Diverse and Experienced Group Discovery via Hypergraph Clustering

Ilya Amburg* Nate Veldt† Austin R. Benson‡

Abstract

In forming teams or groups, one often aims to balance expertise in a main focus area while also encouraging diversity of skills in each team. In this paper we model the problem of finding diverse groups of individuals who have expertise in a given task as a clustering problem on hypergraphs with heterogeneous edge types. Here, the hyperedge types encode past experience types of groups, and the output of the clustering is groups of individuals (nodes). Unlike complementary problems that seek to find fair or balanced clusters (e.g., in terms of some protected node attributes), our model encourages diversity of past *experience* within these groups by striking a balance between experience and diversity with respect to node participation in edge types. We show that naive objectives lead to no diversity-experience tradeoff, which motivates our refined model based on regularizing an edge-based hypergraph clustering objective. While optimizing our objective is NP-hard, we design a 2-approximation algorithm that works for a more general class of problems where each node is allowed to have a preference for a particular cluster, and illustrate a technique for computing regularization strength bounds that reveal meaningful diversity/experience tradeoff regimes. We illustrate the utility of our framework on several real-life datasets – most notably to online review platform data – to curate sets of reviews for a given type of product which exhibit a tradeoff between reviewer experience, or familiarity with a product type, and experience, or the reviewer's tendency to also review related product types. In the setting allowing for node preferences, we show that our framework discovers sets of reviews sensitive to user preference.

1 Introduction

Team formation within social and organizational contexts is ubiquitous, as success often relies on forming the "right" teams. Diversity within these teams, both with respect to socioeconomic attributes and expertise across disciplines, often leads to synergy, and brings fresh perspectives which facilitate innovation. The study of diverse team formation with respect to expertise has a rich history spanning decades of work in sociology, psychology and business management [12, 15, 19]. In this paper, we explore algorithmic approaches to diverse team formation, where "diversity" corresponds to a tendency of individuals to have a variety of experiences. In particular, we present a new algorithmic framework that focuses on forming groups which are *diverse* and *experienced* in terms of past group interactions. As a motivating example, consider a diverse team formation task in which the goal is to assign a task to a group of people who (1) already have some level of experience working together on the given task, and (2) are diverse in terms of their previous work experience. As another example, a recommender system may want to display a diverse yet cohesive set of reviews for a class of products.

Here, we formalize diverse and experienced group formation as a clustering problem on edge-labeled hypergraphs. In this setup, a hyperedge represents a set of individuals that have participated in a group interaction or experience. The hyperedge label encodes the *type* or *category* of interaction (e.g., a type of team project). The output is then a clustering of nodes, with cluster labels corresponding to hyperedge types. The goal is to form clusters whose nodes are balanced in terms of *experience* and *diversity*. By *experience* we mean that a cluster with label ℓ should contain nodes that have previously participated in hyperedges of type ℓ. By *diversity*, we mean that clusters should also include nodes that have participated in other hyperedge types.

Our mathematical framework for diverse and experienced clustering builds on an existing objective for clustering edge-labeled hypergraphs [4]. This objective encourages cluster formation in such a way that hyperedges of a certain label tend to be contained in a cluster with the same label We add a diversity-encouraging regularization term governed by a tunable hyperparameter $\beta \geq 0$ to this objective encouraging clusters to contain nodes that have participated in many different hyperedge types. Although the resulting objective is NP-hard in general, we design an LP algorithm that guarantees a 2-approximation for any β. We show that certain values of β reduce to extremal solutions of the diversity-regularized objective with closed-form solutions where just diversity or just experience is maximized. In order to guide a meaningful hyperparameter selection, we show how to bound the region in which non-extremal solutions occur by leveraging LP sensitivity techniques. Furthermore, we show that our approximation actually applies to a larger class of optimization problems, where each node has a preference distribution for cluster assignments. The diversity regularization framework is then the special case where a node's preference for a cluster is inversely related to its past experience for participating in that cluster.

We demonstrate the utility of our framework by ap-

[*] Cornell University ia244@cornell.edu
[†] Texas A&M University nveldt@tamu.edu
[‡] Cornell University arb@cs.cornell.edu

plying it to team formation of users posting answers on Stack Overflow, and the task of aggregating a diverse set of reviews for categories of establishments and products on review sites (e.g., Yelp or Amazon). We find that the framework yields meaningfully more diverse clusters at a small cost, and that our approximation algorithms produce solutions within a factor of no more than 1.3 of optimality empirically. A second set of experiments examines the effect of iteratively applying the diversity-regularized objective while keeping track of the experience history of every individual. We observe in this synthetic setup that regularization greatly influences team formation dynamics over time, as increasing β leads to more frequent role swapping.

1.1 Related work Our work on diversity in clustering is partly related to recent research on algorithmic fairness and fair clustering. These results are based on ideas that machine learning algorithms may make decisions that are biased or unfair towards a subset of a population [5, 10, 11]. There are now a variety of algorithmic fairness techniques to combat this issue [16, 22]. For clustering problems, fairness is typically formulated in terms of protected attributes on data points — a cluster is "fair" if it exhibits a proper balance between nodes from different protected classes, and the goal is to optimize clustering objectives while adhering to balance constraints on the protected attributes [3, 2, 9]. These approaches are similar to our notion of diverse clustering; in both cases, the clusters are more heterogeneous with respect to node attributes. While the primary attribute addressed in fair clustering is the protected status of a data point, in our case it is the "experience" of that point. In this sense, we have similar algorithmic goals, but our approach targets discovering diverse groups with respect to past experience.

There is also research on algorithmic diverse team formation [17, 26]. However, this research largely focuses diversity with respect to inherent node-level attributes, without an emphasis on diversity of expertise; our work is the first to explicitly address this issue.

Our framework also facilitates a novel take on diversity within recommender systems. An application we study in Section 4 is selecting expert, yet diverse sets of reviews for product categories. This differs from existing recommendation paradigms on two fronts: First, the literature focuses on user-centric recommendations; for us, a set of reviews is curated for a *category* of products that allows any user to glean both expert and diverse opinions regarding it. Further, recommender systems research has defined diversity for a set of objects based on dissimilarity derived from pairwise relations [7, 21]. There are some set-proxies for diverse recommendations [8, 25], but they do not deal explicitly with higher-order interactions among objects. In contrast, our work encourages diversity in recommendations through an objective that captures higher-order information about relations between subsets of objects.

2 Clustering with Diversity and Experience

After introducing notation for edge-labeled clustering, we analyze a seemingly natural approach for clustering based on experience and diversity that leads to only trivial solutions. This motivates us to develop a more meaningful objective through regularization of the categorical edge clustering objective, to which the rest of the paper is devoted.

