A Model-based Embedding Technique for Segmenting Customers

‘Customer segmentation’ is the practice of grouping customers into non-overlapping segments (or clusters) such that customers in the same segment have similar needs and preferences. It is now a ubiquitous practice carried out by firms, since it allows them to effectively customize their product offerings, promotions, and recommendations to the particular preferences of each segment. Segmentation also subsumes personalization as a special case by placing each customer into a separate segment of her own. Personalization has gained a lot of traction recently. Yet, in most settings, customizing offerings to coarser segments is more meaningful than personalizing to individual customers simply because firms lack sufficient data for each customer. For example, in the sample dataset we use for our eBay case study, we see that customers often interact with less than 5 items, of eBay’s massive online catalog consisting of more than 4M items.

The biggest challenge to carrying out segmentation is precisely this data sparsity. This challenge has become even more severe with firms being able to collect increasingly fine-grained observations such as direct purchases, ratings, and clicks, in addition to any demographic data such as age, gender, income, etc. These data are not only “big” (consisting of millions of customers and items), but also “complicated” in that they are (a) unstructured, with the items lacking well-defined feature information, (b) diverse, including actions that are represented on different scales (a click is not the same as a purchase is not the same as a rating), and (c) highly sparse, spanning only a small fraction of the entire item universe. Going back to the example above, eBay has a large and diverse product catalog consisting of products ranging from a Fitbit tracker/iPhone (products with well-defined attributes) to obscure antiques and collectibles (that lack any reasonable feature structure). Of these, each customer may click, purchase, or rate only a few items.

In this paper, we revisit the problem of segmentation but in the context of “big” and “complicated” data. These data characteristics pose new and unique challenges. First, traditional techniques within marketing don’t apply. They assume that both customers and items have well-defined and consistent feature representations and often analyze small samples of customer populations. But, when the item universe is large and unstructured, customers can only be represented as large vectors with millions of entries, where each entry captures an action (say, purchase) taken on an item. These representations ‘as is’ are often meaningless for the purposes of segmentation. Almost all of their entries are missing and the lack of consistent feature representations of items means that missing entries can’t be meaningfully imputed—for instance, a customer’s purchase of an iPhone may reveal nothing about her propensity to purchase a particular antique. Existing techniques also become computationally intractable when classifying large populations of customers into segments. Second, the diversity of the types of actions captured in the vectors and their incompleteness
make it difficult to assess similarity of customers. If a customer has clicked an item but
another has purchased it, are they similar? How about customers who have purchased
completely different subsets of items? This difficulty in obtaining a meaningful similarity
measure precludes the application of standard clustering techniques in machine learning,
despite being able to scale to large datasets.

To overcome the above challenges, we propose a model-based embedding technique that
extends extant clustering techniques in machine learning to handle categorical observations
from diverse data sources and having (many) missing entries. We focus on the setting where
the objective of segmentation is to improve the performance on a prediction task. The precise
prediction task depends on the application at hand, and includes predicting the probability
of a customer clicking, purchasing, or liking an item. The algorithm takes as inputs the
observations from a large population of customers and a probabilistic model class describing
how the observations are generated from an individual customer. The choice of the model
class is determined by the corresponding prediction task, as described below, and provides
a systematic way to incorporate domain knowledge by leveraging the existing literature in
marketing, which has proposed rich models describing individual customer behavior. It
outputs an embedding for each customer—a vector representation in a low-dimensional Eu-
clidean space whose dimension is much smaller than the number of items in the universe.
The vector representations are then clustered, using a standard technique such as \( k\)-means,
spectral clustering, mean-shift clustering, etc., to obtain the corresponding segments.

Put together, the algorithm proceeds in two sequential steps: embed and cluster. The
embed step first addresses the issue of diversity of the observed signals by transforming
the categorical observations into a continuous scale that makes different actions (such as
purchases and ratings) comparable. It then deals with the issue of missing data by projecting
the transformed observations onto a low-dimensional space, to obtain a vector representation
for each customer. The cluster step clusters these representations to obtain the segments.

The key novelty of our algorithm is the embed step, which uses a probabilistic model
to convert a categorical observation into its corresponding (log-)likelihood value under the
model. For example, if a customer likes an item with probability \( \alpha \in [0, 1] \), then a “like”
observation is transformed into \( \log \alpha \) and “dislike” observation is transformed into \( \log(1 - \alpha) \).
We call our algorithm model-based because it relies on a probabilistic model. We estimate the
model parameters by pooling together data from all customers and ignoring the possibility
that different customers may have different model parameters. This results in a model that
describes a ‘pooled’ customer—a virtual customer whose preferences reflect the aggregated
preferences of the population. The likelihood transformations then measure how much a par-
ticular customer’s preferences differ from those of the population’s. Our theoretical analysis
shows that under reasonable assumptions, customers in different segments will have different
(log-)likelihood values under the pooled model, allowing us to separate them out.
**Summary of key results.** Our work makes the following contributions:

1. **Novel segmentation algorithm.** Our algorithm is designed to operate on large customer populations and large collections of unstructured items. Moreover, it is (a) *principled*, reducing to standard algorithms in machine learning in special cases; (b) *fast*, with an order of magnitude speedup compared to benchmark latent class models because it requires fitting only one model (as opposed to a mixture model); and (c) *flexible*, allowing practitioners to systematically incorporate problem-dependent structures through the model.

2. **Analytical results.** Under a standard latent class model for generating customer observations, we derive necessary and sufficient conditions for exact recovery of the true segments. Specifically, we bound the asymptotic *misclassification rate*, defined as the expected fraction of customers incorrectly classified, of a nearest-neighbor classifier trained on the embeddings output by the embed step in our algorithm. Our results are similar in spirit to the conditions derived in existing literature for Gaussian mixture models. However, existing proof techniques don’t generalize to our setting. Our results are one of the first to provide such guarantees for latent class preference models.

3. **Empirical results.** We conducted three numerical studies to validate our methodology:

   (a) Using synthetic data, we show that our method recovers more accurate segments, while being upto $17 \times$ faster, than the standard latent class (LC) benchmark.

   (b) On the publicly available MovieLens dataset, we apply our segmentation method to solve the classical *cold-start problem*, which involves recommending new movies to users. We show that segmenting users via our method and customizing recommendations to each segment improves the recommendation accuracy by 48%, 55%, and 84% for drama, comedy, and action genres, respectively, when compared to a baseline method that treats all users as having the same preferences. It also outperforms the standard LC (by upto 13%) and empirical bayesian (by upto 19%) benchmarks used for capturing heterogeneity in customer preferences.

   (c) On a real-world dataset from eBay—spanning a 2-week period and consisting of $\sim 1M$ customers and $\sim 4M$ items—we apply our segmentation methodology for personalizing similar product recommendations. We show that segmenting the population using our approach and customizing recommendations to each segment can result in upto 8% improvement in the recommendation quality, when compared to treating the population as homogeneous. The improvement of 8% is non-trivial because before our method, eBay tried several natural ways to segment (by similarity of demographics, frequency/recency of purchases, etc.), but the best of them resulted in $\sim 1\%$ improvement.