Causal Explanations for Disparate Trends: Where and Why?

Tal Blau
Ben-Gurion University of the Negev
tbl@post.bgu.ac.il

Brit Youngmann
Technion
brity@technion.ac.il

Anna Fariha
University of Utah
Salt Lake City
afariha@cs.utah.edu

Yuval Moskovitch
Ben-Gurion University of the Negev
yuvalmos@bgu.ac.il

ABSTRACT

During data analysis, we are often perplexed by certain *disparities* observed between two groups of interest within a dataset. To better understand an observed disparity, we need *explanations* that can pinpoint the data regions where the disparity is most pronounced, along with its causes, i.e., factors that alleviates or exacerbates the disparity. This task can be complex and tedious, particularly when the dataset is large and high-dimensional, demanding an automatic system for discovering *explanations* (data regions and causes) of an observed disparity in a dataset. When offering explanations for disparities, it is critical that they are not only interpretable but also actionable—enabling users to make informed, data-driven decisions. This requires explanations to go beyond surface-level correlations and instead capture *causal* relationships.

We introduce DisEx, a framework for discovering causal Explanations for Disparities between two groups of interest. DisEx identifies data regions (subpopulations) where disparities are most pronounced (or reversed), and associates specific factors that causally contribute to the disparity. We formally define the DisEx framework and the associated optimization problem, analyze its complexity, and develop an efficient algorithm to solve the problem. Through extensive experiments over three real-world datasets, we demonstrate that DisEx generates meaningful causal explanations, outperforms prior methods, and scales effectively to handle large, high-dimensional datasets.

PVLDB Artifact Availability:
The source code, data, and/or other artifacts have been made available at https://github.com/TalBl/CausalExplanationforDisparity.

1 INTRODUCTION

Data is the main building block of modern, data-driven decision-making. People rely on the trends observed in the data to gain insights, and in turn, use those insights to draw conclusions or even make important decisions. For large, high-dimensional datasets, certain observed trends often require further drilling down or *explanations*. For example, after observing a *disparate* trend that females are more likely to experience nervous breakdowns and anxiety attacks than non-females, one might wonder: "For which subpopulation is this disparity more pronounced?", "Which factors contribute to this disparity?", "Are there particular countries/races where this trend is reversed?", and so on. Manually searching for these answers is like finding a needle in a haystack, which demands automated ways to pinpoint the subpopulation or data region where a trend is more pronounced or substantially reversed than the global trend, and identify factors that are connected to these disparate trends.

Understanding the *causal reasons* behind disparities in average outcomes between two groups is crucial for informed, data-driven decision-making aimed at addressing inequities. For example, if a policymaker recognizes the factors causally contributing to a lower average salary for a certain race or gender in a specific region compared to the rest of the population within that region, they can implement targeted corrective measures.

Gender	Ethnicity	Education	Role	YrsCoding	TC
Non-binary	White	BS	Business analyst	18-20	83K
Male	South Asian	PhD	Data analyst	15-20	124K
Female	South Asian	MS	Back-end developer	25-30	75K
Male	East Asian	BS	Back-end developer	5-10	59K

Table 1: A sample (toy) dataset over a partial schema of the Stack Overflow Annual Developer Survey dataset [1].

In this paper, we propose DisEx for automatically Explaining an observed Disparate trend. We proceed to provide three examples, highlighting unique use cases, to motivate the problem of discovering *explanations* (data regions and associated causes) of an observed disparate trend between two groups of interest within a dataset.

Example 1.1 (Investigating a disparate trend). *A social analyst Miro is examining tech workers' total-compensation data, such as the Developer Survey by Stack Overflow dataset [1], a sample of which is shown in Table 1. The dataset contains information about individuals' demographics (gender, ethnicity, age, etc.), role, experience in coding professionally, education, total compensation (TC), etc. Contrary to the common knowledge, Miro observes that the average TC of data or business analysts ($106K) is about 10% higher than the average TC of back-end developers ($96K)—a surprising trend!*

Miro wants to identify the subpopulations for which the trend is more pronounced and uncover the causes behind it. After several hours of digging, he discovers that one of the pronounced subpopulations is white males (with 40% support). On average, white male analysts ($122K) earn 12% more than white male developers ($109K). Miro further discovers that years of experience in professional coding is a major contributing factor to this disparity for white males. Specifically, among white males, having professional coding experience of 24–26 years causes a TC increase of $89K for analysts, and only $65K for back-end developers, further exacerbating the TC-gap.