Notation. Let $G = (V, E, L, \ell)$ be a hypergraph with labeled edges, where V is the set of nodes, E is the set of (hyper)edges, L is a set of edge labels, and $\ell \colon E \to L$ maps edges to labels, where $L = \{1, \dots, k\}$ and k is the number of labels. Furthermore, let $E_c \subseteq E$ be the edges with label c, and r the largesr hyperedge size. Following graph-theoretic terminology, we often refer to elements in L as "colors"; in data, L represents categories or types. For any node $v \in V$, let d_v^c be the number of hyperedges of color c in which node v appears. We refer to d_v^c as the color degree of v for color c.

We seek a clustering $\mathcal{C}$, where each node is assigned to exactly one cluster, and there is exactly one cluster for each color in L, so that it outputs a color for each node. We use $\mathcal{C}(i)$ to denote the nodes assigned to color i. A target clustering promotes both diversity (clusters have nodes from a range of colored hyperedges), and experience (for all $i \in L$, $\mathcal{C}(i)$ contains nodes that have experience participating in hyperedges of color i).

2.1 A flawed but illustrative first approach We start with an illustrative clustering objective that will prove to be useful in the rest of the paper. For this, we first define *diversity and experience scores* for a color i, denoted $D(i)$ and $E(i)$, as follows: $D(i) = \sum_{v \in \mathcal{C}(i), c \neq i} d_v^c$, $\quad E(i) = \sum_{v \in \mathcal{C}(i)} d_v^i$. In words, $D(i)$ measures how much nodes in cluster i have participated in hyperedges that are *not* color i, and $E(i)$ measures how much nodes in cluster i have participated in hyperedges of color i. A seemingly natural but ultimately naive objective for balancing experience and diversity is:

$$\max_{\mathcal{C}} \sum_{i \in L} [E(i) + \beta D(i)]. \tag{2.1}$$

The regularization parameter β determines the relative importance of the diversity and experience scores. It turns out that the solutions to this objective are overly-simplistic, with a phase transition at $\beta = 1$. We define two simple types of clusterings as follows:

- *Majority vote clustering*: Node v is placed in cluster $\mathcal{C}(i)$ where $i \in \operatorname{argmax}_{c \in L} d_v^c$, i.e., node v is placed in a cluster for which it has the most experience.

- *Minority vote clustering*: Node v is placed in cluster $\mathcal{C}(i)$ where $i \in \operatorname{argmin}_{c \in L} d_v^c$, i.e., node v is placed in a cluster for which it has the least experience.

The following theorem explains why (2.1) does not provide a meaningful tradeoff between diversity and experience.

Theorem 2.1. *A majority vote clustering optimizes (2.1) for all $\beta > 1$, and a minority vote clustering optimizes the same objective for all $\beta < 1$. Both are optimal when $\beta = 1$.*

Proof. Assume w.l.o.g. that colors $1, 2, \dots, k$ are ordered so that $d_i^1 \geq \cdots \geq d_i^k$ for node i. Clustering i to color 1 adds $d_i^1 + \beta \sum_{j=2}^{k} d_i^j$ to the objective, while clustering it to color $c \neq i$ adds $d_i^c + \beta \sum_{j \neq c} d_i^j$. Since $d_i^1 \geq \cdots \geq d_i^k$, the first contribution is greater than or equal to the second if and only if $\beta \leq 1$. Hence, majority vote is optimal when $\beta \geq 1$. A similar argument proves optimality for minority vote when $\beta \leq 1$. □

Objective (2.1) is easy to analyze, but has optimal points that do not provide a balance between diversity and experience. This occurs because a clustering will maximize the total diversity $\sum_{c \in L} D(c)$ if and only if it minimizes the total experience $\sum_{c \in L} E(c)$, as these terms sum to a constant. The following observation formalizes this.

Observation 2.1. *$\sum_{c \in L}[E(c) + D(c)]$ is a constant independent of the clustering $\mathcal{C}$.*

We will use this observation when developing our clustering framework in the next section.

2.2 Diversity-regularized categorical edge clustering

We now turn to a more sophisticated approach: a regularized version of the *categorical edge clustering* objective [4]. For a clustering $\mathcal{C}$, the objective accumulates a penalty of 1 for each hyperedge of color c that is not completely contained in the cluster $\mathcal{C}(c)$. More formally, the objective is:

$$\min_{\mathcal{C}} \sum_{c \in L} \sum_{e \in E_c} x_e, \tag{2.2}$$

where x_e is 1 if hyperedge $e \in E_c$ is *not* contained in cluster $\mathcal{C}(c)$, but is zero otherwise. This penalty encourages *entire* hyperedges to be contained inside clusters of the corresponding color. For our context, this objective can be interpreted as promoting *group* experience in cluster formation: if a group of people have participated together in task c, this is an indication they could work well together on task c in the future. However, we want to avoid the scenario where groups of people endlessly work on the same type of task without the benefiting from the perspective of others with different experiences. Therefore, we regularize objective (2.2) with a penalty term $\beta \sum_{c \in L} E(c)$. Since $\sum_{c \in L}[E(c) + D(c)]$ is a constant (Observation 2.1), this regularization encourages higher diversity scores $D(c)$ for each cluster $\mathcal{C}(c)$.

While the "all-or-nothing" penalty in (2.2) may seem restrictive at first, it is a natural choice for our objective function for several reasons. First, we are building on recent research showing applications of Objective (2.2) on datasets similar to ours, namely edge-labeled hypergraphs [4], and this type of penalty is a standard in hypergraph partitioning [6, 14, 18]. Second, if we consider an alternative penalty which incurs a cost of one for every node that is split away from the color of the hyperedge, this reduces to the "flawed first approach" in the previous section, where there is no diversity-experience tradeoff. Developing algorithms that can optimize more complicated alternative hyperedge cut penalties is an active area of research [20, 27]. Translating these ideas to our setting constitutes an interesting open direction for future work, but here we focus on the standard hyperedge cut penalty. Our experimental results indicate that this approach produces meaningfully diverse clusters on real-world and synthetic data.

We now formalize our objective, which we call *diversity-regularized categorical edge clustering* (DRCEC), that will be the focus for the remainder of the paper. We state it as an integer linear program (ILP):

$$\begin{array}{lll} (2.3) & & \\ \min & \sum_{c \in L} \sum_{e \in E_c} x_e + \beta \sum_{v \in V} \sum_{c \in L} d_v^c (1 - x_v^c) & \\ \text{s.t.} & \text{for all } v \in V\text{: } \sum_{c=1}^{k} x_v^c = k - 1, & \\ & \text{for all } c \in L,\ e \in E_c\text{: } x_v^c \leq x_e \text{ for all } v \in e; & \\ & \text{for all } c \in L,\ v \in V,\ e \in E\text{: } x_v^c, x_e \in \{0, 1\}. & \end{array}$$

The binary variable x_v^c equals 1 if node v is not assigned label c, and is 0 otherwise. The first constraint guarantees every node is assigned to exactly one color, while the second constraint guarantees that if a single node $v \in e$ is not assigned to the cluster of the color of e, then $x_e = 1$.

