Faster estimation of discrete choice models via weighted dataset reduction

Nicola Ortelli*[1,2], Matthieu de Lapparent[1], and Michel Bierlaire[2]

[1]School of Management and Engineering Vaud, HES-SO, Switzerland
[2]École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

Short summary

When estimating discrete choice models, the prospect of using ever-larger datasets is limited by the poor scalability of maximum likelihood estimation. This paper proposes a simple and fast dataset reduction method that is specifically designed to preserve the richness of observations originally present in a dataset, while reducing its size. Our approach leverages locality-sensitive hashing to create clusters of similar observations, from which representative observations are then sampled and weighted. We demonstrate the efficacy of our approach by applying it on a real-world mode choice dataset; the obtained results confirm that a carefully selected and weighted subsample of observations is capable of providing close-to-identical estimation results while being, by definition, less computationally demanding.
Keywords: discrete choice models, maximum likelihood estimation, dataset reduction, sample size, locality-sensitive hashing

1 Introduction

When estimating DCMs, the use of ever-larger datasets raises two issues: (i) the number of possible model specifications exponentially grows with the number of potential explanatory variables, implying that analysts must spend more time searching for good models; and (ii) the computational cost of maximum likelihood estimation increases with the number of observations, quickly becoming intractable even for basic model structures. While the first issue has spurred great interest (van Cranenburgh et al., 2021), the second has received much less attention: to deal with the increased computational cost associated with large datasets, effort has mostly been dedicated to improving the optimization methods used to estimate DCMs (Lederrey et al., 2021; Rodrigues, 2022) or to enhancing their implementation (Molloy et al., 2021; Arteaga et al., 2022).

This study explores an alternative approach, which consists in reducing the size of datasets by sampling their observations. Removing observations from a dataset is usually advised against by econometricians and choice modelers, but has nevertheless become common practice when training machine learning models on large amounts of data: given the iterative nature of model specification, the use of a smaller sample that provides good approximations of the model's quality allows for early modeling decisions to be taken significantly faster (Park et al., 2019).

To the best of our knowledge, only two studies explore this same idea in the context of discrete choice modeling. The first one is van Cranenburgh & Bliemer (2019): their proposed method scales down any dataset to a predefined fraction of its original size while iteratively minimizing an estimate of the D-error, obtained by means of a simplified version of the model of interest. The second direct precedent of this study is Schmid et al. (2022), in which the authors use the k-means algorithm to identify clusters of similar observations and sample from those as a pre-processing step. Both methods are computationally heavy, which severely limits their usage.

We propose a simple and extremely fast dataset reduction technique that is designed to introduce as little bias as possible in the parameter estimates of the model of interest. Our approach leverages locality-sensitive hashing (LSH) to create clusters from which representative observations are sampled, similar to Schmid et al. (2022). The observations obtained in such way are then given weights that are proportional to the sizes of the clusters they represent, so as to mimic the full dataset during the model estimation process. As argued in the following sections, we believe that a carefully selected *and weighted* subsample of observations is capable of providing close-to-identical estimation results while being, by definition, less computationally demanding.

2 Methodology

Intuition

Consider a choice dataset of N observations (x_n, i_n), each consisting of a vector x_n of explanatory variables associated with individual n, together with the observed choice i_n of that same individual among J alternatives. In its simplest form, a discrete choice model $P(i \mid x_n; \theta)$ calculates the probability that individual n chooses alternative i as a function of x_n and θ, where θ is a vector of model parameters to be estimated from the data.
The values of the model parameters are typically determined through maximum likelihood estimation, which consists in finding the values of θ that maximize the joint probability of replicating all observed choices in the dataset. In practice, the logarithm of the likelihood is usually maximized instead, for numerical reasons. The *log likelihood* function is therefore defined as

$$\mathcal{L}(\theta) = \sum_{n=1}^{N} \log P\left(i_n \mid x_n; \theta\right). \tag{1}$$

