, and display advertising elasticities – based on $M$. We denote the aggregated parameters as $\gamma_{0,m}^{M}$, $\gamma_{1,m}^{M}$, $\gamma_{2,m}^{M}$, and $\gamma_{3,m}^{M}$, respectively, with the superscript $M$ indicating the sampled parameter clustering.

### 4.3. Results

We use MCMC sampling to fit our method (BDC) and the alternatives. For the ddCRP-based methods (BDC, ddCRP-data, and ddCRP-LCA), we run three parallel chains with different starting values for 500 iterations each. For LSP-LCA, we run three parallel chains with different starting values for 50,000 iterations each. For each chain, we discard the first half as burn-in and use the last half for analysis. We compute the Gelman-Rubin statistic (Gelman and Rubin 1992) on the post-burn-in iterations for each parameter. The statistic is always less than 1.2, suggesting that the model convergence is satisfactory.

**4.3.1. Model fit.** We use the log of the in-sample Bayes factor (BF) and the in-sample weighted average percentage error (WAPE), which are comparable across granularities (e.g., Kim, Bradlow, and Iyengar

[12] We do not take sales-weighted averages, as they may induce endogeneity issues. However, concerns regarding endogeneity and model misspecification may remain, as the demand model in equation (18) does not account for competitors' prices and advertising. We acknowledge this as a limitation in the conclusion section.

2022), to compare the in-sample fit of our method with those of the extant methods. First, we denote the log of the in-sample Bayes factor of our method in comparison to the extant methods (ddCRP-data, ddCRP-LCA, and LSP-LCA) as $\log \mathrm{BF}_{\mathrm{in,data}}$, $\log \mathrm{BF}_{\mathrm{in,ddCRP}}$, and $\log \mathrm{BF}_{\mathrm{in,LSP}}$, respectively. For example, $\log \mathrm{BF}_{\mathrm{in,data}}$ denotes the difference between the log of the in-sample *scaled* marginal likelihood of BDGC ($\mathrm{SML}_{\mathrm{in,BDGC}}$) and that of the ddCRP-data ($\mathrm{SML}_{\mathrm{in,data}}$):

$$\log \mathrm{BF}_{\mathrm{in,data}} = \log \mathrm{SML}_{\mathrm{in,BDC}} - \log \mathrm{SML}_{\mathrm{in,data}} \tag{19}$$

We estimate the scaled marginal likelihood by taking the harmonic mean of the scaled likelihood in post-burn-in MCMC iterations. For instance, we estimate $\mathrm{SML}_{\mathrm{in,BDC}}$ by taking the harmonic mean of the scaled likelihood ($\mathrm{SL}_{\mathrm{in,BDC}}$). If there are $S$ post-burn-in MCMC iterations, $\mathrm{SML}_{\mathrm{in,BDGC}}$ is denoted as:

$$\mathrm{SML}_{\mathrm{in,BDC}} = \frac{S}{\sum_{s=1}^{S} \frac{1}{\mathrm{SL}_{\mathrm{in,BDC,s}}}} \tag{20}$$

We measure $\log \mathrm{BF}_{\mathrm{in,ddCRP}}$ and $\log \mathrm{BF}_{\mathrm{in,LSP}}$ in the similar way. Note that the log of Bayes factor greater than 5 is considered as strong evidence for our method over extant methods (Kass and Raftery 1995). We find that $\log \mathrm{BF}_{\mathrm{in,data}} = 25.52$, $\log \mathrm{BF}_{\mathrm{in,ddCRP}} = 43.71$, and $\log \mathrm{BF}_{\mathrm{in,LSP}} = 44.24$, thus indicating that our method outperforms the alternatives in sample. We also find that the posterior mean of the in-sample WAPE (in estimating unit sales) is lower under our method than that under the extant methods. Specifically, $\mathrm{WAPE}_{\mathrm{in,BDC}} = 27\%$ is lower than $\mathrm{WAPE}_{\mathrm{in,data}} = 34\%$, $\mathrm{WAPE}_{\mathrm{in,ddCRP}} = 36\%$, and $\mathrm{WAPE}_{\mathrm{in,LSP}} = 36\%$.