A polynomial-time 2-approximation algorithm. Optimizing the case of $\beta = 0$ is NP-hard [4], so DRCEC is also NP-hard. Although the optimal solution to (2.3) may vary with β, we develop a simple algorithm based on solving an LP relaxation of the ILP that rounds to a 2-approximation for every value of β. Our LP relaxation of the ILP in (2.3) replaces the binary constraints $x_v^c, x_e \in \{0, 1\}$ with linear constraints $x_v^c, x_e \in [0, 1]$. The LP can be solved in polynomial time, and the objective score is a lower bound on the optimal solution score to the NP-hard ILP. The values of x_v^c can then be *rounded* into integer solutions to produce a clustering that is within a bounded factor of the LP lower bound, and therefore within a bounded factor of optimality. Our algorithm is simply stated:

```
Algorithm 1
1. Solve the LP relaxation of the ILP in (2.3).
2. For each $v \in V$, assign $v$ to any $c \in \text{argmin}_j x_v^j$.
```

The LP relaxation gives a 2-approximation:

Theorem 2.2. *For any $\beta \geq 0$, Algorithm 1 returns a 2-approximation for Objective (2.3).*

Proof. Let the relaxed solution be $\{x_e^*, x_v^{*c}\}_{e \in E, v \in V, c \in L}$ and the rounded solution be $\{x_e, x_v^c\}_{e \in E, v \in V, c \in L}$. Let $y_v^c = 1 - x_v^c$ and $y_v^{*c} = 1 - x_v^{*c}$. Our objective evaluated at the relaxed and rounded solutions respectively is

$$S^* = \sum_e x_e^* + \beta \sum_{v\in V}\sum_{c\in L} d_v^c y_v^{*c},\ S = \sum_e x_e + \beta \sum_{v\in V}\sum_{c\in L} d_v^c y_v^c.$$

We will show that $S \le 2S^*$ by comparing the first and second terms of S and S^* respectively. The first constraint in (2.3) ensures that $x_v^c < 1/2$ for at most a single color c. Thus, for every edge e with $x_e = 1$, $x_v^{*c} \ge 1/2$ for some $v \in e$. In turn, $x_e^* \ge 1/2$, so $x_e \le 2x_e^*$. If $x_e = 0$, then $x_e \le 2x_e^*$ holds trivially. Thus, $\sum_e x_e \le 2\sum_e x_e^*$. Similarly, since $x_v^c = 1$ ($y_v^c = 0$) if and only if $x_v^{*c} \ge 1/2$ ($y_v^{c*} \le 1/2$), and $x_v^c = 0$ otherwise, it follows that $y_v^c \le 2y_v^{*c}$. Thus, $\sum_{v\in V}\sum_{c\in L} d_v^c y_v^c \le 2\sum_{v\in V}\sum_{c\in L} d_v^c y_v^{*c}$. □

2.3 A general preference-regularized objective

In fact, Algorithm 1 offers a 2-approximation for a much larger class of *preference-regularized* categorical edge clustering (PRCEC) objectives. This happens because Objective (2.3) still admits a 2-approximation via Algorithm 1 if we replace the color degree distribution $[d_v^1, \dots, d_v^k]$ of node v with an arbitrary *preference* distribution $[p_v^1, \dots, p_v^k]$, where each non-negative component p_v^c represents node v's *reluctance* to be in cluster c. Formally, the PRCEC objective is

$$\min \textstyle\sum_{c\in L}\sum_{e\in E_c} x_e + \beta \sum_{v\in V}\sum_{c\in L} p_v^c(1 - x_v^c) \tag{2.4}$$

with the same constraints as in Objective (2.3). We can prove that we obtain a 2-approximation for the PRCEC objective by replacing d_v^c with p_v^c throughout the proof of Theorem 2.2. This generalized result opens the door to a host of other applications, such as forming experienced teams while at the same time attempting to satisfy team assignment preferences.

2.4 Extremal LP and ILP solutions at large enough values of β

In general, Objective (2.3) provides a meaningful way to balance group experience (the first term) and diversity (the regularization). However, when $\beta \to \infty$, the objective corresponds to simply minimizing experience, (i.e., maximizing diversity), which is solved via the minority vote assignment. We formally show that the optimal integral solution (2.3), as well as the relaxed LP solution under certain conditions, transitions from standard behavior to extremal behavior (specifically, the minority vote assignment) when β increases past the maximum degree in the hypergraph. In Section 3, we show how to bound these transition points numerically, to ensure meaningful solutions.

We first consider a bound on β above which minority vote is optimal. Let d_{max} be the largest number of edges any node participates in.

Theorem 2.3. *For every $\beta > d_{max}$, a minority vote assignment optimizes* (2.3).

Proof. Let $\{x_e, x_v^c\}$ encode a clustering for (2.3) that is not a minority vote solution. This means there exists at least one node v so that $x_v^c = 0$ for some color $c \notin \operatorname{argmin}_{i\in L} d_v^i$. If we move node v from cluster c to some cluster $m \in \operatorname{argmin}_{i\in L} d_v^i$, then the regularization term would decrease by $\beta(d_v^c - d_v^m) \ge \beta > d_{max}$, since degrees are integer-valued and $d_v^c > d_v^m$. Meanwhile, the first term would increase by at most $\sum_{e:v\in e} x_e = d_{max} < \beta$. So deviating from the minority vote assignment cannot be optimal when $\beta > d_{max}$. □

A slight variant of this result also holds for the LP relaxation. For a node $v \in V$, let $\mathcal{M}_v \subset L$ be the set of minority vote clusters for v, i.e., $\mathcal{M}_v = \operatorname{argmin}_{c\in L} d_v^c$ (treating argmin as a set). The next theorem says that for $\beta > d_{max}$, the LP places all "weight" for v on its minority vote clusters. We call this a *relaxed minority vote LP solution*, and Algorithm 1 will round the LP relaxation to a minority vote clustering.