Let us now assume that the dataset contains some observations that are identical in all explanatory variables and in the observed choice. By partitioning the observations into $G < N$ groups of identical observations, we may rewrite (1) as

$$\mathcal{L}(\theta) = \sum_{g=1}^{G} N_g \cdot \log P\left(i_g \mid x_g; \theta\right), \tag{2}$$

where N_g denotes the size of group g and i_g and x_g are the observed choice and explanatory variables shared by all observations in group g, respectively. (1) and (2) are equivalent and, as such, yield the exact same parameter estimates when maximized. However, since $G < N$, the computational cost of evaluating (2) is smaller, by a ratio of approximately $\frac{G}{N}$.
The idea behind our dataset reduction method is to extend this "factorization trick" to observations that are nearly identical. The number of distinct groups is thereby further reduced and so is the computational time associated with evaluating the log likelihood function and its gradient. This comes at the cost of degrading the estimation results, because part of the information contained in the dataset is lost; still, the use of an adequate clustering scheme limits said degradation while granting our method a certain reliability. The clustering technique chosen for this purpose is locality-sensitive hashing (LSH), which we introduce now.

Locality-sensitive hashing

LSH is an efficient method for gathering similar data points into clusters — or *buckets*. It achieves this goal by combining the outcomes of several hashing functions, designed in such way that pairs of items are more likely to be hashed to the same bucket if they are close to each other in their original space than if they are far apart. A considerable advantage of LSH over other clustering techniques is that its computational complexity is linear in the number of items to be hashed.
A *family* of LSH functions $\mathcal{H} = \{h : (M, d) \to \mathbb{Z}\}$ is a collection of functions h that map elements of a metric space (M, d) onto the set of integers $\mathbb{Z}$ (Leskovec et al., 2020). Each integer represents a different bucket, and two data points x_p and x_q belong to the same bucket of function h if and only if $h(x_p) = h(x_q)$. For instance, a well-known family of LSH functions is given by

$$h_{a,b}(x) = \left\lfloor \frac{a \cdot x + b}{w} \right\rfloor, \tag{3}$$

where $\lfloor \cdot \rfloor$ denotes the floor function, a is a vector whose entries are independently drawn from a normal distribution $\mathcal{N}(0, 1)$, b is a real value chosen uniformly from the range $[0, w)$ and w is the bucket width (Arnaiz-González et al., 2016). One may see (3) as a projection of all data points onto a line whose direction is given by vector a; an offset equal to b is added to all projected points before the line is discretized into uniform intervals of size w. All data points that fall in the same interval are therefore assigned to the same bucket.
The value of w is context-dependent. By changing the bucket width, one can choose an appropriate degree of similarity within buckets: a sufficiently small w only groups points that are exactly identical, whereas greater values result in fewer buckets that contain larger amounts of increasingly dissimilar points.

Another way of improving the discriminative power of LSH is to combine several hash functions. In the case of the family defined by (3), suppose a and b are drawn R times: now, two data points x_p and x_q belong to the same bucket if and only if they are hashed together by all R functions, *i.e.*:

$$H_{A,B}(x_p) = H_{A,B}(x_q) \iff h_{a_r,b_r}(x_p) = h_{a_r,b_r}(x_q) \ \forall r = 1, \dots, R, \tag{4}$$

where $A = (a_1, \dots, a_R)$ and $B = (b_1, \dots, b_R)$ gather the R realizations of a and b, respectively. Increasing R reduces the joint probability that two data points are grouped together by all projections.

LSH-based dataset reduction (LSH-DR)

Our dataset reduction algorithm has three ingredients: (i) an LSH function or a combination of LSH functions capable of partitioning a dataset of size N into buckets that contain similar observations; (ii) a sampling strategy, based on which some observations are selected from each bucket; and (iii) a weighting scheme that assigns a weight N_g to each selected observation (x_g, i_g). The G observations obtained in such way, together with their associated weights $N_1, \dots, N_G$, constitute the outcome of our method. Any model of interest may then be estimated on the obtained subsample rather than on the whole dataset by using the log likelihood function of (2), with i_g and x_g now referring to the observed choice and explanatory variables associated with the g-th selected observation, respectively.