While the primary motivation for our approach is model selection using in-sample fit, researchers may wish to use out-of-sample prediction as their primary goal. To this end, we also assess the out-of-sample predictive validity. We use the log of the out-of-sample Bayes factor ($\log \mathrm{BF}_{\mathrm{out}}$) to assess whether the improvement in marginal likelihood observed in-sample holds out-of-sample as well. We find that $\log \mathrm{BF}_{\mathrm{out,data}} = 25.38$, $\log \mathrm{BF}_{\mathrm{out,ddCRP}} = 60.21$, and $\log \mathrm{BF}_{\mathrm{in,LSP}} = 60.66$, thus suggesting that our method performs significantly better than the alternatives out of sample as well. The posterior mean of the out-of-sample WAPE is also lower under our method than under the extant methods. Specifically,

$\text{WAPE}_{\text{out,BDGC}} = 27\%$ is lower than $\text{WAPE}_{\text{out,data}} = 35\%$, $\text{WAPE}_{\text{out,ddCRP}} = 45\%$, and $\text{WAPE}_{\text{out,LSP}} = 45\%$.[13]

Therefore, the results for both the in-sample and out-of-sample model fits demonstrate the importance of selecting the levels of $D$ and $M$ simultaneously.

**4.3.2. Inference for data and parameter clusterings.** One may consider inferring data and parameter clusterings by drawing SKU-by-SKU heatmaps, each component of which captures the proportion of the post-burn-in iterations that the corresponding SKU pair was in the same data/parameter clusters (e.g., Smith, Rossi, and Allenby 2019). For a small number of SKUs (e.g., 30 SKUs), this heatmap can provide a way of interpreting the data and parameter clusterings. However, in applications with a large number of SKUs like this one, such heatmap becomes too large to offer simple interpretation.

Alternatively, our method proposes a way of inferring data and parameter clusterings even with a large number of SKUs. Specifically, since our method estimates the effects of each SKU feature on the choice of data/parameter clustering, researchers can interpret drivers behind the choice by, as in Figure 12, inferring the corresponding weight parameters and comparing them across the features.

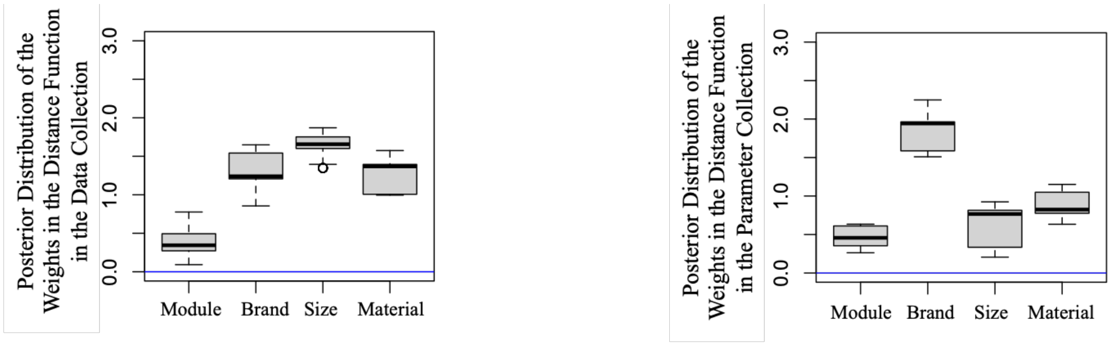
Figure 12[A] plots the posterior distributions of the 'module', 'brand', 'size', and 'material' weight parameters in data clustering. The posterior means for the 'brand', 'size', and 'material' weight parameters are not significantly different from each other, indicating that the data for SKUs within the same brand, package size, and package material are more likely to be aggregated together than SKUs within the same module. The result in Figure 12[A] suggests that while researchers and practitioners tend to assume that data variance across brands and sizes are solely important in understanding the effects of marketing actions

[13] One might suggest hierarchical Bayes as a solution to the imprecise inference. In this regard, we apply a hierarchical Bayes while fixing the data at the most granular level and assuming a brand–SKU parameter hierarchy (e.g., Voleti, Gangwar, and Kopalle 2017). We find that BDC performs significantly better than hierarchical Bayes, both in sample and out of sample. For example, the log of the in-sample and out-of-sample Bayes factor of BDC compared to hierarchical Bayes are 53.26 and 46.59, respectively – both greater than the threshold of 5.

on sales, as previously noted, it is important to also include the data variance across package materials as well.

