Learning Interpretable Feature Context Effects in Discrete Choice

Kiran Tomlinson
Cornell University
kt@cs.cornell.edu

Austin R. Benson
Cornell University
arb@cs.cornell.edu

ABSTRACT

Individuals are constantly making choices—purchasing products, consuming Web content, making social connections—so understanding what contributes to these decisions is crucial in many settings. A major interest is understanding context effects, which occur when the set of available options itself affects an individual's relative preferences. These violate traditional rationality assumptions but are commonly observed in human behavior. At the same time, identifying context effects from choice data remains a challenge; existing models posit a specific context effect a priori and then measure its effect from (often effect-targeting) data. Here, we develop discrete choice models that capture a broad range of context effects, which are learned from choice data rather than baked into the model. Our models yield intuitive, interpretable, and statistically testable context effects, all while being simple to train. We evaluate our model on several empirical choice datasets, discovering, e.g., that people are more willing to book higher-priced hotels when presented with options that are on sale. We also provide the first analysis of context effects in online social network growth, finding that users forming connections place relatively more emphasis on shared neighbors when popular users are an option.

CCS CONCEPTS

• **Mathematics of computing** → **Probabilistic inference problems**; • **Applied computing** → **Economics**; • **Information systems** → **Social networks**.

KEYWORDS

discrete choice; context effects; preference learning

ACM Reference Format:
Kiran Tomlinson and Austin R. Benson. 2021. Learning Interpretable Feature Context Effects in Discrete Choice. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '21), August 14–18, 2021, Virtual Event, Singapore.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3447548.3467250

1 DISCRETE CHOICE & CONTEXT EFFECTS

In a *discrete choice* setting, an individual chooses between a finite set of available items called a *choice set*. This general framework describes a host of important scenarios, including purchasing [2], transportation decisions [59], voting [14], and the formation of new social connections [21, 43]. Discovering and understanding the factors that contribute to the choices people make has broad applications in, e.g., recommender systems [53, 67], Web search [24], online dating platforms [9], and policy design [8].

Two popular models of human choice are the Plackett–Luce [36, 48] and conditional logit (CL) [39] models. Both of these obey the axiom of *independence of irrelevant alternatives* (IIA) [36], that relative preferences between items are unaffected by the choice set—if someone prefers x to y, they should still do so when z is also an option. However, experiments on human decisions [23, 55, 56, 60] as well as direct measurement on choice data [6, 54, 57] have found that this assumption often does not hold in practice. These "IIA violations" are termed *context effects* [49, 50]. Examples include the *attraction effect* [23], where including an inferior item makes a better option more attractive, and the *similarity effect* [61], where similar items split the preferences of the chooser.

The ubiquity of context effects has driven the development of more nuanced models capable of capturing them. In machine learning, the goal is typically to design models for better predictions via learned context effects [7, 10, 11, 47, 51, 53, 54]. However, the effects accounted for by models using neural networks or item embeddings [11, 47, 51] are difficult to interpret. Other models learn context effects at the level of individual items [10, 42, 53, 54], preventing generalization to items not in the training set and making it difficult to discover context effects coming from item features (e.g., price). Within behavioral economics, context effect models tend to be engineered for specific effects and are often only applied to controlled special-purpose datasets [38, 50, 62].

Here, we provide methods for learning a wide class of context effects from large, pre-existing choice datasets in a variety of domains. The key advantage of our approach is that we can take a choice dataset collected in any domain (possibly collected passively), efficiently train a model, and directly interpret the learned parameters as intuitive context effects. For example, we find in a hotel booking dataset that users presented with more hotels on sale showed increased willingness to pay. This lets us hypothesize that "on sale" tags on hotels exerts a context effect on the user, making them feel better about selecting a more expensive option. Context effects extracted by our methods could then motivate further experimental work such as A/B testing, or choice set design to steer behavior. We focus on the setting where items are described by a set of features (e.g., for hotels: price, star rating) and where the utility of each item is a function of its features. This setup has two major benefits, as it enables (i) making predictions about new items not observed in training data, and (ii) learning generalizable and testable effects that can inform marketing, advertising, or recommendation.

KDD '21, August 14–18, 2021, Virtual Event, Singapore

ACM ISBN 978-1-4503-8332-5/21/08…$15.00
https://doi.org/10.1145/3447548.3467250

We define *feature context effects* that describe changes in importance of features when determining choice as a function of features of the choice set. For instance, suppose a diner has two choice sets on two different occasions, one consisting of fast food chains and the other of high-end restaurants. In the choice set with lower prices, the diner could value service speed relatively more, and in the choice set with higher prices, the diner might place more weight on wine selection. We introduce two models, the linear context logit (LCL) and decomposed linear context logit (DLCL), to learn these types of feature context effects directly from choice data.

We perform an extensive analysis of choice datasets using our models, showing that statistically significant feature context effects occur in empirical data and recovering intuitive effects. For example, we find evidence that people pick more expensive hotels when their choice sets have high star ratings, that people offered more oily sushi show more aversion to oiliness (a possible similarity effect), and that when deciding whose Facebook wall to post on, people care more about mutual connections when choosing from popular friends.[1] Accounting for feature context effects also improves prediction accuracy in many datasets, although our primary focus is learning interpretable context effects. Additionally, we show to statistically test for effects and how sparsity-encouraging regularization can identify the most influential context effects.

Our empirical study is split into two parts. First, we examine datasets collected to understand preferences, covering a variety of choices including sushi, hotel bookings, and cars. Second, we apply our methods to social network analysis, where we demonstrate context effects in competing theories of triadic closure, the of new friendships to form among friends-of-friends [15, 19]. Discrete choice models have recently found compelling use in analyzing social network dynamics [16, 21, 43, 44]. Here, we find new insights by incorporating context effects.

1.1 Additional related work

Within machine learning, our LCL model is similar in spirit to the context-dependent random utility model (CDM) [54] in that we consider pairwise contextual interactions, but with the important distinction that our model operates on features rather than on items, allowing for the discovery of general, non-item-specific effects. Our framework for context-dependent utilities is related to set-dependent weights [51] and FETA [47]; these methods are optimized for prediction accuracy and are difficult to interpret. Other models for context effects include the blade-chest model [10, 11] for pairwise comparisons and the salient features model [7], which considers different subsets of features in each choice set.

Recent research has framed network growth (the formation of new connections in, e.g., communication or friendship networks) as discrete choice and have suggested context effects as a means for more flexible modeling [43]. The models we introduce are a step in this direction, and we find context effects useful for both improved predictions and gaining new social insights. Other research has explored mixed [21] and de-mixed [44] choice models for network growth, but these approaches do not reveal if or how the features of items in each choice set affect the preferences of choosers.

[1] By necessity, these are all correlative rather than causal claims; we discuss this more in the results section.

2 DISCRETE CHOICE BACKGROUND

We briefly review the discrete choice modeling framework (see the book by Train [59] for a thorough treatment). In a discrete choice setting, an individual selects one item from a set of available items, the choice set. We use $\mathcal{U}$ to denote the universe of all items and $C \subseteq \mathcal{U}$ the choice set in a particular choice instance. A choice dataset $\mathcal{D}$ is a set of n pairs (i, C), where $i \in C$ is the item selected. Each item i is described by a vector of d features $y_i \in \mathbb{R}^d$ that determine the preferences of the chooser.

Random utility models [37] (RUMs) are of particular interest and are based on the idea that individuals try to maximize their utility, but can only do so noisily. In a RUM, an individual draws a random utility for each item (where each item has its own utility distribution) and selects the item with maximum observed utility. The workhorse RUM using item features is the conditional logit (CL) [39], which has interpretable parameters that are readily estimated from data (the CL is sometimes called the *multinomial logit*). In the CL model, the observed utility of each item i is the random quantity $\theta^T y_i + \epsilon$, where the latent parameter $\theta \in \mathbb{R}^d$ (the *preference vector*) stores the relative importance of each feature (the *preference coefficients*) and the random noise term ϵ follows a standard Gumbel distribution with CDF $e^{-e^{-x}}$. This noise distribution is chosen so that the CL choice probabilities have a simple closed form [59]: a softmax over the utilities. Under a CL, the probability that i is chosen from the choice set C, denoted $\Pr(i, C)$, is

$$\Pr(i, C) = \frac{\exp(\theta^T y_i)}{\sum_{j \in C} \exp(\theta^T y_j)}. \tag{1}$$

The CL model famously obeys the axiom of *independence of irrelevant alternatives* (IIA) [36], stating that relative choice probabilities are unaffected by the choice set. Formally, a model satisfies IIA if for any two choice sets C, C' and items $i, j \in C \cap C'$,

$$\frac{\Pr(i, C)}{\Pr(j, C)} = \frac{\Pr(i, C')}{\Pr(j, C')}.$$

As we have discussed, this assumption is often violated in practice due to context effects. One model that can account for (some) context effects is the mixed logit. The DLCL model that we will introduce is related to a discrete mixed logit, so we briefly describe it here. In a discrete mixed logit, there are M populations, each of which has its own preference vector θ_m. The mixing parameters $\pi_1, \ldots, \pi_M$, with $\sum_{m=1}^{M} \pi_m = 1$, describe the relative sizes of the populations. This results in choice probabilities

$$\Pr(i, C) = \sum_{m=1}^{M} \pi_m \frac{\exp(\theta_m^T y_i)}{\sum_{j \in C} \exp(\theta_m^T y_j)}. \tag{2}$$

While mixed logit can produce IIA violations, it does so by hypothesizing populations each with their own context-effect-free preferences, meaning that context effects only appear in the aggregate data. In contrast, our models are designed to identify context effects in individual preferences.