After this discovery, Miro wonders if there are more data regions with similar properties. What are the major causes of the disparity in those regions? Unfortunately, the dataset contains 114 *attributes, and many of them are multi-valued, which makes manual exploration of all possible data regions an infeasible option. Furthermore, exploring all possible factors that can cause a significant disparity in the target attribute (TC) requires a more involved search. This makes it impossible for Miro to find explanations of the disparity manually.*

Example 1.2 (Debugging bias). *While analyzing a health insurance coverage dataset [66], Soha observes a disparate trend that individuals with an occupation that involves manual labor have a 13% lower chance (78%) of being covered by health insurance than* *the general population (91%). This indicates a "blue-collar bias" [24] and Soha wishes to discover data regions where this bias is more pronounced and uncover why. Turns out that for white people who never married, which constitute 29% of the population, the disparity is even more pronounced. Within this subpopulation, the chance of having health insurance for manual-labor workers (72%) is 18% lower than the general population (90%). Furthermore, among white people who never married, lacking formal education hurts the chance of having health insurance for manual-labor workers by 21%, where it boosts the chance for the general population by 7%!*

Example 1.3 (Discovering reverse trends). *Generally, males have a lower likelihood (37%) of feeling nervous frequently than non-males (45%). Madison is interested in finding subpopulations where a reverse trend exists, i.e., males have a higher likelihood of feeling nervous than non-males. To discover such a reverse trend, they need to examine all possible subpopulations, which is tedious. Our proposed system, DisEx can quickly discover some interesting subpopulations showing the reverse trend.[1] One such subpopulation is divorced people with age between 51–63 who have a recommendation to exercise from doctor, where males have a higher likelihood (47%) of feeling nervous than non-males (43%). Interestingly, within this subpopulation, not currently smoking exacerbates the situation for males (increases the likelihood of feeling nervous by 21%) but improves the situation for non-males (decreases the likelihood of feeling nervous by 14%).*

The above examples motivate the problem of investigating disparities in an *outcome variable* (e.g., total compensation) between *two groups* of interest (e.g., analysts vs back-end developers), aiming to identify (1) *where* the disparities are most pronounced (or reversed), such as a specific data region or subpopulation and (2) *why*, i.e., what factors further alleviate/exacerbate the disparity.

Desiderata. There are three key desiderata for the aforementioned problem. **First,** rarely a single causal explanation can satisfy the observed disparity for the entire population. In fact, in different subpopulations, disparity between two groups may be caused by different reasons, some contributing more to the disparity than the others. Therefore, our first goal is to discover *high-utility* subpopulations, for which a strong causal explanation exists for the observed disparity. **Second,** small subpopulations with low support w.r.t the entire population may be of high utility in terms of having a strong causal factor, however, insights drawn from such a small subpopulation are not statistically significant. Therefore, the reported subpopulations should have reasonably *high support* (data coverage). **Third,** finding the top-k high-utility and high-support subpopulations where the disparity is most pronounced, may result in redundancy. For example, "principal engineers" and "people with age between 35–45" may comprise the same individuals as most principal engineers are 35–45 years old. Therefore, beyond finding high-utility and high-support subpopulations, we must minimize the overlap among the reported k subpopulations. This alludes to the notion of *diverse* selection of subpopulations [44, 70, 77].

Problem. Given a database D, an outcome variable O, causal background knowledge in the form of a causal DAG $\mathcal{G}_D$ by Pearl's graphical causal model [47], two groups g_1 and g_2 of interest, and a parameter k, our goal is to generate a set of k *disparity explanations* that, collectively, best explain the disparities between g_1 and g_2 w.r.t the aggregated outcome variable O, according to $\mathcal{G}_D$. In this work, we consider AVERAGE as the aggregation function since it satisfies the requirements for causal analysis (more details are in Section 3.1). Each explanation consists of two components: (1) a *subpopulation* where the disparity is pronounced (or reversed), and (2) a *treatment pattern* that causally affects g_1 and g_2 disparately, within that subpopulation. The goodness of a set of explanations depends on three factors: (1) strength of the treatment patterns in the explanations to cause the observed disparity, measured in terms of *Average Treatment Effect* (ATE) [47], (2) *support* of the explanations (what fraction of tuples are covered by the reported subpopulations), and (3) *diversity* of the subpopulations in the explanations, measured in terms of non-overlap among the subpopulations.