Clustering Our method uses the family of LSH functions introduced in (3) with, as parameters, the discretization step w and the number of projections R. It is crucial that prior to hashing, all explanatory variables are normalized such that their values are between 0 and 1. The individuals' choices are not taken into account during the clustering, which implies that the buckets might be heterogeneous, *i.e.*, observations with different chosen alternatives might end up in the same bucket.

Sampling The current version of our method randomly selects one observation from each alternative in each bucket.

Weighting Each selected observation (x_g, i_g) is given a weight N_g that is equal to the number of observations that share the same bucket $H_{A,B}(x_g)$ and the same chosen alternative i_g:

$$N_g = \left| \{(x_n, i_n) \mid H_{A,B}(x_g) = H_{A,B}(x_n),\ i_g = i_n\} \right|. \tag{5}$$

Jointly, the adopted sampling strategy and weighting scheme guarantee that the sum of all weights is equal to the number of observations in the full dataset.

3 Results and discussion

The efficacy of our method is demonstrated by means of a series of experiments based on the London passenger mode choice (LPMC) dataset (Hillel et al., 2018). The dataset consists of more than 81'000 trip records collected over three years. Four modes are distinguished: walk, cycle, ride public transport and drive. We divide the dataset into two parts: the first two years of data — 54'766 observations — are used for model estimation whilst the final year — 26'320 observations — is set aside for out-of-sample validation.

We use the data to train two multinomial logit models that we borrow from Hillel (2019). We refer to those as "MNL-S" and "MNL-L". The former includes 10 continuous variables and 13 associated parameters, whereas the latter considers 11 continuous variables, 8 categorical variables encoded using binary indicators and 53 associated parameters. All model estimations are performed using the Biogeme package for Python (Bierlaire, 2018, 2020) on a 2.3 GHz 32-core cluster node with 192 GB of RAM.

Experiment A

We begin by estimating the MNL-S on samples generated by our proposed method. We apply the LSH-DR algorithm on the LPMC data 10'000 times, with $R = 4$ and w ranging from 0.02 to 0.2. The obtained samples range from 1'361 to 48'206 observations in size, that is, from 2% to

88% of the full dataset. The results are shown in Figure 1. We report the following quantities: (i) the execution time, which consists of the sampling and estimation times; (ii) the normalized out-of-sample log likelihood (OSLL), *i.e.*, the log likelihood yielded by the estimated model on the validation data, normalized by the number of observations; (iii) the mean absolute percentage error (MAPE) of the parameter estimates; and (iv) the value of time for the "drive" alternative, computed as the ratio between the estimates of the parameters associated with travel time and cost. For comparative purposes, we also report the results obtained on random samples.
Figure 1 demonstrates that LSH-DR is capable of producing substantially better samples than random sampling, for a negligible increase in execution time: down to approximately 40% of the full dataset size, the samples generated by LSH-DR yield smaller MAPEs of the parameters and more accurate estimates of the value of time.