3 MODELS OF FEATURE CONTEXT EFFECTS

In order to capture context effects at the individual level, the choice set itself needs to influence the preferences of a chooser. In the most general extension of the CL, we could replace θ with $\theta + F(C)$,

where $F\colon \mathcal{P}(\mathcal{U}) \to \mathbb{R}^d$ is an arbitrary function of the choice set (this is analogous to the set-dependent weights model [51], but framed as a RUM). This allows each feature to exert an arbitrary influence on the base preference coefficient of each other feature. We say that a *feature context effect* occurs when $F(C) \neq 0$.

We make two simplifying assumptions on the choice set effect function $F(C)$ that will aid interpretability. The first is that the effect of a choice set additively decomposes into effects of its items, i.e., $F(C)$ is proportional to $\sum_{j\in C} f(y_j)$ for some function $f\colon \mathcal{U} \to \mathbb{R}^d$. While in principle higher-order interactions are possible, the number of such interactions is exponential in the size of the choice set. This makes it difficult to extract such effects from typical choice datasets that do not contain observations from every possible choice set; moreover, higher-order interactions are usually sparse [3]. Second, we assume that the effect of each item is diluted in large choice sets and we model this with a proportionality constant of $1/|C|$ so that $F(C) = 1/|C| \sum_{j\in C} f(y_j)$.

3.1 Linear Context Logit

In principle, features could exert arbitrary influences on each other, but we focus on the case when context effects are linear, which makes inference tractable and, crucially, preserves interpretability. We use $y_C = 1/|C| \sum_{j\in C} y_j$ to denote the mean feature vector of the choice set C. For f linear, we can write $f(y_j) = Ay_j$ for some matrix $A \in \mathbb{R}^{d\times d}$, and the choice set context function F is

$$F(C) = \frac{1}{|C|}\sum_{j\in C} f(y_j) = \frac{1}{|C|}\sum_{j\in C} Ay_j = Ay_C.$$

We call this model the *linear context logit (LCL)*, and it produces choice probabilities

$$\Pr(i, C) = \frac{\exp([\theta + Ay_C]^T y_i)}{\sum_{j\in C} \exp([\theta + Ay_C]^T y_j)}. \tag{3}$$

In the LCL, A_{pq} specifies the effect of feature q on the coefficient of feature p. If A_{pq} is positive (resp. negative), then higher values of q in the choice set result in a higher (resp. lower) preference coefficient for p. If $A = 0$, then the LCL reduces to CL.

When analyzing data in Section 6, we often see large diagonal entries of A. The signs of the diagonal entries of A can be explained by known context effects. The case of $A_{pp} > 0$ relates to the attraction effect (high values of a feature amplify fine-grained differences along that dimension), and the case of $A_{pp} < 0$ is consistent with the similarity effect (high values of a feature devalue it),

Just as in the CL, we can derive the closed form in (3) if choosers have random utilities $[\theta + Ay_C]^T y_i + \epsilon$, where ϵ follows a standard Gumbel distribution and the random variable samples are i.i.d. If we want a more parsimonious model, we can impose sparsity on A through L_1 regularization (we do this in our empirical analysis) or we could use a low-rank approximation of A. A constant-rank approximation makes the number of parameters linear in d.

3.2 Decomposed Linear Context Logit

The LCL implicitly assumes that the intercepts of all linear context effects exerted by one feature are the same (we have d^2 slopes in A, but only d intercepts in θ). Motivated by varying intercepts in empirical data (Figure 1), we develop a second model that decomposes the LCL into context effects exerted by each feature, which we call the *decomposed linear context logit (DLCL)*. In the language of choice set effect functions, we now have d context effect functions $F_1, \ldots, F_d$ where F_k only depends on the values of feature k. We also replace θ with d base preference vectors $B_1, \ldots, B_d$ (which we combine into a $d\times d$ matrix B; subscripts index columns) that provide varying intercepts. This gives us d contextual utilities $B_1 + F_1(C), \ldots, B_d + F_d(C)$ that we combine in a mixture model.

Making the same assumptions as for the LCL, we decompose each choice set effect function $F_k(C) = \frac{1}{|C|}\sum_{j\in C} f_k((y_j)_k)$ (here, f_k is a function of only the kth feature, $(y_j)_k$). Assuming linearity (and storing context effects exerted by feature k in the kth column of A), we arrive at

$$F_k(C) = \frac{1}{|C|}\sum_{j\in C} f_k((y_j)_k) = \frac{1}{|C|}\sum_{j\in C} A_k(y_j)_k = A_k(y_C)_k.$$

We use mixture weights $\pi_1, \ldots, \pi_d$ with $\sum_{k=1}^d \pi_k = 1$ to describe the relative strengths of effects exerted by each feature. The DLCL is then a mixture of d logits, where each component captures the context effects from a single feature. The choice probabilities are

$$\Pr(i, C) = \sum_{k=1}^{d} \pi_k \frac{\exp\left([B_k + A_k(y_C)_k]^T y_i\right)}{\sum_{j\in C} \exp\left([B_k + A_k(y_C)_k]^T y_j\right)}. \tag{4}$$

Each component corresponds to an LCL with the constraint that all columns of A except the kth are zero. The matrix A has the same interpretation as in the LCL, while B_{pq} represents the importance of feature p when feature q is zero (i.e., the intercept of the linear context effect exerted on p by q).

4 IDENTIFIABILITY OF THE LCL

Identifiability is a key feature of models that ensures we can uniquely learn parameters and thus interpret them meaningfully. We provide three results characterizing the identifiability of the LCL. Most significantly, we prove a necessary and sufficient condition that exactly determines when the model is identifiable (Theorem 4.1). However, the condition is somewhat hard to reason about, so we also prove a simple necessary condition (Proposition 4.2) and a simple sufficient condition (Proposition 4.3). These results give further insight into the main theorem. Proofs are in Appendix A.

Following Seshadri et al. [54], we use $\mathcal{C}_\mathcal{D}$ to denote the set of unique choice sets appearing in the dataset $\mathcal{D}$, and we say that an LCL is *identifiable* from a dataset if there do not exist two distinct sets of parameters (θ, A) and (θ', A') that produce identical probability distributions over every choice set $C \in \mathcal{C}_\mathcal{D}$. In the following, $\otimes$ denotes the Kronecker product.

Theorem 4.1. *A d-feature linear context logit is identifiable from a dataset $\mathcal{D}$ if and only if*

$$\operatorname{span}\left\{\begin{bmatrix} y_C \\ 1 \end{bmatrix} \otimes (y_i - y_C) \mid C \in \mathcal{C}_\mathcal{D}, i \in C\right\} = \mathbb{R}^{d^2+d}. \tag{5}$$

Theorem 4.1 says that identification requires enough choice sets with sufficiently different mean features containing enough sufficiently different items (with coupling between the two requirements). The condition of Theorem 4.1 is often satisfied in practice if there are no redundant features (18 out of 22 that we analyze uniquely identify the LCL).

To better understand the span condition, we provide a simple necessary condition for indentifiability. Recall that a set of vectors $\{y_0, \ldots, y_d\} \subset \mathbb{R}^d$ is *affinely independent* if the set of vectors $\{y_1 - y_0, \ldots, y_d - y_0\}$ is linearly independent.

Proposition 4.2. *If a d-feature linear context logit is uniquely identifiable from a dataset $\mathcal{D}$, then the dataset must contain $d+1$ choice sets with affinely independent mean feature vectors.*

This necessary condition stems from formulating item utility as the affine transformation $\theta + Ay_C$, which requires $d+1$ points to be identified. The span condition in Theorem 4.1 is more difficult to reason about because of the coupling between individual feature vectors y_i and mean feature vectors y_C. We therefore provide a simple sufficient condition for identifiability that decouples these requirements and is optimal in the number of distinct choice sets.

Proposition 4.3. *If a dataset $\mathcal{D}$ contains $d+1$ distinct choice sets $C_0, \ldots, C_d$ such that*

i. *the set of mean feature vectors $\{y_{C_0}, \ldots, y_{C_d}\}$ is affinely independent (the necessary condition from Proposition 4.2) and*
ii. *in each choice set C_i, there is some set of $d+1$ items with affinely independent features,*

then we can uniquely identify a d-feature LCL.