Figure 12[B] plots the posterior distributions of the ‘module’, ‘brand’, ‘material’, and ‘size’ weight parameters in parameter clustering. The posterior mean for the ‘brand’ weight parameter is significantly greater than that for the other weight parameters, indicating that the parameters for SKUs in the same brand are more likely to be aggregated together than SKUs in the same module, size, or material. We find this result particularly interesting because it cautions against the aforementioned common practice of setting the parameter granularity at the brand-size level. For instance, while researchers tend to assume that price elasticities would differ across both brands and sizes, our result suggests that the price elasticities in this context differ across the brands and less so across the sizes.

**Figure 12: The Posterior Distribution of the Latent Weights in the Distance Functions**



**[A] Data clustering**

**[B] Parameter clustering**

Finally, it is important to note that our chosen data and parameter clusterings differ from those under the extant clustering approaches. Specifically,

**(1) ddCRP-data.** The data clustering ($D$) under our method (BDC) differs from $D$ based on ddCRP-data. Specifically, for 82% of the SKU pairs, the posterior parameter clustering probability is greater under ddCRP-data than under BDC. That is, ddCRP-data selects coarser data than BDC does, as we expected.

**(2) ddCRP-LCA.** The parameter clustering ($M$) under BDGC differs from $M$ based on ddCRP-LCA. Specifically, we find that for 99% of the SKU pairs, the posterior parameter clustering probability is greater under BDC than under ddCRP- LCA. That is, ddCRP-LCA selects more granular parameters than BDC does.

**(3) LSP-LCA.** Like ddCRP-LCA, LSP-LCA selects more granular parameters than BDC does. Specifically, 99% of the SKU pairs, the posterior parameter clustering probability is greater under BDC than under LSP-LCA.

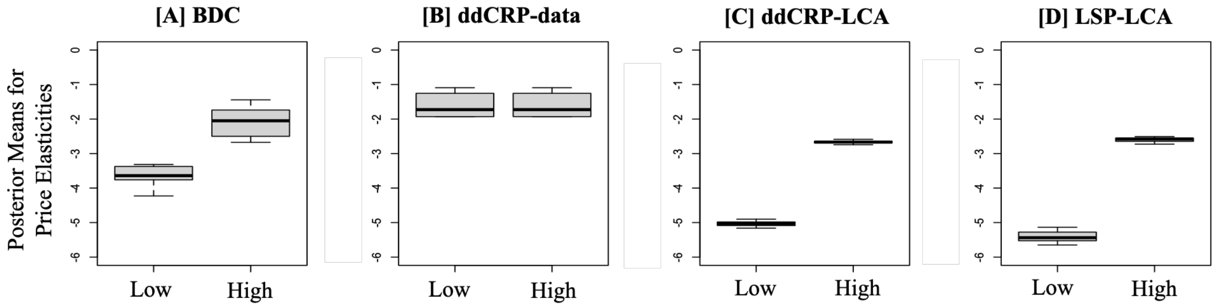
**4.3.3. Inference for price elasticities.** To further highlight our findings, we next compare the price elasticities from our proposed method (BDC) with those from the extant methods. We focus on two brands: Minute Maid (which is in the lowest price tier – i.e., 3.50 cents per oz on average) and Evolution (which is in the highest price tier – i.e., 28.38 cents per oz on average). Figure 13[A] shows that price elasticities sampled from our method are more elastic for the low-price-tier brand than for the high-price-tier brand.