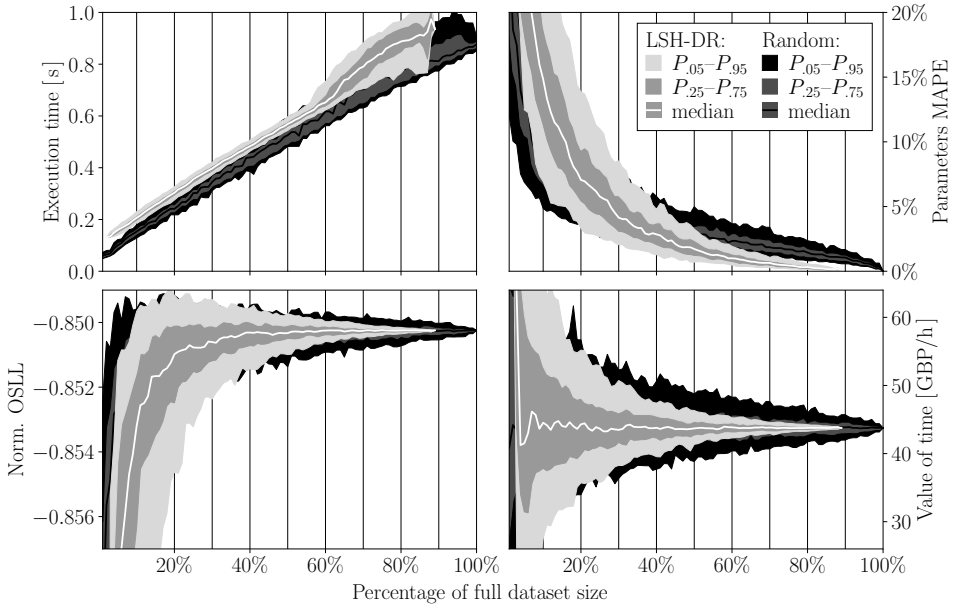

Figure 1: Estimation of the MNL-S on samples generated by LSH-DR.

Experiment B

In this experiment, we compare the performance of our method with three other dataset reduction techniques, namely: (i) random sampling; (ii) k-means clustering, similar to the approach taken in Schmid et al. (2022); and (iii) sampling of observations (SoO), as proposed by van Cranenburgh & Bliemer (2019). We proceed as follows: a certain percentage of the full dataset size is chosen and we retrieve from Experiment–B the 100 samples of size closest to that percentage; the three other dataset reduction techniques are then used to generate samples of those exact same sizes, which are finally used to train the MNL-S model. Figure 2 reports the sampling time, normalized OSLL, parameters MAPE and value of time for the "drive" alternative for 25%, 50% and 75% of the full dataset. The boxplot whiskers indicate the 5^{th} and 95^{th} percentiles. The size of the samples retrieved from Experiment B ranges from 13'480 to 13'984, from 27'146 to 27'615, and from 40'879 to 41'273, respectively.
Overall, Figure 2 illustrates that the samples producing the most accurate results are obtained via k-means. Still, despite its superiority, k-means is practically unusable because of its runtime: it takes from 8 up to 26 minutes to obtain a sample from a relatively small dataset. That is between 4'000 and 16'000 times longer than LSH-DR and up to 400'000 times longer than random sampling. As regards SoO, the method is shown to provide the worst results, while also displaying the largest runtimes. This is due to the fact that SoO is designed to maximize the efficiency of the parameter estimates rather than their precision or the model's predictive accuracy.

Experiment C

Lastly, we estimate the MNL-L model on samples generated by LSH-DR. To this end, we apply the LSH-DR algorithm on the LPMC data 10'000 times, with $R = 4$ and w ranging from 0.1 to