Theorem 2.4. *For every $\beta > d_{max}$, an optimal solution to the LP relaxation of* (2.3) *will satisfy $\sum_{c\in\mathcal{M}_v}(1 - x_v^c) = 1$ for every $v \in V$. So the rounded solution from Algorithm 1 is a minority vote clustering.*

Proof. Let $\{x_e, x_v^c\}$ encode an arbitrary solution to the *LP relaxation* of (2.3), and assume that it is *not* a minority vote solution. For every $v \in V$ and $c \in L$, let $y_v^c = 1 - x_v^c$. The y_v^c indicates the "weight" of v placed on cluster c, with $\sum_{c\in L} y_v^c = 1$. Since $\{x_e, x_v^c\}$ is not a minority vote solution, there exists some $v \in V$ and $j \notin \mathcal{M}_v$ such that $y_v^j = \varepsilon > 0$.

We will show that when $\beta > d_{max}$, we obtain a strictly better solution by moving this weight of ε from cluster j to a cluster in $\mathcal{M}_v$. Choose any $m \in \mathcal{M}_v$, and define a new set of variables $\hat{y}_v^j = 0$, $\hat{y}_v^m = y_v^m + \varepsilon$, and $\hat{y}_v^i = y_v^i$ for all other $i \notin \{m, j\}$. Define $\hat{x}_v^c = 1 - \hat{y}_v^c$ for all $c \in L$. For any $u \in V$, $u \ne v$, we keep variables the same: $\hat{y}_u^c = y_u^c$ for all $c \in L$. Set edge variables $\hat{x}_e$ to minimize the LP objective subject to the $\hat{y}_c$ variables, i.e., for $c \in L$ and every $e \in E_c$, let $\hat{x}_e = \max_{u\in e} \hat{x}_u^c$.

The new variables take ε weight from cluster j and move it to $m \in \mathcal{M}_v$. This improves the regularization term by at least $\beta\varepsilon$: $\beta\sum_{c\in L} d_v^c[y_v^c - \hat{y}_v^c] = \beta d_v^m(y_v^m - \hat{y}_v^m) + \beta d_v^j(y_v^j - \hat{y}_v^j) = -\beta d_v^m \varepsilon + \beta d_v^j \varepsilon = \beta\varepsilon(d_v^j - d_v^m) \ge \beta\varepsilon$.

Next, the first part of the objective increases by at most εd_{max}. To see this, note that for $e \in E_j$ with $v \in e$, $\hat{x}_e \ge 1 - \hat{y}_v^j = 1 \implies \hat{x}_e = 1$ and $x_e \ge 1 - y_v^j = 1 - \varepsilon$. Therefore, for $e \in E_j$, $v \in e$, we know $\hat{x}_e - x_e = 1 - x_e \le 1 - (1 - \varepsilon) = \varepsilon$. For $e \in E_m$ with $v \in e$ we know $\hat{x}_e - x_e \le 0$, since $\hat{x}_e = \max_{u\in e}(1 - \hat{y}_u^m)$ and $x_e = \max_{u\in e}(1 - y_u^m)$, but the only difference between y_u^m and $\hat{y}_u^m$ is that $\hat{y}_v^m = y_v^m + \varepsilon \implies (1 - \hat{y}_v^m) < (1 - y_v^m)$. For all other edge sets E_c with $c \notin \{m, j\}$, $\hat{x}_e = x_e$. So $\sum_{e:v\in e}[\hat{x}_e - x_e] \le \varepsilon d_{max}$. So when $\beta > \varepsilon$, we improve the objective by moving weight $y_v^j = \varepsilon$ from a non-minority vote cluster $j \notin \mathcal{M}_v$ to some $m \in \mathcal{M}_v$. Hence for every $v \in V$, $\sum_{c\in\mathcal{M}_v} y_v^c = 1$ at optimality. □

Theorem 2.4 implies that if there is a unique minority vote clustering, then it is optimal for both the original objective and the LP relaxation when $\beta > d_{max}$. Whether or not the the optimal solution to the LP is the same as the ILP one, the rounded solution still corresponds to some minority vote clustering that does not meaningfully balance diversity and experience. The bound $\beta > d_{max}$ is loose in practice; our experiments show that the transition occurs for smaller β. In the next section, we use LP sensitivity analysis to better bound the phase transition computationally.

3 Bounding Hyperparameters that Yield Extremal Solutions

In order to find a meaningful balance between experience and diversity, we would like to first find the *smallest* value of β, call it β^*, for which $\beta > \beta^*$ yields a minority vote clustering. After, we could consider the hyperparameter regime $\beta < \beta^*$. Given that the objective is NP-hard in general, computing β^* exactly may not be feasible. However, we will show that we can *exactly compute* the minimum value $\hat{\beta}$ for which a *relaxed* minority vote solution is no longer optimal for the LP relaxation. This has several useful implications. First, when the minority vote clustering is unique, Theorem 2.4 says that this clustering is also optimal for the ILP for large enough β. Even when the minority vote clustering is not unique, an integral minority vote solution may still be optimal for the LP relaxation for large enough β; indeed, we later observe this in real datasets. In these cases, we know that $\beta^* \leq \hat{\beta}$, which allows us to rule out a wide range of parameters leading to solutions that effectively ignore the *experience* part of our objective. Still, even in cases where an integral minority vote solution is never optimal for the LP relaxation, computing $\hat{\beta}$ lets us avoid parameter regimes where Algorithm 1 does not return a minority vote clustering.

Our approach for computing $\hat{\beta}$ is based on techniques for bounding the optimal parameter regime for a relaxed solution to a clustering objective [13, 24]. We adapt these results for our regularized objective.

The LP relaxation of our regularized objective can be written abstractly in the following form

$$\min_{\mathbf{x}} \ \mathbf{c}_e^T\mathbf{x} + \beta\mathbf{c}_d^T\mathbf{x} \ \text{ s.t. } \mathbf{A}\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq 0, \tag{3.5}$$

where $\mathbf{x}$ stores variables $\{x_e, x_v^c\}$, $\mathbf{A}\mathbf{x} \geq \mathbf{b}$ encodes constraints given by the LP relaxation of (2.3), and $\mathbf{c}_e, \mathbf{c}_d$ denote vectors corresponding to the experience and diversity terms in our objective, respectively. In this format, the LP-relaxation is a parametric linear program in β. Standard results on parametric linear programming [1] guarantee that any solution to (3.5) for a fixed value of β will in fact be optimal for a range of values $[\beta_\ell, \beta_u]$ containing β. The optimal solutions to (3.5) as a function of β correspond to a piecewise linear, concave, increasing curve, where each linear piece corresponds to a range of β values for which the same feasible LP solution is optimal.