We leave characterization of DLCL identifiability for future work, as even mixed logits have notoriously complex identifiability conditions [12, 20, 70].

5 ESTIMATION

Given a dataset $\mathcal{D}$ consisting of observations (i, C), where i was selected from the choice set C, we wish to recover the parameters of a model that best describe the dataset. In this section, we describe estimation procedures for the LCL and DLCL. First, we show that the likelihood function of the LCL is log-concave and simple to optimize. On the other hand, the DLCL does not have a log-concave likelihood, but we derive an expectation-maximization algorithm that only requires optimizing convex subproblems.

We wish to find parameters that minimize the negative log-liklihood (NLL) of a model, which is equivalent to maximizing the likelihood. The NLL of the linear context logit is

$$-\ell(\theta, A; \mathcal{D}) = -\sum_{(i,C)\in\mathcal{D}} \log \frac{\exp([\theta + Ay_C]^T y_i)}{\sum_{j\in C} \exp([\theta + Ay_C]^T y_j)} \tag{6}$$

$$= \sum_{(i,C)\in\mathcal{D}} -(\theta + Ay_C)^T y_i + \log \sum_{j\in C} \exp([\theta + Ay_C]^T y_j). \tag{7}$$

This function is convex in θ and A (equivalently, the likelihood is log-concave). To see this, notice that the first term in the summand of (7) is a linear combination of entries of θ and A, so it is jointly convex in θ and A. Meanwhile, log-sum-exp is convex and monotonically increasing, so its composition with the linear functions $[\theta + Ay_C]^T y_j$ is also convex. We then have that $-\ell(\theta, A; \mathcal{D})$ is convex, as the sum of convex functions is convex. Moreover, the second partial derivatives of the NLL function are all bounded (by a constant depending on the dataset), so its gradient is Lipschitz continuous. We can therefore use gradient descent to efficiently find a global optimum of $-\ell(\theta, A; \mathcal{D})$.

On the other hand, the NLL of the DLCL (like that of the mixed logit) is not convex, so we can only hope to find a local optimum with gradient descent. To address this challenge, we develop an expectation-maximization (EM) algorithm for DLCL estimation. The algorithm mirrors the EM algorithm for estimating a mixed logit [59], except that the M step updates estimates for A and B. (see Appendix B). An advantage of EM for DLCL is that it only requires optimizing convex functions with Lipschitz-continuous gradients, and EM is guaranteed to improve the log-likelihood at each step. While EM may still arrive at a local optimum, we find that for most of our datasets, it finds better model parameters than stochastic gradient descent on the likelihood.

6 DATA ANALYSIS

We apply our LCL and DLCL models to two collections of empirical choice datasets. First, we examine datasets specifically collected to understand preference in various domains, such as car purchasing and hotel booking. The features describing items naturally differ in these datasets. The second collection of datasets comes from a particular choice process in social networks, namely the formation of new connections. Here, we use graph properties as features (such as in-degree, a proxy for popularity [41]), allowing us to compare social dynamics across email, SMS, trust, and comment networks. In both dataset collections, we first establish that context effects occur and that our models better describe the data than traditional context-effect-free models, CL, and mixed logit. We then show how the learned models can be interpreted to recover intuitive feature context effects. Our code, results, and links to documented versions of every dataset are available at https://github.com/tomlinsonk/feature-context-effects.

Estimation details. For prediction experiments, we use 60% of samples for training, 20% for validation, and 20% for testing. When testing model fit with likelihood-ratio tests, we estimate models from the entire dataset. We use PyTorch's Adam optimizer for maximum likelihood estimation, with batch size 128 and the `amsgrad` flag. We run the optimizer for 500 epochs or 1 hour, whichever comes first. For the whole-data fits, we use weight decay 0.001 and search over learning rates of 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1, selecting the one that results in the highest likelihood. For our prediction experiments, we perform a grid search over the weight decays 0, 0.0001, 0.0005, 0.001, 0.005, 0.01 and the same learning rates as above, selecting the pair with the best likelihood on the validation set. Predictions are evaluated on the held-out test set. We use d (the number of features) components for mixed logit to provide a fair comparison against DLCL (which always uses d components).

6.1 General Choice Datasets

We analyze six choice datasets from online and survey data (Table 1) previously used in discrete choice research: SUSHI [28]; EXPEDIA [27]; DISTRICT and DISTRICT-SMART [7, 29]; CAR-A and CAR-B [1]; and CAR-ALT [8, 40]. In all datasets, we standardize the features to have zero mean and unit variance, which allows us to more meaningfully compare learned parameters across datasets. The LCL is identifiable in DISTRICT-SMART, EXPEDIA, and SUSHI, but not the others. However, the L_2 regularization we apply (via weight decay) identifies the model in all cases.

Table 1: General choice datasets summary.

Dataset	Choices	Features	Largest Choice Set
DISTRICT	5376	27	2
DISTRICT-SMART	5376	6	2
SUSHI	5000	6	10
EXPEDIA	276593	5	38
CAR-A	2675	4	2
CAR-B	2206	5	2
CAR-ALT	4654	21	6

6.2 Network Datasets

Recent work cast many network growth models in terms of discrete choice [43]. In a directed graph, the formation of the edge $u \to v$ can be thought of as a choice by the node u to initiate new contact with v (the graph might be a citation, communication, or friendship network). The set from which u chooses can vary, including all nodes in the graph or only a subset of "close" nodes. We focus specifically on directed *triadic closure* [15, 19], where the node u closes a triangle $u \to v \to w$ by adding the edge $u \to w$. This phenomenon is used in many influential network growth models [22, 26, 63] and real-world networks show evidence of triadic closure in the form of high clustering coefficients [15] and closure coefficients [69].

Choices from temporal network data. Our network analysis assumes that the graphs grow according to a multi-mode model that combines triadic closure with a method of global edge formation. In particular, we assume that at each step, an initiating node decides to either form an edge to any node in the graph with probability r or close a triangle with probability $1 - r$. This setup, also used by the Jackson–Rogers model [25] and the (r, p)-model [43], singles out instances of triadic closure to study separately from global edge formation. When a node u chooses to close a triangle, we assume u first picks one of its neighbors v uniformly at random before choosing one of v's neighbors as a new connection.

Each time we observe a new edge $u \to w$ closing a previously unclosed triangle, we select a hypothesized intermediate v uniformly at random ($u \to w$ can close multiple triangles at once through different intermediates). We consider the choice set for the closure to be the out-neighbors of v that are not out-neighbors of u.

Node features. The features of each node in the choice set are computed at the instant before the edge is closed (the features evolve as the network grows). In our datasets, we have timestamps on each edge and an edge may be observed many times (e.g., in an email network, u may send w many emails). The number of times an edge is observed is its *weight*; an edge not in the graph has weight 0. We use six features to describe each node w that could be selected by the chooser u: (1) *in-degree*: the number of edges entering the target node w; (2) *shared neighbors*: the number of in- or out-neighbors of u that are also in- or out-neighbors of w; (3) *reciprocal weight*: the weight of the reverse edge $u \leftarrow w$; (4) *send recency*: the number of seconds since w initiated any outgoing edge; (5) *receive recency*: the number of seconds since w received any incoming edge; (6) *reciprocal recency*: the number of seconds since the reverse edge $u \leftarrow w$ was last observed.

Following Overgoor et al. [44], we log-transform features 1 and 2. We take $\log(1 + \text{feature 3})$ to handle weight 0 (in-degree and shared neighbors are never 0, since v is always a shared neighbor of u and w). Lastly, we transform the temporal features with $\log^{-1}(2 + \text{feature})$ and set them to 0 if the event has never occurred. This ensures that (1) we can handle 0 seconds since the last event, (2) higher values mean more recency, and (3) "no occurrence" results in the lowest possible value of the transformed feature.

Table 2: Network datasets summary.

Dataset	Nodes	Edges	Triangle closures
SYNTHETIC-CL	1000	391294	50000
SYNTHETIC-LCL	1000	380584	50000
EMAIL-ENRON	18592	53477	19900
EMAIL-EU	986	24929	19603
EMAIL-W3C	20082	33409	3271
SMS-A	44430	68834	6311
SMS-B	72146	100974	9376
SMS-C	14433	23285	2732
BITCOIN-ALPHA	3783	24186	8823
BITCOIN-OTC	5881	35592	12750
REDDIT-HYPERLINK	23499	91946	37115
WIKI-TALK	22067	81125	27505
FACEBOOK-WALL	46952	274086	68776
MATHOVERFLOW	24818	239978	137455
COLLEGE-MSG	1899	20296	6267

Network datasets. We examine 13 network datasets: three email datasets (EMAIL-ENRON [4], EMAIL-EU [35, 68], EMAIL-W3C [5, 13]); three SMS datasets (SMS-A, SMS-B, and SMS-C [66]), two Bitcoin trust datasets (BITCOIN-ALPHA and BITCOIN-OTC [30, 32]), an online messaging dataset (COLLEGE-MSG [45]), a hyperlink dataset (REDDIT-HYPERLINK [31]), and three online forum datasets (FACEBOOK-WALL [64], MATHOVERFLOW [46], and WIKI-TALK [33, 34]). In addition, we generate two synthetic networks, SYNTHETIC-CL and SYNTHETIC-LCL. Specifically, we begin with 1000 isolated nodes. At each step, we add an edge uniformly at random with probability 0.9. With probability 0.1, we close a triangle by selecting a node u and one of its neighbors v uniformly at random. We then use either a CL (for SYNTHETIC-CL) or LCL (for SYNTHETIC-LCL) to choose which triangle $u \to v \to ?$ to close (if there are no triangles for u to close, we add a random edge). We use the same features as in the empirical datasets, with Poisson-distributed simulated timestamp gaps between successive edges until 50000 triangles are closed.