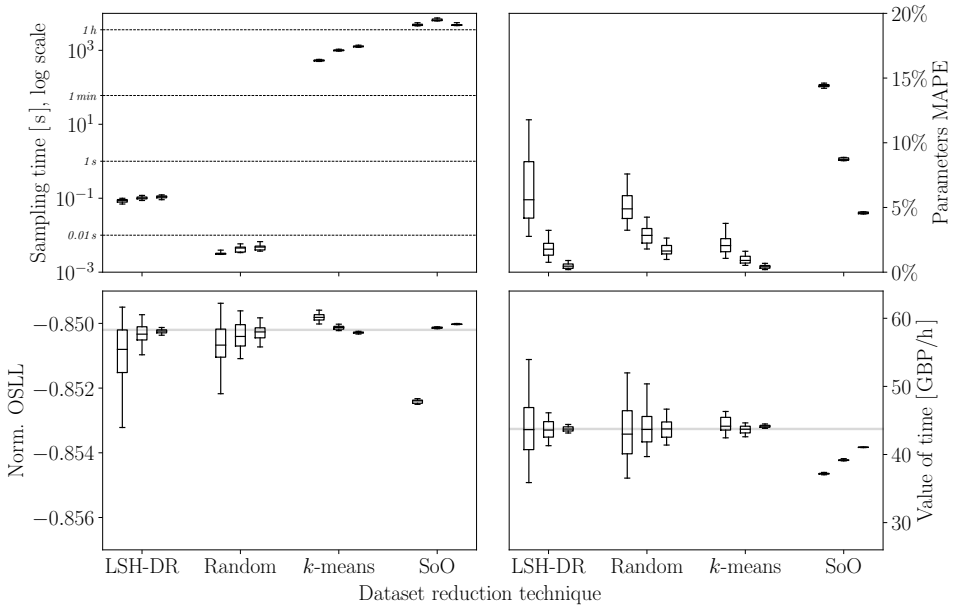

Figure 2: Comparison of dataset reduction techniques.

1. Note that the MNL-L model includes several discrete explanatory variables, but those are not treated differently by the LSH-DR method. The generated samples range from 3'584 to 51'574 observations in size, that is, from 7% to 94% of the full dataset. Figure 3 displays the achieved results. For the sake of comparison, the results obtained on random samples are also shown.
Figure 3 demonstrates that our method may also be beneficial to larger models. Overall the improvement in comparison to random sampling may be less remarkable than with the smaller model, but it is worth noting that the MAPE remains within a reasonable level of accuracy even for samples down to 50% of the full dataset size.

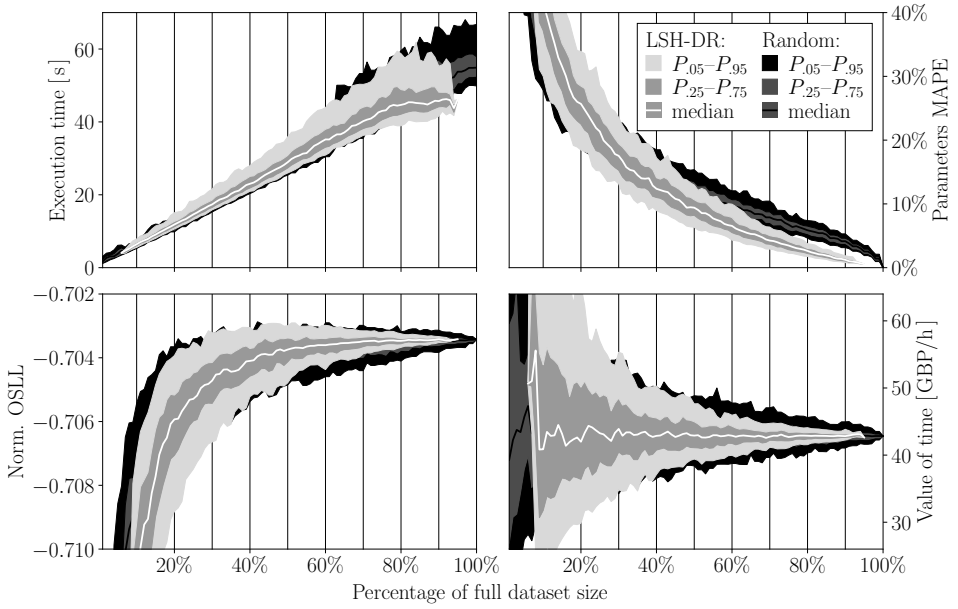

Figure 3: Estimation of the MNL-L on samples generated by LSH-DR.