We begin by solving this LP for some $\beta_0 > d_{max}$, which is guaranteed to produce a solution vector $\mathbf{x}_0$ that is at least a relaxed form of minority vote (Theorem 2.4) that would round to a minority vote clustering via Algorithm 1. Our goal is to find the largest value $\hat{\beta}$ for which $\mathbf{x}_0$ no longer optimally solves (3.5). To do so, define $\mathbf{c}^T = \mathbf{c}_e^T + \beta\mathbf{c}_d^T$ so that we can re-write objective (3.5) with $\beta = \beta_0$ as

Table 1: Summary statistics of datasets. The computed $\hat{\beta}$ bounds using the tools in Section 3 are much smaller than the $d_{\max}$ bound in Theorem 2.4.

Dataset	$\lvert V\rvert$	$\lvert E\rvert$	L	d_{max}	$\hat{\beta}$
music-blues-reviews	1106	694	7	127	0.50
madison-restaurants-reviews	565	601	9	59	0.42
vegas-bars-reviews	1234	1194	15	147	0.50
algebra-questions	423	1268	32	375	0.50
geometry-questions	580	1193	25	260	0.50

$$\min_{\mathbf{x}} \ \mathbf{c}^T\mathbf{x} \ \text{ s.t. } \mathbf{A}\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq 0. \tag{3.6}$$

Finding $\hat{\beta}$ amounts to determining how long the minority vote solution is "stable" as the optimal solution to (3.6). Consider a perturbation of (3.6),

$$\min_{\mathbf{x}} \ \mathbf{c}(\theta)^T\mathbf{x} = \mathbf{c}^T\mathbf{x} - \theta\mathbf{c}_d^T\mathbf{x} \text{ s.t. } \mathbf{A}\mathbf{x} \geq \mathbf{b}, \mathbf{x} \geq 0, \tag{3.7}$$

where $\theta = \beta_0 - \beta$ for some $\beta < \beta_0$, so that (3.7) corresponds to our clustering objective with the new parameter β. Since $\mathbf{x}_0$ is optimal for (3.6), it is optimal for (3.7) when $\theta = 0$. Solving the LP below provides the range $\theta \in [0, \theta^+]$ for which $\mathbf{x}_0$ is still optimal for (3.7):

$$\max_{\mathbf{y},\theta} \theta \text{ s.t. } \mathbf{A}^T\mathbf{y} \leq \mathbf{c} - \theta\mathbf{c}_d, \ \mathbf{b}^T\mathbf{y} = \mathbf{c}^T\mathbf{x}_0 - \theta\mathbf{c}_d^T\mathbf{x}_0. \tag{3.8}$$

Let $(\mathbf{y}^*, \theta^*)$ be the optimal solution to (3.8). The constraints imply that $(\mathbf{x}_0, \mathbf{y}^*)$ satisfy primal-dual optimality conditions for the perturbed LP (3.7) and its dual, and the objective function seeks to find the maximum value of θ such that these conditions hold. Thus, $\theta^* = \theta^+$, and $\beta = \beta_0 - \theta^+$ will be the smallest parameter value such that $\mathbf{x}_0$ is optimal for the LP relaxation.

Finally, after entering a regime where $\mathbf{x}_0$ is no longer optimal, the objective function strictly decreases. Again, by Theorem 2.4, for large enough β, the relaxed LP solution is a (relaxed) minority vote. Since we find the minimizer of the LP, the solution is the (relaxed) minority vote solution with the smallest objective. Thus, moving to the new parameter regime will no longer correspond to minority vote, either in the LP relaxation or in the Algorithm 1 rounding.

4 Numerical Experiments

Here we present three sets of experiments on real-world data to demonstrate our theory and methods. The first uses the diverse clustering objective to measure the quality of the LP relaxation and our bounds on $\hat{\beta}$; we find that regularization costs little while greatly improving diversity within clusters. Further, we show that we can use diversity regularization to discover diverse sets of reviews within product categories. The next set of experiments involves clustering regularized by user preference, and we find that we can satisfy a high percentage of preferences at a small cost. The last set of experiments studies what happens

Table 2: Summary statistics of datasets with hyperedges based on product ratings. Fast runtimes indicate the scalability of our approach.

				Runtime (seconds)			
Dataset	vol.	$\|V\|$	$\|E\|$	$\beta = 0.00$	0.07	0.14	0.21
software	11.1K	1.82K	2.00K	0.82	0.58	0.26	0.23
beauty	26.6K	3.81K	3.45K	0.55	2.33	0.5	0.47
pantry	126K	14.2K	11.2K	2.22	4.52	4.07	2.77
digital-music	137K	16.5K	17.7K	3.26	59.21	3.24	3.48
instruments	209K	27.5K	21.7K	6.19	10.08	6.74	6.52
arts	419K	56.2K	41.8K	15.22	18.77	16.6	15.39
office	714K	101K	59.9K	30.15	83.97	23.38	22.74
patio	714K	103K	73.4K	40.93	80.19	29.57	25.33
grocery	1.02M	127K	88.7K	38.31	188.05	78.9	38.36
automotive	1.56M	194K	156K	62.9	112.98	74.87	66.83

if we apply the diversity-regularized clustering interatively. Here, we see a clear effect of the regularization on team dynamics over time. An implementation of our algorithm, and all code and datasets used to run these experiments is found at https://tinyurl.com/diverse-and-experienced-groups.

4.1 Datasets and algorithm scalability The datasets we use come from online user reviews sites and the MathOverflow question-and-answer site. We procure two distinct types of datasets. In in first case, the nodes are users on the given site while hyperedges are groups of users that post reviews or answer questions in a certain time period. Table 1 contains summary statistics for these datasets. In the second case, nodes are still users while hyperedges now link groups of reviewers who gave the same rating to the same product for a given Amazon product category. Table 2 shows summary statistics and runtimes for these datasets.

Hyperedges based on posting time.
1. music-blues-reviews. This dataset comes from a crawl of Amazon product reviews [23]. We consider all reviews on products that include the tag "regional blues," a subset of vinyl music. We partition the reviews into month-long segments. For each time segment, we create hyperedges of all users who posted a review for a product with a given sub-tag (hyperedge category) of the regional blues tag (e.g., Chicago Blues).
2. madison-restuarants-reviews, vegas-bars-reviews. These datasets are derived from reviews on Yelp[1] for restaurants in Madison, WI and bars in Las Vegas, NV. We perform the same time segmentation as the music-blues-reviews dataset, creating hyperedges of groups of users who reviewed a place with a given sub-tag (e.g., Thai restaurant for Madison) in a given time segment.
3. algebra-questions, geometry-questions. These are derived from users answering questions on MathOverflow that contain the tag "algebra" or "geometry". We use the same time segmentation and hyperedge construction as for the reviews datasets. The sub-tags are given by all tags matching the regular expressions `*algebra*` or `*geometry*` (e.g., lie-algebras or hyperbolic-geometry).