Figure 13[B] provides two findings regarding the price elasticities under ddCRP-data. First, the price elasticities are the same between the two brands. It is because ddCRP-data fixes parameters at the coarsest level and assumes that the price elasticities are homogeneous across SKUs. If our method (BDC) selects the correct parameter clustering and price elasticities should differ across the two brands (as in Figure 13[A]), the result suggests that ddCRP-data can mislead managers to overlook heterogeneity in price elasticity and to set uniform prices across SKUs when they should differ. Next, the price elasticities under ddCRP-data are less elastic than those under BDC. It is because ddCRP-data, which conditions on the coarsest parameters, selects a coarser data granularity than BDC. If BDC selects the correct parameter clustering, the result suggests that ddCRP-data can mislead managers to select overly aggregated data, thus underestimating price elasticities and, in turn, setting overly high prices (e.g., Christen et al. 1997).

Figures 13[C]–[D] provides an additional finding regarding the price elasticities under ddCRP-LCA and LSP-LCA. The price elasticities under both LCA methods are more elastic than those under BDC. It is because the LCA methods condition on the most granular data, which is more granular than data granularity chosen by BDC. If BDC selects the correct data clustering, the result suggests that the LCA methods can mislead managers to select overly granular data, thus overestimating price elasticities and, in turn, setting overly low prices.

Therefore, the results in Figure 13 caution researchers against fixing either data or parameter granularity and highlight the importance of inferring both simultaneously, which is in line with our findings from the simulations.

**Figure 13: The Posterior Means for the Price Elasticities Across Brands**



## 5. Conclusions and Future Research

Researchers often use well-known model selection tools (e.g., BIC, marginal likelihood) to select the best-fitted model. While researchers pay a lot of attention to their model specification, they unknowingly make two important modeling decisions – data and parameter granularities – and select a model *conditional* on these dual decisions. While extant research has applied various heuristics to justify the decisions – e.g., using the most granular data and parameters, or following a common practice (e.g., Dubé and Gupta 2008; see Section 4) – how to do so is not straightforward.

In this research, we propose a Bayesian dual clustering method (BDC) as a novel way to select data and parameter granularities jointly. Formally, we represent each unit as a node in two collections – data and parameter collections – and cluster the nodes in the collections to select the best-fitting levels of data and parameter granularities. To do so, we propose a novel extension of the Bayesian non-parametric clustering method, the distance-dependent Chinese Restaurant Process (ddCRP), originally used to cluster texts and images. By representing data and parameters in a generalized way (particularly, in a node collection structure), our proposed method (BDC) is *flexible* and allows for many different types of units (e.g., SKU, person, time, space). A notable feature of our model is the *interpretability* of the results. By relating distances between nodes (e.g., SKUs) in the two collections to their observed attributes (e.g., brand, package size), the proposed method sheds light on why certain levels of data and parameter granularities are chosen.

We highlight the performance of our proposed method (BDC) as compared to that from the extant methods using an extensive simulation study and a large-scale real data application. In the simulation study, we show that BDC recovers the true levels of data and parameter granularities better as compared to the extant methods. It is because unlike our method, the extant methods select *either* data or parameter granularity while *conditioning* on the other. In particular, the LCA methods choose overly granular parameters because it fixes the data at the most granular level and chooses the level of parameter granularity conditional on the most granular data. We apply our proposed method as well as the extant ones to the Nielsen scanner data and confirm several findings from the simulation study. Furthermore, the inference of demand parameters (e.g., price elasticity) obtained by our method differ substantially from those by the extant methods.

Beyond its methodological contributions (discussed above), our proposed framework offers conceptual and substantive insights. Conceptually, we show that data and parameter granularities should not be treated as fixed components in demand modeling. Instead, we demonstrate that these granularities interact with model inference and can significantly affect empirical results. To our knowledge, this is the first study to treat both data and parameters as parameters to be estimated instead. Substantively, our empirical study shows that our approach can lead to granularity choices that differ from those implied by common heuristics. For example, our method suggests aggregating SKU-level data by brand, package size, and package material, which differs from the typical brand-size aggregation. This suggests that data variance across package materials may play a more important role in understanding how marketing actions affect sales than is commonly assumed. Furthermore, we find that using our method significantly improves both in-sample estimation and out-of-sample forecasting accuracy compared to heuristic-based granularity choices, highlighting its practical value.