4 Conclusions

In this paper, we propose a simple and fast dataset reduction technique that speeds up the estimation of discrete choice models. The gain in computational time naturally comes at the cost of

deteriorating the model estimation results; however, our method is specifically designed to mitigate this deterioration by preserving as much diversity as possible among the observations. As a result, the quality of the parameter estimates stays within reasonable ranges even for large reduction rates. The presented results additionally highlight the benefits of our method on the estimation of models of small and medium size.
Intended future work includes the development and testing of more elaborate sampling strategies for selecting observations from buckets. For instance, those could be designed to increase the probability of choosing the most representative observations within each bucket, or to rely on the content of each bucket to generate synthetic prototypical observations. Additional investigation could also consist in developing LSH functions that can accommodate the analyst's knowledge of the dataset or the structure of the model of interest, for instance by giving more importance to some specific variables during the hashing. Finally, another promising direction of research consists in embedding the LSH-DR method within a stochastic gradient descent algorithm for model estimation, such as the one proposed in Lederrey et al. (2021). The use of carefully selected and weighted batches of data, rather than random ones, could result in significantly better approximates of the gradient and, as a result, speed up convergence.

References

Arnaiz-González, Á., Díez-Pastor, J.-F., Rodríguez, J. J., & García-Osorio, C. (2016). Instance selection of linear complexity for big data. *Knowledge-Based Systems*, *107*, 83–95.

Arteaga, C., Park, J., Beeramoole, P. B., & Paz, A. (2022). xlogit: An open-source Python package for GPU-accelerated estimation of mixed logit models. *Journal of Choice Modelling*, *42*, 100339.

Bierlaire, M. (2018). *PandasBiogeme: a short introduction* (Tech. Rep.). TRANSP-OR 181219. Transport and Mobility Laboratory, ENAC, EPFL.

Bierlaire, M. (2020). *A short introduction to pandasBiogeme* (Tech. Rep.). TRANSP-OR 200605. Transport and Mobility Laboratory, ENAC, EPFL.

Hillel, T. (2019). *Understanding travel mode choice: A new approach for city scale simulation* (Unpublished doctoral dissertation). University of Cambridge.

Hillel, T., Elshafie, M. Z., & Jin, Y. (2018). Recreating passenger mode choice-sets for transport simulation: A case study of London, UK. *Proceedings of the Institution of Civil Engineers-Smart Infrastructure and Construction*, *171*(1), 29–42.

Lederrey, G., Lurkin, V., Hillel, T., & Bierlaire, M. (2021). Estimation of discrete choice models with hybrid stochastic adaptive batch size algorithms. *Journal of choice modelling*, *38*, 100226.

Leskovec, J., Rajaraman, A., & Ullman, J. D. (2020). *Mining of massive data sets*. Cambridge university press.

Molloy, J., Becker, F., Schmid, B., & Axhausen, K. W. (2021). mixl: An open-source R package for estimating complex choice models on large datasets. *Journal of choice modelling*, *39*, 100284.

Park, Y., Qing, J., Shen, X., & Mozafari, B. (2019). BlinkML: Efficient maximum likelihood estimation with probabilistic guarantees. In *Proceedings of the 2019 international conference on management of data* (pp. 1135–1152).

Rodrigues, F. (2022). Scaling bayesian inference of mixed multinomial logit models to large datasets. *Transportation research part B: methodological*, *158*, 1–17.

Schmid, B., Becker, F., Molloy, J., Axhausen, K. W., Lüdering, J., Hagen, J., & Blome, A. (2022). Modeling train route decisions during track works. *Journal of Rail Transport Planning & Management*, *22*, 100320.

van Cranenburgh, S., & Bliemer, M. C. (2019). Information theoretic-based sampling of observations. *Journal of choice modelling*, *31*, 181–197.

van Cranenburgh, S., Wang, S., Vij, A., Pereira, F., & Walker, J. (2021). Choice modelling in the age of machine learning-discussion paper. *Journal of Choice Modelling*, 100340.