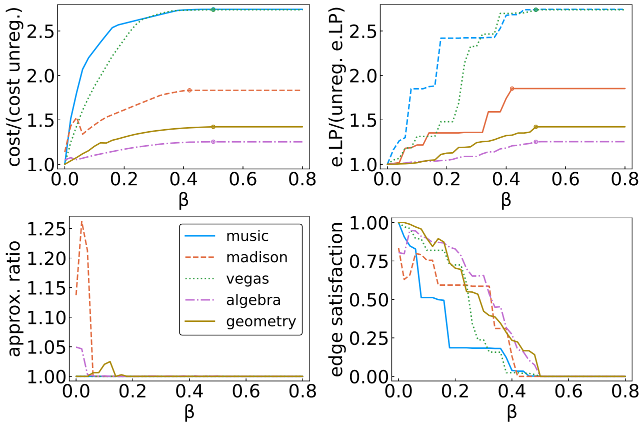

Figure 1: Various performance metrics as a function of β. Dots mark the corresponding $\hat{\beta}$.

Hyperedges based on product ratings.
1. software, beauty, pantry, digital-music, (musical)-instruments, arts, office, patio, grocery, automotive. Here, hyperedges connect all reviewers who gave a particular rating (1–5 stars) to a product in one of 10 Amazon product categories. We chose 10 medium/large categories among the total 29 to keep the list of results manageable and runtimes/computational expenses reasonable, as we ran the code on a laptop computer.

Here, a diverse clustering of users from a review platform corresponds to composing groups of users for a particular category that contains both experts (with reviews in the given category) and those with diverse perspectives (having reviewed other categories). The reviews from these users could then be used to present a "group of reviews" for a given category. A diverse clustering for the question-and-answer platforms joins users with expertise in one math topic with those who have experiences in another topic. This serves as an approximation to how one might construct experienced and diverse teams, given historical data on experiences.

Scalability. The datasets in Table 2 are arranged in order of increasing volume. We can see that runtimes for hypergraphs with hundreds of thousands of nodes and hyperedges are on the order of a minute on a laptop computer. These results indicate good scalability of our method across all regularization strengths. Runtimes for the first set of (smaller) datasets is on the order of 1 second or less, and are omitted for brevity.

4.2 Diversity regularization Here, we analyze the performance of Algorithm 1 on datasets with hyperedges constructed based on posting time (Table 1) and those constructed based on product ratings (Table 2).

I. Hyperedges based on posting time. Here, we examine the performance of Algorithm 1 for various regularization strengths β and compare the results to the unregularized case (Figure 1). We observe that the regularization only yields mild increases in cost compared to the optimal solution of the original unregularized objective. This "cost of diversity" ratio is always smaller than

[1] https://www.kaggle.com/yelp-dataset/yelp-dataset

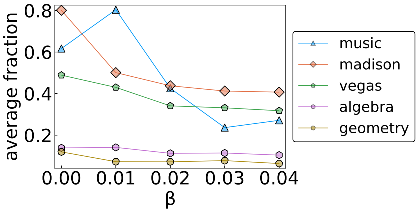

Figure 2: f_{within} for within-cluster reviews/posts.

3 and is especially small for the MathOverflow datasets (Figure 1, top left). Furthermore, the ratio between the LP relaxation of the regularized objective and the LP relaxation of the unregularized ($\beta = 0$) objective has similar properties (Figure 1, top right). This is not surprising, given that every node in each of the datasets has a color degree of zero for some color, and thus for very large values of β, each node is put in a cluster where it has a zero color degree, so that the second term in the objective is zero. Also, the approximation factor of Algorithm 1 on the data is small (Figure 1, bottom left), which we obtain by solving the exact ILP, indicating that the relaxed LP performs very well. In fact, solving the relaxed LP often yields an integral solution, meaning that it solves the ILP. The computed $\hat{\beta}$ bound also matches the plateau of the rounded solution (Figure 1, top left), which we also expect from the small approximation factors and the fact that each node has at least one color degree of zero. We also examine the "edge satisfaction", i.e., the fraction of hyperedges whose nodes are clustered to the same color as the hyperedge [4] (Figure 1, bottom right). As regularization increases, more diversity is encouraged, and edge satisfaction decreases. Lastly, we note that the runtime of Algorithm 1 is small in practice, taking at most a couple of seconds.

Within-cluster diversity. Next, we examine the effect of regularization strength on diversity within clusters. To this end, we measure the average fraction of within-cluster reviews/posts. Formally, for a clustering $\mathcal{C}$, this measure, which we call f_{within}, is calculated as follows: $f_{\text{within}} = \sum_{i \in L} |\mathcal{C}(i)|/|V| \sum_{v \in \mathcal{C}(i)} d_v^i / d_v$. In computing this measure, within each cluster we compute the fraction of all user reviews/posts having the same category as the cluster. Then we average these fractions across all clusters, weighted by cluster size. Figure 2 shows that f_{within} decreases with regularization strength, indicating that our clustering framework yields meaningfully diverse clusters.

Case study: Mexican restaurants in Madison, WI. We now take a closer look at the output of Algorithm 1 on one dataset to better understand the way in which it encourages diversity within clusters. We cluster each reviewer in madison-restaurant-reviews to write reviews of restaurants falling into one of nine cuisine categories. After that, we examine the set of reviewers grouped to review Mexican restaurants. To compare the diversity of experience for various regularization strengths, we plot the distribution of reviewers' *majority vote assignment categories* in Figure 3 (left). In other words, the majority category is the one in which they have published the most reviews. We see that as β increases, the cluster becomes more diverse, as the dominance of the Mexican majority category gradually subsides, and it is overtaken by the Chinese category. At $\beta = 0$ (no regularization), 95% of nodes in the Mexican reviewer cluster have a majority category of Mexican, while at $\beta = 0.04$, only 20% still do. Thus, as regularization increases, we see greater diversity within the cluster, as "expert" reviewers from other cuisines are clustered to review Mexican restaurants.

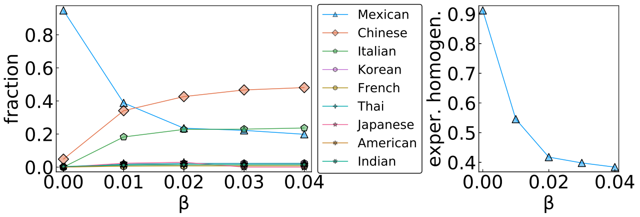

Figure 3: (Left) Distribution of node (reviewer) majority categories within the Mexican restaurant review cluster. (Right) The fraction (experience homogeneity score) of user reviews in the Mexican cluster that were written in that same category.