Challenges and limitations of prior work. The key challenge here is to strike the right balance among the three desiderata. Prior work focuses on finding the top-k subpopulations in terms of support and observed disparity in the subpopulations [3, 46]. However, finding subpopulations while simultaneously satisfying the three criteria (utility, support, and diversity) poses additional challenges. Since we want to minimize the overlaps among subpopulations, we must take a *holistic* approach of finding a set of size k—a harder problem than the top-k version—due to very large search space. Furthermore, we not only focus on subpopulations with high disparity in the outcome variable but also the ones that entail good *causal explanations* that can explain the observed disparity. In summary, we extend prior work in this space in two ways: (1) we consider a harder version of the problem, finding k-sized set of subpopulations while minimizing their overlap (maximizing diversity), and (2) we prioritize high-quality causal explanations while choosing the subpopulations.

Contributions. We make the following contributions:

(1) We **formalize the problem** of generating a set of causal explanations to explain the disparity in outcomes between two, possibly overlapping, groups of interest. We define this as an optimization problem to maximize the causal explainability of these explanations while minimizing their overlap, subject to a size constraint on the number of explanations. We show its NP-hardness and also prove that the objective function for the optimization problem is submodular (Section 3).

(2) We **develop the DisEx framework** based on a three-step algorithm, building on and extending prior work in this space. First, we find candidate subpopulations [46] to identify data regions where the disparity is most pronounced. Then we identify promising *treatments* (local explanations) for each candidate subpopulation [74]. Finally, we exploit the submodularity property of our objective function to devise an effective greedy strategy and apply it to find a k-sized explanation set (Section 4).

(3) We present a thorough **empirical analysis** over 3 real-world datasets and present **3 case studies** that include 3 baselines, and 4 variations of our approach as additional comparison points. We show that DisEx generates higher quality explanations than the existing approaches and can find alternative explanations that existing approaches miss. We also find DisEx scalable and efficient in practice, with its runtime being quadratic w.r.t k and linear w.r.t the number of data tuples (Section 5).

[1] The phenomenon when all subpopulations show reverse trends is known as Simpson's Paradox [69] and DisEx can help investigate such phenomena.

2 BACKGROUND ON CAUSAL INFERENCE

We use Pearl's model for *observational causal analysis* [47] and present below a few concepts according to it. The broad goal of *causal inference* is to estimate the effect of a *treatment variable* T on an outcome variable O (e.g., the effect of YrsCoding on TC).

The Average Treatment Effect (ATE) is a commonly used measure in causal analysis, quantifying the difference in expected outcomes between treated and untreated groups [47, 53]. ATE conceptually assumes a scenario where treatment is assigned randomly. However, in observational data, treatment is not assigned randomly, and *confounding variables* that influence both treatment and outcome must be accounted for. To estimate ATE for a binary treatment T on an outcome O, the following definition is used:

$$ATE(T, O) = \mathbb{E}_Z\left[\mathbb{E}[O \mid T = 1, \mathbf{Z}] - \mathbb{E}[O \mid T = 0, \mathbf{Z}]\right]$$

Here, $\mathbf{Z}$ represents the set of confounding variables that affect both the treatment T and the outcome O, ensuring that the causal effect is isolated from confounding influences.

Example 2.1. *Suppose that we want to estimate the causal effect of* YrsCoding *on* TC *from the Stack Overflow (SO) dataset. Since the values of* YrsCoding *were not assigned at random, and having more or fewer years of coding experience and obtaining a high total compensation may depend on other attributes like* Gender*,* Education*, and* Role*, we must control for these confounding variables when estimating the causal effect of* YrsCoding *on* TC*.*

Pearl's model provides ways to account for these confounding attributes $\mathbf{Z}$ to get an unbiased causal estimate under additional assumptions: (1) The unconfoundedness assumption states that if we condition on the confounding variables $\mathbf{Z}$, then treatment T is independent of the potential outcome O, given $\mathbf{Z}$. Intuitively, this means that after conditioning on $\mathbf{Z}$, the treatment T is as good as randomly assigned. (2) The Overlap assumption ensures that for every combination of confounders, there is a nonzero probability of receiving each treatment, allowing for valid comparisons across treatment groups.

Pearl's model gives a systematic way (e.g., the backdoor criterion [47]) to find such a sufficient set of confounding variables $\mathbf{Z}$ when a *causal DAG* is available. Causal DAGs are graphical models that provide a simple way to graphically represent causal relationships among variables. A causal DAG is a specific type of a Bayesian network, where nodes represent random variables (i.e., data attributes) and edges signify potential direct causal influence.