Table 2 summarizes the network data. Using Theorem 4.1, we find that the LCL is uniquely identifiable in every network dataset. Whereas we split the general choice datasets into training, validation, and testing sets uniformly at random, we instead split the network datasets temporally so that future edges are predicted based on parameters estimated from past edges.

6.3 Results

Our analysis focuses on two issues: whether significant linear feature context effects appear in practice and if so, how we can identify and interpret them using our models.

Binned CLs for visualizing feature context effects. As a first step towards identifying whether linear context effects occur, we

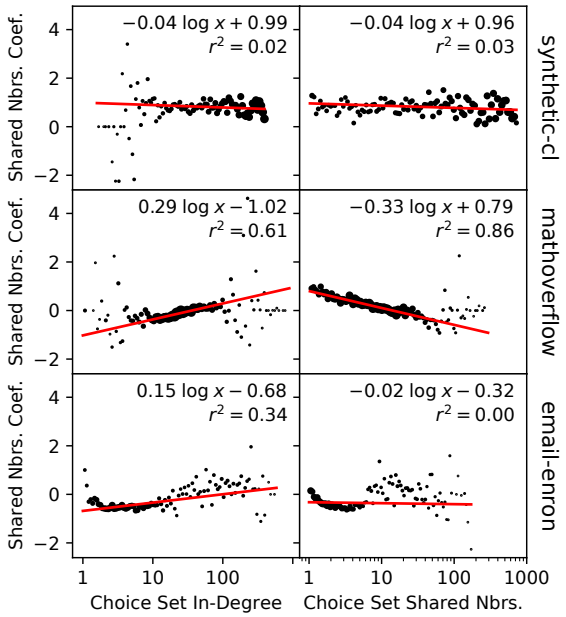

Figure 1: Learned preference coefficients of CLs trained on samples binned by mean choice set feature. Each point shows the preference coefficient of the shared neighbors feature in choice sets with varying mean in-degree (left column) and shared neighbor counts (right column). These coefficients were found by splitting observations into 100 bins according to their mean feature values and learning a CL for each bin separately. The area of each point is proportional to the square root of the number of observations in its bin. The red lines are weighted least squares fits.

bin the samples of each dataset according to the mean feature values in the choice set. We then fit CLs within each bin, examining whether the preference coefficients of features vary with the mean choice set features. Figure 1 shows two clear linear (with the respect to the log-transformed feature) context effects in MATHOVERFLOW: (1) as the mean in-degree of the choice set increases, so does the shared neighbors preference coefficient and (2) the shared neighbors coefficient decreases in choice sets with higher mean shared neighbors. Colloquially, (1) close ties are a stronger predictor of new connections when selecting between a set of popular individuals and (2) common connections matter less when choosing from a closely connected group. The different intercepts of these two effects in MATHOVERFLOW also motivate decomposing the LCL to the DLCL. The figure also shows some evidence of non-linear context effects in EMAIL-ENRON, which is of interest for future work.

Evaluating model fit. With our evidence that context effects are worth capturing, we compare our LCL and DLCL models to the traditional choice models they subsume (CL and mixed logit) with likelihood-ratio tests. To correct for multiple hypotheses, we use $p < 0.001$ as our significance threshold.

Table 3 shows the total NLL of every dataset under the four models, along with markers indicating the significant likelihood-ratio tests. In the empirical network datasets, all likelihood-ratio tests are significant (all with $p < 10^{-9}$), indicating that feature

Table 3: Dataset negative log-likelihoods. Bolded entries indicate the highest likelihood for a dataset.

	CL	LCL	Mixed logit	DLCL
DISTRICT	3313	**3130**	3258	3206
DISTRICT-SMART	3426	**3278***	3351	3303†
EXPEDIA	839505	837649*	839055	**837569**†
SUSHI	9821	9773*	9793	**9764**
CAR-A	1702	1694	1696	**1692**
CAR-B	1305	1295	1297	**1284**
CAR-ALT	7393	**6733***	7301	7011†
SYNTHETIC-CL	**210473**	210486	210503	210504
SYNTHETIC-LCL	140279	**137232***	139539	137937†
WIKI-TALK	99608	97748*	95761	**95134**†
REDDIT-HYPERLINK	135108	132880*	133766	**132473**†
BITCOIN-ALPHA	19675	19190*	19093	**18877**†
BITCOIN-OTC	26968	26101*	25768	**25348**†
SMS-A	8252	**8056***	8239	8154†
SMS-B	13153	**12823***	13147	12975†
SMS-C	4988	4880*	4928	**4871**†
EMAIL-ENRON	73015	70061*	71450	**69254**†
EMAIL-EU	53025	51822*	51988	**51431**†
EMAIL-W3C	11012	10677*	9898	**9758**†
FACEBOOK-WALL	118208	**116062***	117210	116328†
COLLEGE-MSG	14575	14120*	13849	**13712**†
MATHOVERFLOW	500537	479999*	440482	**435932**†

*Significant likelihood-ratio test vs. CL ($p < 0.001$)

†Significant likelihood-ratio test vs. mixed logit ($p < 0.001$)

context effects are occurring. In the general choice datasets, EXPEDIA ($p < 10^{-16}$), DISTRICT-SMART ($p < 10^{-16}$), SUSHI ($p = 1.6 \times 10^{-7}$), and CAR-ALT ($p < 10^{-16}$) have significant tests for the LCL.

Evaluating predictive power. The likelihood-ratio tests provide strong evidence for the presence of feature context effects. A related question is whether our methods improve out-of-sample predictions. To address this question, we measure the mean relative rank of the true selected item in the output ranking of each method. We define the *relative rank* of an item i to be its index when the choice set C is sorted in descending probability order (with ties resolved by taking the mean of all possible indices), divided by $|C|-1$. The mean relative rank is a measure of how good the model's predictions are, from 0 (best) to 1 (worst). We use this rather than mean reciprocal rank because the choice sets have variable sizes [18]. Figure 2 shows that LCL and DLCL make better predictions than CL and mixed logit across many datasets. In some cases, the improvement are quite large; for example, in BITCOIN-OTC, the mean relative rank is 24% better in LCL than in CL.

Interpreting learned models on general choice datasets. The previous analyses of model fit and predictive power indicate that linear context effects are indeed a significant factor. We now investigate what these effects are and show how our models can be interpreted to discover choice behaviors. We focus on the LCL because of its simpler structure and convex objective. For the general choice datasets, we select two datasets for detailed examination:

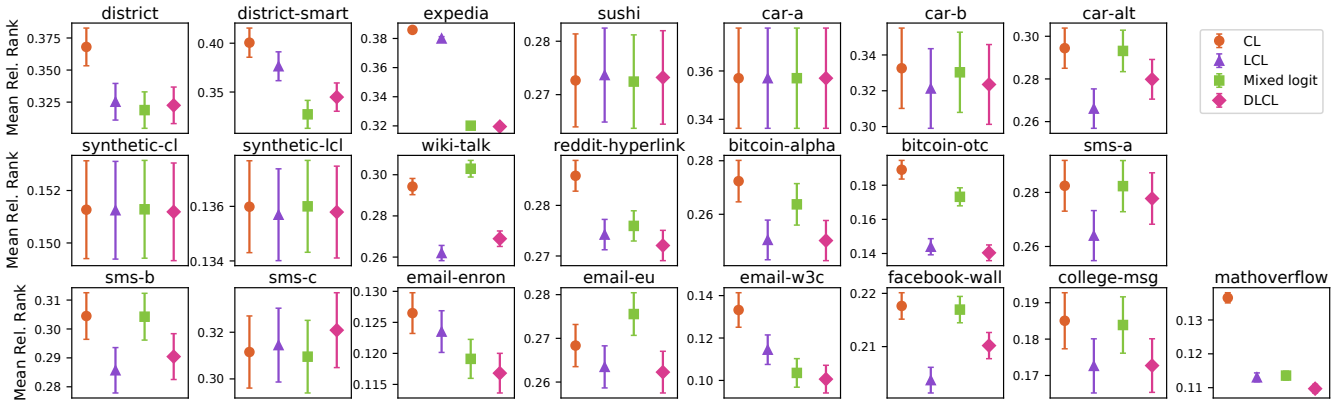

Figure 2: Mean relative rank of predictions on held-out test data (lower is better). Error bars show standard error of the mean.