Similarly, as β increases we see a decrease in the fraction of users' reviews that are for Mexican restaurants, when this fraction is averaged across all users assigned to the Mexican restaurant cluster (Figure 3, right side). We refer to this ratio as the *experience homogeneity score*, which for a cluster $\mathcal{C}(i)$ is formally written as experience_homogeneity_score$(C(i)) = \sum_{v \in \mathcal{C}(i)} d_v^i / d_v$. This measure is similar to f_{within} except that we look at only one cluster. However, this score does not decrease as much as the corresponding fraction in Figure 3 (left side), falling from 91% to 38%, which illustrates that while the "new" reviewers added to the cluster with increasing β have expertise in other areas, they have also reviewed some Mexican restaurants in the past.

II. Hyperedges based on ratings: finding diverse review sets. Here, we assess the quality of solutions given by Algorithm 1 for the diversity-regularized objective on the 10 Amazon product category datasets shown in Table 2. To do this quantitatively, for each hypergraph we assign to reviewer v a *reviewer score* r_v, equal to the average rating that this reviewer gave to products in the dataset. This provides a measure of how negative or positive the reviewer tends to be when rating products, on a scale from 1 to 5. We can in turn use these scores to provide an aggregate measure of how positive or negative an entire cluster $C(i)$ of reviewers tends to be. This is accomplished by averaging reviewer scores: average_cluster_reviewer_score$(C(i)) = 1/|C(i)| \sum_{v \in C(i)} r_v$. This score is plotted in Figure 4 for the 1-star review cluster, $C(1)$. For low β, the average score of the cluster is closer to 1 while tending to 5 with increased β. This suggests that without

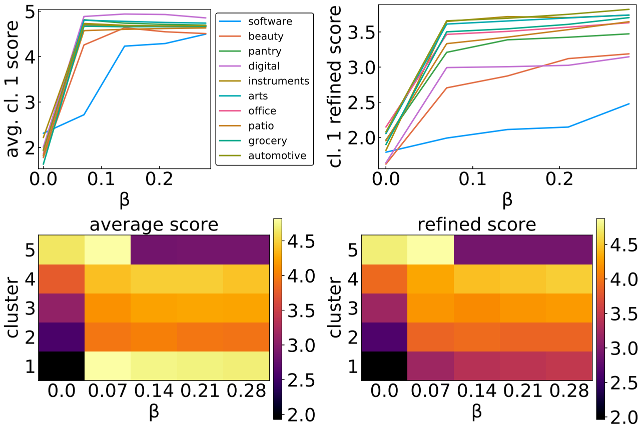

Figure 4: (Left) Average cluster reviewer score for cluster 1. (Middle left) Refined average cluster reviewer score for cluster 1. (Middle right) Distribution of average cluster reviewer score in the Amazon Pantry product category. (Right) Distribution of refined average cluster reviewer score in the same category.

regularization, cluster $C(1)$ is simply bringing together reviewers that tend to give low ratings to all products. On the other hand, increasing β means more positive reviewers are assigned to the 1-star cluster (the average cluster score increases), revealing products that receive low ratings *even from reviewers who otherwise tend to give more positive scores*. However, this measure does not distinguish whether the score increases because of *whole* hyperedges (i.e., products) are placed in the 1-star cluster, or because a handful of positive individuals were pulled from different hyperedges in order to improve the regularization term in our objective. To see whether more whole hyperedges are indeed being placed in the 1-star cluster, we define a refined cluster reviewer score given by $\text{refined\_score}(C(i)) = \frac{1}{|E_{internal}(i)|}\sum_{e\in E_{internal}(i)}\left[1/|e|\sum_{v\in e} r_v\right]$, where $E_{internal}(i) = \{e \in E \text{ s.t } e \subset C(i), \ell(e) = i\}$. In words, for each i-star hyperedge, we average the reviewer scores in that hyperedge, and then average that value across all hyperedges in the cluster. Figure 4 (top right) shows that this score increases with regularization for the 1-star cluster, meaning that we isolate 1-star products that have received poor reviews even from reviewers that are otherwise positive.

4.3 Preference regularization Here, we show that our objective is able to accommodate user preferences at a low cost. Figure 5 shows the preference satisfaction score (fraction of nodes assigned to their preferred category) for a majority vote (assign p_v^c=0 for one majority color and the rest to 1) preference distribution (top left), and a random preference distribution (top right) based on the solution given by Algorithm 1. As β increases these scores increase in both cases but the increase is even more profound in the case of a random preference distribution. This is intuitive since a majority vote preference is symbiotic with the "edge" part of the objective, while a random preference distribution almost surely competes against it. At the same time, the cost of the preference-regularized solution from Algorithm 1 in terms of the cost of the optimal solution to the unregularized objective (Figure 5 (bottom left) and Figure 5 (bottom right)) is comparatively very modest (never more than 3) even in the case of a random node preference distribution.

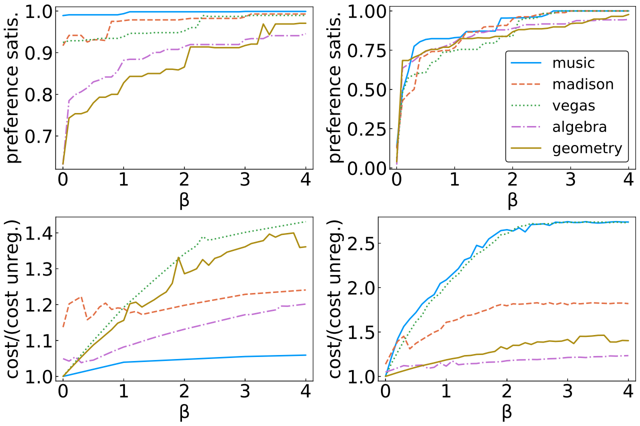

Figure 5: Metrics for preference regularization.

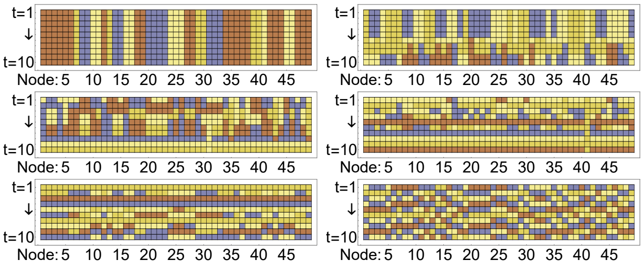

Figure 6: Color assignments over time for a subset of nodes and tags in the geometry-questions dataset for different regularization parameters β (from left to right and top to bottom: $\beta = 0$, 0.07, 0.1, 0.2, 0.4, 0.7).