Example 2.2. *Figure 1 depicts a partial causal DAG for the SO dataset over a subset of attributes in Table 1 as variables. Given this causal DAG, we can observe that* YrsCoding *depends on the values of an individual's* Role*,* Gender*, and* Education*.*

A causal DAG can be constructed by a domain expert as in the above example, or using existing *causal discovery* algorithms [23]. In this work, we assume the causal DAG is given as part of the input, which is a common assumption in prior work [19, 74].

In our framework for providing causal explanations for disparities between two groups, we focus on estimating the causal effect of a treatment T on an outcome O within a specific subpopulation, characterized by a *pattern* ψ. (We define the set of patterns considered in Section 3.) Consequently, our goal is to compute the Conditional Average Treatment Effect (CATE) [27, 52] rather than the ATE, as CATE captures the treatment effect within a targeted subpopulation. To estimate the CATE for a binary treatment T on an outcome O for a subpopulation ψ, we use the following definition:

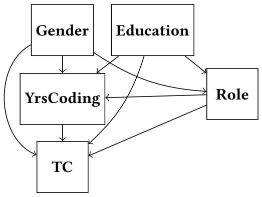


Figure 1: Partial causal DAG for the Stack Overflow dataset.

$$CATE(T, O|\psi) = \mathbb{E}_Z\left[\mathbb{E}[O \mid T = 1, \mathbf{Z}, \psi] - \mathbb{E}[O \mid T = 0, \mathbf{Z}, \psi]\right]$$

where $\mathbf{Z}$ represents a sufficient set of confounding variables.

3 DISCOVERING DISPARITY EXPLANATIONS

We consider a single-relation database over a schema $\mathbb{A}$. The schema is a vector of attributes $\mathbb{A}=(A_1, \ldots, A_m)$, where each A_i is associated with a domain $\mathtt{dom}(A_i)$. We have a categorical or continuous outcome attribute $O \in \mathbb{A}$. We also assume all other attributes in $\mathbb{A}$ are categorical (when an attribute is not categorical, we can discretize them to make them categorical). A database instance D populates the schema with a set of tuples $t=(a_1, \ldots, a_m)$ where $a_i \in \mathtt{dom}(A_i)$. We use bold letters to represent a subset of attributes $\mathbf{A} \subseteq \mathbb{A}$.

To specify a *subpopulation* (subset of tuples) from the database D, we use *patterns* [15, 37, 51, 72] that comprise conjunctive *predicates* on attribute values. Patterns are commonly used in the literature of query results explanations [35, 72, 74].

Definition 3.1 (Pattern). *Given a database instance D over schema $\mathbb{A}$, a simple predicate φ is an expression of the form A_i op a_i, where $A_i \in \mathbb{A}$, $a_i \in \mathtt{dom}(A_i)$, and op $\in \{=, \neq\}$. A* pattern *ψ is a conjunction of simple predicates, i.e., $\psi = \varphi_1 \wedge \ldots \wedge \varphi_k$. We use $\psi(D) \subseteq D$ to denote the subpopulation within D defined by ψ.*

Example 3.1. *Examples of two simple predicates in the SO dataset (Table 1) are* Gender = Female *and* Role = Data analyst*. An example of a pattern is* Gender = Female ∧ Role = Data analyst*.*

The groups of interest, g_1 and g_2, are defined by the patterns ψ_{g_1} and ψ_{g_2} respectively. In this work, we only consider equality or inequality predicates, in line with previous work on explanations that deem such predicates intuitive and more understandable [3, 16, 50]. We leave for future work the consideration of a richer class of predicates, as even this limited space of patterns is sufficiently large and handling them is computationally challenging.

3.1 Disparity explanations

Given a database D defined over a schema $\mathbb{A}$ and an outcome attribute $O \in \mathbb{A}$, we aim to discover explanations for an observed disparity in average(O) between g_1 and g_2. Our building blocks are *disparity explanations* that identify *where* the average outcomes for g_1 and g_2 differ significantly and *why*.

Example 3.2. *An analyst over the SO dataset (Table 1) is interested in finding explanations of a surprising disparate observation: the average* TC *of data or business analysts (\$106k) is \$10k higher than the average* TC *of back-end developers (\$96k).*

Why only average? We focus on comparing the average outcomes of two groups. Our framework is designed to uncover *causal explanations* for differences in these averages, leveraging the concept of CATE as discussed in Section 2. Since CATE inherently relies on *expectations* (weighted averages), it is particularly suited for analyzing aggregate averages of outcomes. Aggregate functions such as `SUM` or `COUNT`, on the other hand, depend on the *number of tuples* in the data, which does not directly align with causal effect estimates. While non-causal approaches to explanations [33, 35, 42, 51, 71, 72] can support a variety of aggregate functions, methods that are based on causal estimates typically focus on averages [38, 54, 74, 75].