Table 4: Five largest context effects in SUSHI.

Effect (q **on** p)	A_{pq} (std. err.)	p-value
popularity on *popularity*	−0.28 (0.15)	0.066
availability on *is maki*	0.24 (0.14)	0.087
oiliness on *oiliness*	−0.20 (0.08)	0.0089
popularity on *availability*	0.19 (0.14)	0.16
availability on *oiliness*	−0.18 (0.10)	0.064

Table 5: Five largest context effects in EXPEDIA.

Effect (q **on** p)	A_{pq} (std. err.)	p-value
location score on *price*	−0.47 (0.05)	$< 10^{-16}$
on promotion on *price*	0.27 (0.03)	$< 10^{-16}$
review score on *price*	−0.19 (0.03)	1.4×10^{-9}
star rating on *price*	0.15 (0.04)	6.7×10^{-5}
price on *star rating*	0.10 (0.00)	$< 10^{-16}$

EXPEDIA and SUSHI. The five context effects with largest magnitude in each dataset are shown in Tables 4 and 5. Note that features are all standardized, so picking the largest entries of A is meaningful.

Using the asymptotic normality of the maximum likelihood estimator [65], we can compute standard errors for the parameter estimates and p-values for the null hypothesis that a particular context effect is zero. This procedure is inexpensive: we need a single pass over the dataset after training to estimate the Fisher information matrix, from which standard errors can be directly computed (this is a standard procedure in statistical inference [65]).[2]

First, we examine SUSHI, which has randomly chosen choice sets. The most significant effect is that respondents given more oily sushi options showed more aversion to oily sushi (Table 4). The randomization of choice sets allows us to hypothesize that this is causal: too much oiliness on the menu makes oily foods less appealing, which could be an example of the similarity effect. The other context effects with largest magnitude in A are not significant.

In EXPEDIA, all five of the largest-magnitude effects are statistically significant (Table 5). The largest effect in the full model is a decrease in willingness to pay (i.e., cheaper options are more preferred) when the mean location score of the choice set is high. Additionally, if many of the options are marked as "on promotion," people seem more willing to book higher priced hotels. Interestingly, when the available hotels tend to be well-reviewed by other Expedia users, people are more price-averse, but they are less price-averse when the available hotels tend to have high star ratings. This may be because people searching for five-star hotels are not looking for the cheapest options, whereas people searching for well-reviewed hotels are looking for good deals. (The dataset does not include this information, only the location, length of stay, booking window, adult/children count, and room count of the search.) Finally, people choosing between more expensive hotels placed more weight on high star rating. When interpreting these effects, it is important to keep in mind that the choice sets in EXPEDIA may be influenced by user preferences to begin with, so we cannot determine whether the effects are causal. Nonetheless, the learned LCL model could motivate a randomized controlled trial aimed at determining causal effects. It also illustrates an important point to keep in mind when using choice data from recommender systems: choice sets are not necessarily independent from preferences.

Interpreting learned models on network growth datasets. We take a different approach to examine context effects in the network datasets, showcasing another useful application of the LCL. To visualize what context effects influence choice in the network datasets, we apply L_1 regularization of varying strength to the LCL matrix A during training, which encourages sparsity. Figure 3 visualizes the learned A matrices. Recall that a column of A corresponds to the feature exerting an effect and a row to the influenced feature.

Figure 3 reveals several effects shared by multiple datasets. For example, in MATHOVERFLOW, FACEBOOK-WALL, SMS-A, SMS-B, and REDDIT-HYPERLINK, feature 1 (in-degree) has a positive effect on feature 2's coefficient (shared neighbors). This suggests that close connections matter more when choosing from a popular group. And in EMAIL-ENRON and EMAIL-W3C, there is a negative effect of feature 6 (reciprocal recency) on feature 1 (in-degree): high-volume email recipients are less likely to be targeted when the sender's inbox has recent messages from other potential targets.

In both of these examples, when increasing regularization causes those entries of A to go to 0, we see a jump in the likelihood,

[2] Another useful (but more computationally expensive) approach is to constrain A to zero in all but one entry of interest. This preserves NLL convexity, still allows for likelihood ratio tests, and can be used to determine the effect size of a context effect.

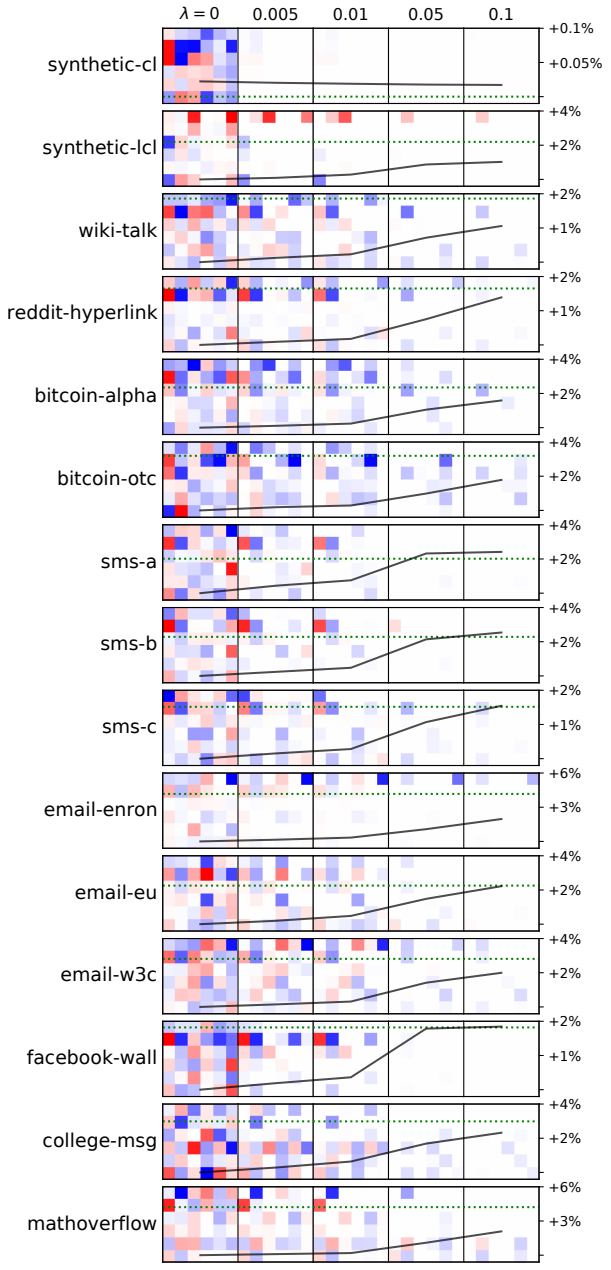

Figure 3: Effect of L_1 regularization on the LCL context effect matrix. The parameter λ (increasing left to right) controls the strength of regularization. Each box visualizes the learned matrix A (blue = negative, red = positive, white = zero; consistent color scales within but not between rows) at the given λ value. Features in A are in the order from Section 6.2, top-down and left-right. The black line tracks the total NLL of the LCL (the % on the y-axes is relative to the NLL of the best model plotted for that dataset). The dotted green line is the significance threshold of a likelihood-ratio test against a CL ($p < 0.001$; black line below the threshold means the LCL is a significantly better fit than CL).

indicating that these are important effects to capture (note that we plot NLL, so lower is better). Additionally, we see in the top row how a dataset with no context effects (SYNTHETIC-CL) behaves: A immediately goes to 0 when any L_1 regularization is applied, without any worsening of the likelihood.

7 DISCUSSION

Discovering intuitive context effects from choice data using our models has a number of potential applications. In recommender systems, insight into context effects could inform the set of options suggested to the user. Our models can also produce hypotheses for more controlled investigation in economics or psychology. A key contribution is showing how intuitive and general context effects can be automatically recovered from observed choices and tested for significance. While we focused on linear context effects for simplicity, some datasets (e.g., EMAIL-ENRON in Figure 1) show evidence of non-linearity. Capturing these more complex effects while retaining ease of training and interpretation would be valuable.

Our network analysis revealed several context effects in network growth, which can aid modeling within network science and social network analysis. We focused on triadic closure, where context effects can be observed in small choice sets. Incorporating context effects in other modes of network growth (such as connections with unrelated nodes) is an interesting avenue for future research. A challenge is that global modes of edge formation have large choice sets, requiring negative sampling for effective estimation [44], which seems difficult to adapt for models with context effects.

A limitation of our approach is that the generalizability of identified effects is constrained by correlations in the data. For example, choice sets arising from recommender systems (such as EXPEDIA) are correlated with the preferences of their users by design. This makes it difficult to distinguish between how a user's preferences are affected by the choice set and how the user's preferences influence the choice set. Our recent research adapts causal inference methods to the discrete choice setting to address this issue [58].