4.4 Dynamic group formation Here, we consider a dynamic variant of our diversity-regularized framework where we iteratively update the hypergraph. More specifically, given the hypergraph up to time t, we (i) solve our regularized objective to find a clustering $\mathcal{C}$ and (ii) create a set of hyperedges at time $t + 1$ corresponding to $\mathcal{C}$, i.e., all nodes of a given color create a hyperedge. At the next step, experience levels of all nodes change. This mimics a scenario in which teams are repeatedly formed via Algorithm 1 for various types of tasks. We only track the experiences from a window of the last w time steps; in other words, the hypergraph just consists of the hyperedges appearing in the previous w steps. We initialize node histories based on the aforementioned datasets. After, we run the iteration for w steps to "warm start" the dynamical process, and consider this state to be the initial condition. Finally, we run the iterative procedure for T times.

When $\beta = 0$ (i.e., no regularization), after the first step, the clustering will create new hyperedges that increase the experience levels of each node for some

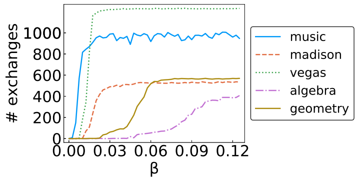

Figure 7: Mean number of node exchanges.

color. In the next step, no node has any incentive to cluster with a different color than the previous time step, so the clustering will be the same. Thus, the dynamical process is entirely static. At the other extreme, if $\beta > d_{max}$ at every step, then the optimal solution is a minority vote assignment by Theorem 2.4. In this case, after each step, each node v will increase its color degree in one color, which may change its minority vote solution in the next iteration. With randomly broken times, this leads to uniformity in the historical cluster assignments of each node as $T \to \infty$.

For several datasets, we ran the dynamical process for $T = 50$ steps. We say that a node *exchanges* if it is clustered to different colors in consecutive time steps. Figure 7 shows the mean number of exchanges. As expected, for small β, nodes are always assigned the same color, resulting in no exchanges; for large enough β, nearly all nodes exchange in the minority vote regime. Figure 6 shows the clustering of nodes on a subset of the geometry-questions dataset for different regularization levels. For small β, nodes accumulate experience before exchanging. When β is large, nodes exchange at every iteration. This is the large-β regime in Figure 7.

5 Discussion

We present a new framework for clustering that balances diversity and experience or preference and experience in cluster formation. We cast our problem as a hypergraph clustering task, where a regularization parameter controls cluster diversity, and write an algorithm that achieves a 2-approximation for any value of the regularization parameter. In numerical experiments, the approximation algorithm is effective and finds solutions that are nearly as good as the unregularized objective.

Managing hyperparameters is generally daunting. Remarkably, we are able to characterize solutions for extremal values of the regularization parameter and also compute intervals for which it provides a meaningful tradeoff for our objective. As the regularization parameter changes from zero to infinity, our problem transitions from being NP-hard to polynomial time solvable. In future work, we plan to explore how and when this transition occurs, and whether we can obtain better parameter-dependent approximation guarantees.

6 Acknowledgements

This research was supported in part by ARO Award W911NF-19-1-0057, ARO MURI, NSF CAREER Award IIS-2045555, and NSF Award DMS-2146079.

References

[1] I. Adler and R. D. Monteiro. A geometric view of parametric linear programming. *Algorithmica*, 1992.

[2] S. Ahmadian, A. Epasto, R. Kumar, and M. Mahdian. Clustering without over-representation. In *KDD*, 2019.

[3] S. Ahmadian, A. Epasto, R. Kumar, and M. Mahdian. Fair correlation clustering. In *AISTATS*, 2020.

[4] I. Amburg, N. Veldt, and A. Benson. Clustering in graphs and hypergraphs with categorical edge labels. In *TheWebConf*, pages 706–717, 2020.

[5] S. Barocas and A. D. Selbst. Big data's disparate impact. *Calif Law Rev*, 104:671, 2016.

[6] A. R. Benson, D. F. Gleich, and J. Leskovec. Higher-order organization of complex networks. *Science*, 2016.

[7] P. Castells, N. J. Hurley, and S. Vargas. Novelty and diversity in recommender systems. In *Recommender systems handbook*, pages 881–918. Springer, 2015.

[8] L. Chen, G. Zhang, and E. Zhou. Fast greedy map inference for determinantal point process to improve recommendation diversity. In *NeurIPS*, 2018.

[9] F. Chierichetti et al. Fair clustering through fairlets. In *NeurIPS*, pages 5029–5037, 2017.

[10] A. Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2):153–163, 2017.

[11] S. Corbett-Davies and S. Goel. The measure and mismeasure of fairness: A critical review of fair machine learning. *arXiv:1808.00023*, 2018.

[12] D. R. Forsyth. *Group dynamics.* Cengage, 2018.

[13] J. Gan et al. Graph Clustering in All Parameter Regimes. In *MFCS*, 2020.

[14] S. Hadley. Approximation techniques for hypergraph partitioning problems. *Discrete Appl. Math*, 1995.

[15] S. E. Jackson and M. N. Ruderman. *Diversity in work teams: Research paradigms for a changing workplace.* American Psychological Association, 1995.

[16] J. Kleinberg et al. Algorithmic fairness. In *AEA papers and proceedings*, 2018.

[17] J. Kleinberg and M. Raghu. Team performance with test scores. *ACM TEAC*, 6(3-4):1–26, 2018.

[18] E. L. Lawler. Cutsets and partitions of hypergraphs. *Networks*, 3(3):275–285, 1973.

[19] D. Levi. *Group dynamics for teams.* Sage, 2015.

[20] P. Li and O. Milenkovic. Inhomogeneous hypergraph clustering with applications. In *NeurIPS*, 2017.

[21] L. Lü et al. Recommender systems. *Phys. Rep*, 2012.

[22] N. Mehrabi et al. A survey on bias and fairness in machine learning. *arXiv:1908.09635*, 2019.

[23] J. Ni, J. Li, and J. McAuley. Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In *EMNLP-IJCNLP*, pages 188–197, 2019.

[24] S. Nowozin and S. Jegelka. Solution stability in linear programming relaxations: Graph partitioning and unsupervised learning. In *ICML*, 2009.

[25] R. L. Santos, C. Macdonald, and I. Ounis. Exploiting query reformulations for web search result diversification. In *TheWebConf*, pages 881–890, 2010.

[26] M. Stratigi et al. Fair sequential group recommendations. In *SAC*, 2020.

[27] N. Veldt, A. R. Benson, and J. Kleinberg. Hypergraph cuts with general splitting functions. *arXiv:2001.02817*, 2020.