Future research may consider several methodological extensions. First, while our method aggregates data and parameters each along a single dimension, there are contexts where researchers wish to aggregate each along multiple dimensions. This can be done by extending our dual clustering method to accommodate multiple clusterings. For instance, consider a researcher who wishes to perform a demand analysis using customer-week-level panel data. The researcher can aggregate the data and parameters across both customers and weeks by clustering four (instead of two) collections of nodes – i.e., two collections for cross-customer and cross-week data aggregation and the other two for parameter aggregation. Second, while our study assumes a parametric demand model, researchers may wish to apply nonparametric demand models where the number of parameters grows with the number of observations in the data. This can be done by setting parameter clustering to be equivalent to data clustering. Third, future applications may incorporate competitors' marketing actions (e.g., pricing and advertising) into the demand model to address potential endogeneity and model misspecification issues. Fourth, while we focus on structured attributes (e.g., brand), researchers may consider unstructured attributes (e.g., product image), which are typically high-dimensional. It would be worthwhile to investigate how these unstructured attributes could be incorporated into our framework.

Fifth, while BDC focuses on inferring granularities conditional on modeling choices made prior to inference, one might wish to make such modeling choices jointly with the inference of granularities. For instance, consider a researcher who wishes to choose $g^{\mathrm{D}}$ (the functional form of the distance function) from a set of candidates. The researcher can make such selection by running BDC separately for each candidate and then choosing the one with the highest posterior probability. In this case, the overall approach should be framed as a Bayesian optimization method rather than a sampler.

Lastly, while BDC demonstrates reasonable computational efficiency for large data sets, future research may further enhance it. Promising directions include mini-batching (e.g., De Sa, Chen, and Wong 2018; Li and Wong 2018), stochastic gradient Hamilton Monte Carlo methods (e.g., Dang et al. 2019), and doubly stochastic variational Bayes (e.g., Titsias and Lázaro-Gredilla 2014). In particular, mini-batching reduces computational cost by randomly subsampling nodes, iterating over these nodes (rather than all nodes), and sampling their links in each iteration. Recent studies in Computer Science have highlighted its computational benefits in Bayesian inference. For example, De Sa, Chen, and Wong (2018) showed that Metropolis-Hastings samplers with mini-batching converged to the true posterior and were significantly faster than those without. Li and Wong (2018) further demonstrated that samplers with mini-batching converged up to 100 times faster than those without.

Future research may also consider substantive extensions. The flexibility of BDC (as explained above) makes it applicable to a variety of contexts relevant to marketers. For instance, researchers often need to decide temporal and spatial granularities – such as whether to use data at the most granular level (e.g., daily) or aggregate it to a coarser level (e.g., weekly). This decision becomes particularly important when highly granular data (e.g., data with a per-second frequency) are readily available, as such data can be noisy and sparse. Another example is a customer (or group) level analysis where researchers often cluster customers and then aggregate data and/or parameters based on the chosen clustering.

In summary, we hope that the generality, interpretability, and empirical performance of BDC make it a valuable tool for marketing scholars who have to make critical decisions regarding data and parameter granularities across a wide variety of contexts.

## Funding and Competing Interests

The authors are grateful to have received financial support from INFORMS and American Statistical Association.

## References

Arfa R, Yusof R, Shabanzadeh P (2019) Novel trajectory clustering method based on distance dependent Chinese restaurant process. *Peer Journal of Computer Science*. 8:1-5.

Ataman MB, Van Heerde H, and Mela CF (2010) The long-term effect of marketing strategy on brand sales. *J. Marketing Res*. 47(5): 862-882.

Bell ET (1934) Exponential Polynomials. *Annals of Mathematics.* 35(2):258-277.