In other situations, we might have random choice sets (as in SUSHI) or we might have no information about how choice sets are determined. In the latter case, our approach could also be used to find evidence of choice sets targeted at chooser preferences: if we observe many positive self-effects (i.e., preference for star-rating is higher in sets with high star-rating), this could mean that choice sets are being catered to people's preferences. In some cases, this could be undesirable (e.g., if the party presenting individuals with options is supposed to be impartial), and our methods could provide a mechanism for identifying unwanted interventions.

Another challenging direction for future work would be a method of discovering more complex relational context effects from choice data. The feature context effects we study describe the influence of one feature on another, but some of the traditional context effects studied in economics and psychology (e.g., the compromise effect) are based on the relationship between the features of several items. These effects are typically studied with targeted models that are hand-crafted to capture the desired effect. A general method of encoding and learning relational context effects could enable the discovery of new complex effects not yet envisioned by choice theorists, but nonetheless appear in choice data.

ACKNOWLEDGMENTS

This research was supported by ARO MURI, ARO Award W911NF19-1-0057, NSF Award DMS-1830274, and JP Morgan Chase & Co. We thank Jan Overgoor and Johan Ugander for helpful conversations and Sophia Franco for naming suggestions.

REFERENCES

[1] Ehsan Abbasnejad, Scott Sanner, Edwin V Bonilla, and Pascal Poupart. 2013. Learning community-based preferences via dirichlet process mixtures of gaussian processes. In *IJCAI*.
[2] Simon P Anderson, Andre De Palma, and Jacques-Francois Thisse. 1992. *Discrete choice theory of product differentiation*. MIT press.
[3] Richard R Batsell and John C Polking. 1985. A new class of market share models. *Marketing Science* 4, 3 (1985), 177–198.
[4] Austin R Benson, Rediet Abebe, Michael T Schaub, Ali Jadbabaie, and Jon Kleinberg. 2018. Simplicial closure and higher-order link prediction. *PNAS* (2018).
[5] Austin R Benson and Jon Kleinberg. 2018. Found graph data and planted vertex covers. In *Neurips*.
[6] Austin R Benson, Ravi Kumar, and Andrew Tomkins. 2016. On the relevance of irrelevant alternatives. In *WWW*.
[7] Amanda Bower and Laura Balzano. 2020. Preference Modeling with Context-Dependent Salient Features. In *ICML*.
[8] David Brownstone, David S Bunch, Thomas F Golob, and Weiping Ren. 1996. A transactions choice model for forecasting demand for alternative-fuel vehicles. *Research in Transportation Economics* 4 (1996), 87–129.
[9] Elizabeth Bruch, Fred Feinberg, and Kee Yeun Lee. 2016. Extracting multistage screening rules from online dating activity data. *PNAS* (2016).
[10] Shuo Chen and Thorsten Joachims. 2016. Modeling intransitivity in matchup and comparison data. In *WSDM*.
[11] Shuo Chen and Thorsten Joachims. 2016. Predicting matchups and preferences in context. In *KDD*.
[12] Flavio Chierichetti, Ravi Kumar, and Andrew Tomkins. 2018. Learning a mixture of two multinomial logits. In *ICML*.
[13] Nick Craswell, Arjen P de Vries, and Ian Soboroff. 2005. Overview of the TREC 2005 Enterprise Track. In *TREC*, Vol. 5. 199–205.
[14] Jay K Dow and James W Endersby. 2004. Multinomial probit and multinomial logit: a comparison of choice models for voting research. *Elect Stud.* (2004).
[15] David Easley and Jon Kleinberg. 2010. *Networks, crowds, and markets*. CUP.
[16] Fred Feinberg et al. 2020. Choices in networks: a research framework. *Marketing Letters* (2020), 1–11.
[17] Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. 2001. *The elements of statistical learning*. Vol. 1. Springer Series in Statistics.
[18] Norbert Fuhr. 2018. Some common mistakes in IR evaluation, and how they can be avoided. In *ACM SIGIR Forum*, Vol. 51. 32–41.
[19] Mark S Granovetter. 1977. The strength of weak ties. In *Social Networks*.
[20] Bettina Grün and Friedrich Leisch. 2008. Identifiability of finite mixtures of multinomial logit models with varying and fixed effects. *J. Classif* (2008).
[21] Harsh Gupta and Mason A. Porter. 2020. Mixed Logit Models and Network Formation. arXiv:2006.16516 [cs.SI]
[22] Petter Holme and Beom Jun Kim. 2002. Growing scale-free networks with tunable clustering. *Physical Review E* 65, 2 (2002), 026107.
[23] Joel Huber, John W Payne, and Christopher Puto. 1982. Adding asymmetrically dominated alternatives: Violations of regularity and the similarity hypothesis. *Journal of Consumer Research* 9, 1 (1982), 90–98.
[24] Samuel Ieong, Nina Mishra, and Or Sheffet. 2012. Predicting preference flips in commerce search. In *ICML*.
[25] Matthew O Jackson and Brian W Rogers. 2007. Meeting strangers and friends of friends: How random are social networks? *Am. Econ. Rev* (2007).
[26] Emily M Jin, Michelle Girvan, and Mark EJ Newman. 2001. Structure of growing social networks. *Physical Review E* 64, 4 (2001), 046132.
[27] Kaggle. 2013. Personalize Expedia Hotel Searches - ICDM 2013. https://www.kaggle.com/c/expedia-personalized-sort.
[28] Toshihiro Kamishima. 2003. Nantonac collaborative filtering: recommendation based on order responses. In *KDD*.
[29] Aaron Kaufman, Gary King, and Mayya Komisarchik. 2017. How to measure legislative district compactness if you only know it when you see it. *American Journal of Political Science* (2017).
[30] Srijan Kumar et al. 2018. Rev2: Fraudulent user prediction in rating platforms. In *WSDM*.
[31] Srijan Kumar, William L Hamilton, Jure Leskovec, and Dan Jurafsky. 2018. Community interaction and conflict on the web. In *WebConf*.
[32] Srijan Kumar, Francesca Spezzano, VS Subrahmanian, and Christos Faloutsos. 2016. Edge weight prediction in weighted signed networks. In *ICDM*.
[33] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Predicting positive and negative links in online social networks. In *WWW*.
[34] Jure Leskovec, Daniel Huttenlocher, and Jon Kleinberg. 2010. Signed networks in social media. In *CHI*.
[35] Jure Leskovec, Jon Kleinberg, and Christos Faloutsos. 2007. Graph evolution: Densification and shrinking diameters. *TKDD* (2007).
[36] R Duncan Luce. 1959. *Individual choice behavior: A theoretical analysis*. Courier Corporation.
[37] Jacob Marschak. 1960. Binary choice constraints on random utility indicators. In *Stanford Symposium on Mathematical Methods in the Social Sciences*.
[38] Yusufcan Masatlioglu, Daisuke Nakajima, and Erkut Y Ozbay. 2012. Revealed attention. *American Economic Review* 102, 5 (2012), 2183–2205.
[39] Daniel McFadden. 1973. Conditional logit analysis of qualitative choice behavior. (1973).
[40] Daniel McFadden and Kenneth Train. 2000. Mixed MNL models for discrete response. *Journal of Applied Econometrics* 15, 5 (2000), 447–470.
[41] James Moody, Wendy D Brynildsen, D Wayne Osgood, Mark E Feinberg, and Scott Gest. 2011. Popularity trajectories and substance use in early adolescence. *Social Networks* 33, 2 (2011), 101–112.
[42] Paulo Natenzon. 2019. Random choice and learning. *J Polit Econ* (2019).
[43] Jan Overgoor, Austin Benson, and Johan Ugander. 2019. Choosing to grow a graph: Modeling network formation as discrete choice. In *WebConf*.
[44] Jan Overgoor, George Pakapol Supaniratisai, and Johan Ugander. 2020. Scaling Choice Models of Relational Social Data. In *KDD*.
[45] Pietro Panzarasa, Tore Opsahl, and Kathleen M Carley. 2009. Patterns and dynamics of users' behavior and interaction: Network analysis of an online community. *J. Assoc. Inf. Sci. Technol* (2009).
[46] Ashwin Paranjape, Austin R Benson, and Jure Leskovec. 2017. Motifs in temporal networks. In *WSDM*.
[47] Karlson Pfannschmidt, Pritha Gupta, and Eyke Hüllermeier. 2019. Learning Choice Functions: Concepts and Architectures. arXiv:1901.10860 [cs.LG]
[48] Robin L Plackett. 1975. The analysis of permutations. *J. R. Stat. Soc C* (1975).
[49] Drazen Prelec, Birger Wernerfelt, and Florian Zettelmeyer. 1997. The role of inference in context effects: Inferring what you want from what is available. *Journal of Consumer Research* 24, 1 (1997), 118–125.
[50] Robert P Rooderkerk, Harald J Van Heerde, and Tammo HA Bijmolt. 2011. Incorporating context effects into a choice model. *J Mark Res* (2011).
[51] Nir Rosenfeld, Kojin Oshiba, and Yaron Singer. 2020. Predicting Choice with Set-Dependent Aggregation. In *ICML*.
[52] William E Roth. 1934. On direct product matrices. *Bull. Am. Math. Soc* (1934).
[53] Francisco JR Ruiz, Susan Athey, and David M Blei. 2020. Shopper: A probabilistic model of consumer choice with substitutes and complements. *Ann. Appl. Stat* (2020).
[54] Arjun Seshadri, Alex Peysakhovich, and Johan Ugander. 2019. Discovering Context Effects from Raw Choice Data. In *ICML*.
[55] Eldar Shafir, Itamar Simonson, and Amos Tversky. 1993. Reason-based choice. *Cognition* 49, 1-2 (1993), 11–36.
[56] Itamar Simonson and Amos Tversky. 1992. Choice in context: Tradeoff contrast and extremeness aversion. *Journal of Marketing Research* 29, 3 (1992), 281–295.
[57] Kenneth A Small and Cheng Hsiao. 1985. Multinomial logit specification tests. *International Economic Review* (1985), 619–627.
[58] Kiran Tomlinson, Johan Ugander, and Austin R Benson. 2021. Choice Set Confounding in Discrete Choice. In *KDD*.
[59] Kenneth E Train. 2009. *Discrete choice methods with simulation*. CUP.
[60] Jennifer S Trueblood, Scott D Brown, Andrew Heathcote, and Jerome R Busemeyer. 2013. Not just for consumers: Context effects are fundamental to decision making. *Psychological Science* 24, 6 (2013), 901–908.
[61] Amos Tversky. 1972. Elimination by aspects: A theory of choice. *Psychol. Rev* (1972).
[62] Amos Tversky and Itamar Simonson. 1993. Context-dependent preferences. *Management Science* 39, 10 (1993), 1179–1189.
[63] Alexei Vázquez. 2003. Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations. *PRE* (2003).
[64] Bimal Viswanath, Alan Mislove, Meeyoung Cha, and Krishna P Gummadi. 2009. On the evolution of user interaction in Facebook. In *WOSN*.
[65] Larry Wasserman. 2013. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media.
[66] Ye Wu, Changsong Zhou, Jinghua Xiao, Jürgen Kurths, and Hans Joachim Schellnhuber. 2010. Evidence for a bimodal distribution in human communication. *PNAS* (2010).
[67] Shuang-Hong Yang, Bo Long, Alexander J Smola, Hongyuan Zha, and Zhaohui Zheng. 2011. Collaborative competitive filtering: learning recommender using context of user choice. In *SIGIR*.
[68] Hao Yin, Austin R Benson, Jure Leskovec, and David F Gleich. 2017. Local higher-order graph clustering. In *KDD*.
[69] Hao Yin, Austin R Benson, and Johan Ugander. 2020. Measuring directed triadic closure with closure coefficients. *Network Science* (2020).
[70] Zhibing Zhao, Peter Piech, and Lirong Xia. 2016. Learning mixtures of Plackett–Luce models. In *ICML*.

A PROOFS

The proof of Theorem 4.1 relies on three lemmas.

Lemma A.1 ([54], Appendix A). *For any choice set C, there is a bijection between the choice probabilities $\{\Pr(i,C) \mid i \in C\}$ and the log probability ratios $\{\beta_{i,C} \mid i \in C\}$ defined by*

$$\beta_{i,C} = \log\left(\frac{\Pr(i,C)}{\left[\prod_{j\in C}\Pr(j,C)\right]^{\frac{1}{|C|}}}\right). \tag{8}$$

Proof. We can compute $\beta_{i,C}$ given all choice probabilities in C as defined above. To obtain probabilities given log probability ratios, take

$$\begin{aligned}\frac{\exp(\beta_{i,C})}{\sum_{j\in C}\exp(\beta_{j,C})} &= \frac{\frac{\Pr(i,C)}{(\prod_{h\in C}\Pr(h,C))^{\frac{1}{|C|}}}}{\sum_{j\in C}\frac{\Pr(j,C)}{(\prod_{h\in C}\Pr(h,C))^{\frac{1}{|C|}}}}\\ &= \frac{\Pr(i,C)}{\sum_{j\in C}\Pr(j,C)}\\ &= \Pr(i,C).\end{aligned}$$

□

This means we can prove identifiability from the βs rather than from choice probabilities. We can also get a simple form for $\beta_{i,C}$ under the LCL.

Lemma A.2. *In the LCL, $\beta_{i,C} = (\theta + Ay_C)^T(y_i - y_C)$.*

Proof. Define $\theta_C = \theta + Ay_C$ for brevity.

$$\begin{aligned}\beta_{i,C} &= \log\left(\frac{\Pr(i,C)}{(\prod_{h\in C}\Pr(h,C))^{\frac{1}{|C|}}}\right)\\ &= \log\left(\frac{\exp\left(\theta_C^T y_i\right)}{\sum_{j\in C}\exp\left(\theta_C^T y_j\right)}\Bigg/\left(\prod_{h\in C}\frac{\exp\left(\theta_C^T y_h\right)}{\sum_{j\in C}\exp\left(\theta_C^T y_j\right)}\right)^{\frac{1}{|C|}}\right)\\ &= \log\left(\frac{\exp\left(\theta_C^T y_i\right)}{\left[\prod_{h\in C}\exp\left(\theta_C^T y_h\right)\right]^{\frac{1}{|C|}}}\right)\\ &= \theta_C^T y_i - \frac{1}{|C|}\sum_{h\in C}\theta_C^T y_j\\ &= \theta_C^T(y_i - y_C).\end{aligned}$$

□

Let $\mathrm{vec}(A)$ denote the vectorization of the matrix A (the vector formed by stacking the columns of A).

Lemma A.3 (Special case of the vec trick, [52]). *For any vectors $x \in \mathbb{R}^m, y \in \mathbb{R}^n$ and matrix $A \in \mathbb{R}^{m\times n}$, $x^TAy = (y \otimes x)^T\mathrm{vec}(A)$.*

Proof.

$$\begin{aligned}x^TAy = \sum_{i=1}^m\sum_{j=1}^n A_{ij}x_iy_j &= \sum_{j=1}^n y_j\sum_{i=1}^m A_{ij}x_i\\ &= \begin{bmatrix}y_1x\\ y_2x\\ \vdots\\ y_nx\end{bmatrix}^T\mathrm{vec}(A)\\ &= (y\otimes x)^T\mathrm{vec}(A).\end{aligned}$$

□

With these facts in hand, we are ready to prove Theorem 4.1.

Proof of Theorem 4.1. Consider the log probability ratio of an item i appearing in choice set C:

$$\begin{aligned}\beta_{i,C} &= (y_i - y_C)^T(\theta + Ay_C) && \text{(by Lemma A.2)}\\ &= (y_i - y_C)^T\begin{bmatrix}A & \theta\end{bmatrix}\begin{bmatrix}y_C\\ 1\end{bmatrix}\\ &= \left(\begin{bmatrix}y_C\\ 1\end{bmatrix}\otimes(y_i - y_C)\right)^T\mathrm{vec}\left(\begin{bmatrix}A & \theta\end{bmatrix}\right). && \text{(by Lemma A.3)}\end{aligned}$$

Let $m = |\{(i,C) \mid C \in C_{\mathcal{D}}, i \in C\}|$ be the number of distinct (item, choice set) pairs in the dataset. Index these pairs from 1 to m. We construct the following $m \times (d^2 + d)$ linear system by stacking all the $\beta_{i,C}$ equations:

$$\begin{bmatrix}\left(\begin{bmatrix}y_{C_1}\\ 1\end{bmatrix}\otimes(y_{i_1} - y_{C_1})\right)^T\\ \vdots\\ \left(\begin{bmatrix}y_{C_m}\\ 1\end{bmatrix}\otimes(y_{i_m} - y_{C_m})\right)^T\end{bmatrix}\mathrm{vec}\left(\begin{bmatrix}A & \theta\end{bmatrix}\right) = \begin{bmatrix}\beta_{i_1,C_1}\\ \vdots\\ \beta_{i_m,C_m}\end{bmatrix}.$$

Supposing the choice probabilities are generated according to the LCL, this system is consistent (although it is highly overdetermined with a large dataset). Any solution to this system is a setting of the parameters θ, A that results in the observed log probability ratios (and therefore choice probabilities, by Lemma A.1). Since we know the system is consistent, it has a unique solution (i.e., the LCL is identifiable) if and only if the rows of the matrix span $\mathbb{R}^{d^2+d}$. □

Proof of Proposition 4.2. Suppose that $y_{C_1}, \ldots, y_{C_k}$ ($k < d + 1$) is a maximal set of affinely independent mean feature vectors appearing in the dataset $\mathcal{D}$. In each one of these choice sets C_i, the choice probabilities are determined by $\theta_{C_i} = \theta + Ay_{C_i}$. However, since $k < d + 1$, there are infinitely many affine transformations $\theta + Ay_{C_i}$ that map every y_{C_i} to its corresponding θ_{C_i}. For any other choice set $C' \notin \{C_1, \ldots, C_k\}$, we can express its mean feature vector as an affine combination $y_{C'} = \sum_{i=1}^k \alpha_i y_{C_i}$, where $\sum_{i=1}^k \alpha_i = 1$. We then have $\theta_{C'} = \theta + A(\sum_{i=1}^k \alpha_i y_{C_i}) = \sum_{i=1}^k \alpha_i(\theta + Ay_{C_i}) = \sum_{i=1}^k \alpha_i\theta_{C_i}$, so any of the infinitely many affine transformations that correctly map y_{C_i} to θ_{C_i} will also map $y_{C'}$ to $\theta_{C'}$. This means there are infinitely many parameter settings θ and A that would result in the same choice probabilities, so the LCL is not identifiable. □

Proof of Proposition 4.3. We will use differences in log probability ratios to first identify the choice set dependent utilities

Algorithm 1 EM algorithm for estimating DLCL parameters.