Bijmolt THA, Van Heerde HJ, Pieters RGM (2005) New empirical generalizations on the determinants of price elasticity. *J. Marketing Res.* 42(2):141-156.

Blei DM, Frazier PI (2011) Distance dependent Chinese restaurant processes. *J. Machine Learning Res.* 12:2461-2488.

Chen Y, Yang S (2007) Estimating disaggregate models using aggregate data through augmentation of individual choice. *J. Marketing Res.* 44(November):613-621.

Christen M, Gupta S, Porter JC, Staelin R, Wittink DR (1997) Using market-level data to understand promotion effects in a nonlinear model. *J. Marketing Res.* 34(3).

Dang KD, Quiroz M, Kohn R, Minh-Ngoc T, Villani M (2019) Hamilton Monte Carlo with energy conserving subsampling. *J. Machine Learning Res.* 20:1-31.

De Sa C, Chen V, Wong W (2018) Minibatch Gibbs sampling on large graphical models. *Proceedings of the 35th International Conference on Machine Learning*.

Dubé JP, Gupta S (2008) Cross-brand pass-through in supermarket pricing. *Marketing Sci.* 27(3):324-333.

Gelman A, Rubin DB (1992) Inference from iterative simulation using multiple sequences. *Statistical Sci*. 7(4): 457-472.

Gelman A, Carlin JB, Stern HS, Dunson DB, Vehtari A, Rubin DB (2013) *Bayesian Data Analysis*. 3rd ed. Boca Raton, FL: CRC Press.

Ghosh S, Ungureanu AB, Sudderth EB, Blei DM (2011) Spatial distance dependent Chinese restaurant processes for image segmentation. *Advances in Neural Information Processing Systems*. 24.

Hoch SJ, Kim BO, Montgomery AL, Rossi PE (1995) Determinants of store-level price elasticity. *J. Marketing Res*. 32(1):17-29.

Kamakura, WA, Russell, GJ (1989) A probabilistic choice model for market segmentation and elasticity structure. *J. Marketing Res.* 26(4).

Kass RE, Raftery AE (1995) Bayes factors. *J. American Statistical Association*. 90(430):773-795.

Kim M, Bradlow ET, Iyengar R (2022) Selecting data granularity and model specification using the scaled power likelihood with multiple weights. *Marketing Sci*. 41(4):848-866.

Kvalseth TO (1987) Entropy and correlation: Some comments. *IEEE Transactions on Systems, Man, and Cybernetics*. 17(3):517-519.

Li D, Wong WH (2018) Mini-batch tempered MCMC. *arXiv preprint arXiv:1707.09705*

Morwitz VG, Schmittelein D (1992) Using segmentation to improve sales forecasts based on purchase intent: Which “intenders” actually buy? *J. Marketing Res.* 29(4).

Musalem A, Bradlow ET, Raju JS (2008) Who's got the coupon? Estimating consumer preferences and coupon usage from aggregate information. *J. Marketing Res.* 45(6).

Rand WM (1971) Objective criteria for the evaluation of clustering methods. *J. American Statistical Association*. 66(336):846-850.

Smith AN, Rossi PE, Allenby GM (2019) Inference for product competition and separable demand. *Marketing Sci*. 38(4):690-710.

Smolyakov V, Liu Q, Fisher JW (2018) Adaptive scan Gibbs sampler for large scale inference problems. *Proceedings of the 32nd International Conference on Machine Learning*.

Titsias MK, Lázaro-Gredilla M (2014) Doubly stochastic variational bayes for non-conjugate inference. *Proceedings of the 31st International Conference on Machine Learning*.

Vinh NX, Epps J, Bailey J (2009) Information theoretic measures for clusterings comparison: Is a correction for chance necessary? *Proceedings of the 26th Annual International Conference on Machine Learning*. 1073-1080

Voleti S, Gangwar M, Kopalle PK (2017) Why the dynamics of competition matter for category profitability. *J. Marketing*. 81:1-16.

Wedel M, Zhang J (2004) Analyzing brand competition across subcategories. *J. Marketing Res*. 41(4): 448-456.