Input: m observations $\mathcal{D}$, d features
$A^{(0)}, B^{(0)} \leftarrow d \times d$ randomly initialized matrices
$\pi^{(0)} \leftarrow d$-dimensional vector with all entries equal to $\frac{1}{d}$
$t \leftarrow 0$
while not converged **do**
  $p_{hk} \leftarrow \frac{\exp\left([B_k^{(t)} + A_k^{(t)}(y_C)_k]^T y_i\right)}{\sum_{j \in C} \exp\left([B_k^{(t)} + A_k^{(t)}(y_C)_k]^T y_j\right)}$
    for each $(i, C) = \mathcal{D}_h$ and $k = 1, \ldots, d$
  $r_{hk} \leftarrow \frac{\pi_k^{(t)} p_{hk}}{\sum_{g=1}^{d} \pi_g^{(t)} p_{hg}}$ for each $h = 1, \ldots, m$ and $k = 1, \ldots, d$
  $Q(A, B \mid \theta^{(t)}) \leftarrow \sum_{(i,C)=\mathcal{D}_h} \sum_{k=1}^{d} r_{hk} \Big[[B_k + A_k(y_C)_k]^T y_i - \log \sum_{j \in C} \exp\left([B_k + A_k(y_C)_k]^T y_j\right) \Big]$
  Find a minimizer A^*, B^* of $-Q(A, B \mid \theta^{(t)})$ using gradient descent
  $A^{(t+1)} \leftarrow A^*$, $B^{(t+1)} \leftarrow B^*$
  $\pi_k^{(t+1)} \leftarrow \frac{1}{|\mathcal{D}|} \sum_{h=1}^{|\mathcal{D}|} r_{hk}$ for each $k = 1, \ldots, d$
  $t \leftarrow t + 1$
return $A^{(t)}, B^{(t)}, \pi^{(t)}$

$\theta_C = \theta + A y_C$ in each choice set and then combine those to determine θ and A.

To remove a dependence on mean feature vectors, consider the difference of two log probability ratios in the same choice set:

$$\begin{aligned} \beta_{i_1,C} - \beta_{i_2,C} &= \theta_C^T (y_{i_1} - y_C) - \theta_C^T (y_{i_2} - y_C) \qquad \text{(by Lemma A.2)} \\ &= \theta_C^T (y_{i_1} - y_{i_2}). \end{aligned}$$

In order to identify the vector θ_C, form the following linear system from d such differences, all in the same choice set C:

$$\begin{bmatrix} (y_{i_1} - y_{i_0})^T \\ (y_{i_2} - y_{i_0})^T \\ \vdots \\ (y_{i_d} - y_{i_0})^T \end{bmatrix} \theta_C = \begin{bmatrix} \beta_{i_1,C} - \beta_{i_0,C} \\ \beta_{i_2,C} - \beta_{i_0,C} \\ \vdots \\ \beta_{i_d,C} - \beta_{i_0,C} \end{bmatrix}$$

If the rows of the matrix are linearly independent, then we can uniquely solve this system to find θ_C. For this to be the case, we need the $d+1$ feature vectors $y_{i_0}, \ldots, y_{i_d}$ to be affinely independent.

In order to recover θ and A, we need to solve the affine system $\theta + A y_C = \theta_C$ for θ and A given observations of y_C and θ_C. Affine transformations in d dimensions are uniquely specified by their action on a set of $d+1$ affinely independent vectors. So, if we have $d+1$ observed choice sets $C_0, \ldots, C_d$ whose mean feature vectors $y_{C_0}, \ldots, y_{C_d}$ are affinely independent (and if we know $\theta_{C_0}, \ldots, \theta_{C_d}$), then we can uniquely identify θ and A. As we have seen, we can find $\theta_{C_0}, \ldots, \theta_{C_d}$ if each of $C_0, \ldots, C_d$ has $d+1$ items with affinely independent feature vectors. □

B EM ALGORITHM FOR DLCL ESTIMATION

Let $\mathcal{D}_h$ denote hth observation (i, C) and $\Delta_h \in \{1, \ldots, d\}$ denote the latent mixture component that the observation $\mathcal{D}_h$ comes from (taking the view that each observation belongs to one component).

The EM algorithm (see [17] for a general treatment) is an iterative procedure that begins with initial guesses for the parameters $\theta^{(0)} = (A^{(0)}, B^{(0)}, \pi^{(0)})$ and updates them until convergence. In the update step, we maximize the expectation of the log-likelihood $\ell(A, B; \mathcal{D}, \Delta)$ over the distribution of the unobserved variable Δ conditioned on the observations $\mathcal{D}$ and the current estimates of the parameters, denoted $\mathrm{E}_\Delta[\ell(A, B; \mathcal{D}, \Delta) \mid \mathcal{D}, \theta^{(t)}]$. The new estimates $A^{(t+1)}$ and $B^{(t+1)}$ are the maximizers of this function. The new estimate of the mixture proportions $\pi^{(t+1)}$ has a closed form based on the probability that each observation comes from each mixture component according to the current estimates of A and B. See Algorithm 1 for the complete procedure. We derive the details here, starting with a breakdown of the expectation function:

$$\begin{aligned} &\mathrm{E}_\Delta[\ell(A, B; \mathcal{D}, \Delta) \mid \mathcal{D}, \theta^{(t)}] \\ &\quad = \sum_{(i,C)=\mathcal{D}_h} \sum_{k=1}^{d} \Pr(\Delta_h = k \mid i, C, \theta^{(t)}) \log \Pr(i, C \mid \Delta_h = k, A, B). \end{aligned} \tag{9}$$

We can compute the first part of the summand (the *responsibilities*) using Bayes' Theorem:

$$\Pr(\Delta_h = k \mid i, C, \theta^{(t)}) = \frac{\Pr(i, C \mid \Delta_h = k, \theta^{(t)}) \Pr(\Delta_h = k \mid \theta^{(t)})}{\Pr(i, C \mid \theta^{(t)})} \tag{10}$$

$$= \pi_h^{(t)} \frac{\Pr(i, C \mid \Delta_h = k, \theta^{(t)})}{\Pr(i, C \mid \theta^{(t)})}. \tag{11}$$

The numerator of Equation (11) is the kth component of the DLCL choice probability (with our estimates for A and B):

$$\Pr(i, C \mid \Delta_h = k, \theta^{(t)}) = \frac{\exp\left([B_k^{(t)} + A_k^{(t)}(y_C)_k]^T y_i\right)}{\sum_{j \in C} \exp\left([B_k^{(t)} + A_k^{(t)}(y_C)_k]^T y_j\right)}. \tag{12}$$

Meanwhile, the denominator of Equation (11) is the sum of these probabilities weighted by the mixture weight estimates:

$$\Pr(i, C \mid \theta^{(t)}) = \sum_{k=1}^{d} \pi_k^{(t)} \Pr(i, C \mid \Delta_h = k, \theta^{(t)}). \tag{13}$$

The last term in Equation (9) is a function of the parameters A, B (not their estimates):

$$\log \Pr(i, C \mid \Delta_h = k, A, B) = \log \left[\frac{\exp\left([B_k + A_k(y_C)_k]^T y_i\right)}{\sum_{j \in C} \exp\left([B_k + A_k(y_C)_k]^T y_j\right)} \right] \tag{14}$$

$$= [B_k + A_k(y_C)_k]^T y_i - \log \sum_{j \in C} \exp\left([B_k + A_k(y_C)_k]^T y_j\right). \tag{15}$$

Equation (15) is concave by the same reasoning that the LCL's NLL (Equation (7)) is convex. Thus, the expectation $\mathrm{E}_\Delta[\ell(A, B; \mathcal{D}, \Delta) \mid \mathcal{D}, \theta^{(t)}]$, being the sum of positively scaled concave functions, is also concave. Its gradient is also Lipschitz continuous, just like the LCL's NLL. We can therefore find a global maximum using gradient ascent (in practice, we use gradient descent to minimize $-Q(A, B \mid \theta^{(t)})$).