Mutable and immutable attributes. We assume the attributes set $\mathbb{A} \setminus \{O\}$ is partitioned into two disjoint sets: *mutable* attributes that can be used to define what affects the outcome (e.g., years of coding, education) and *immutable* attributes, which are inherent and cannot be changed (e.g., ethnicity, gender). We use immutable attributes to define the subpopulations. Formally, let $\mathbf{I} \subset \mathbb{A}$, denote the set of immutable attributes and $\mathbf{M} \subset \mathbb{A}$ denote the set of mutable attributes, where $\mathbf{M} \cap \mathbf{I} = \emptyset$ and the outcome $O \notin \mathbf{M} \cup \mathbf{I}$. This categorization makes sure that explanations consist solely of mutable attributes that can imply corrective measures to reduce the disparity among the groups. We assume that a domain expert provides this categorization of attributes. This is similar to prior work on counterfactual explanations, where certain attributes are excluded in the explanation as they are non-actionable [21, 30, 31].

Definition 3.2 (disparity explanation). *Given a database instance D over a schema $\mathbb{A}$, an outcome variable $O \in \mathbb{A}$ and two groups of interest g_1 and g_2, a disparity explanation ϕ is defined as a pair of patterns (ψ_g, ψ_e) where:*

(1) ψ_g is defined by attributes in $\mathbf{I} \subset \mathbb{A}$, highlighting a subpopulation with significant disparity between g_1 and g_2 in terms of `AVG`(O).

(2) ψ_e is defined by attributes in $\mathbf{M} \subset \mathbb{A}$, indicating a treatment that can explain the disparity between g_1 and g_2 within $\psi_g(D)$.

To assess the impact of the treatment ψ_e on the outcome O within the subpopulation $\psi_g(D)$, we compare the causal effect of ψ_e on O within the two subpopulations: $(\psi_g \wedge \psi_{g_1})(D)$ and $(\psi_g \wedge \psi_{g_2})(D)$.

Example 3.3. *Continuing with our running example, where g_1 is `analysts` and g_2 is `back-end developers`, an example disparity explanation is: Among white males, having professional coding experience of 24–26 years boosts `TC` for `analysts` more than `back-end developers`. Here, the subpopulation pattern ψ_g is defined by `Ethnicity=White` ∧ `Gender=Male` and the treatment pattern ψ_e is `YrsCoding` = 24–26 years. Within this subpopulation, the average `TC` for `analysts` and `back-end developers` are \$122K and \$109K, respectively (gap is \$13K).*

3.2 Problem formulation

We now formally define the problem of finding disparity explanations. We assume that we are given a database instance D with schema $\mathbb{A}$, a causal model $\mathcal{G}_D$ on $\mathbb{A}$, outcome $O \in \mathbb{A}$, and two groups g_1 and g_2 defined by the patterns ψ_{g_1} and ψ_{g_2}. Let $\{\phi_1, \phi_2, \ldots, \phi_l\}$ be a set of possible disparity explanations of size l. Our goal is to find a bounded-sized set of disparity explanations Φ to identify subsets of the data that (1) provide insights into the disparity between g_1 and g_2, and (2) avoid redundancy across different subsets to cover different data regions. To this end, we define the *IScore* and *NOverlap* of Φ. Intuitively, the *IScore* of Φ measures the usefulness of Φ in explaining the disparity by combining the *supports* in the data of all disparity explanations in Φ and their explainability scores. *NOverlap* is used to measure the quality of lacking redundancy across different disparity explanations. We next formally define the *IScore* and *NOverlap* of a set of disparity explanations Φ.

Explainability score. A disparity explanation $\phi = (\psi_g, \psi_e)$ is considered useful if (*i*) it reveals a significantly different effect of the treatment ψ_e on the outcome O for the subpopulation $\psi_g(D)$ between g_1 and g_2, and (*ii*) it constitutes a significant portion of the data. We next formalize these terms.

The unnormalized disparity score Δ of a disparity explanation $\phi = (\psi_g, \psi_e)$ measures the absolute difference between the two CATE values: one computed over the subpopulation $(\psi_g \wedge \psi_{g_1})(D)$ and the other over $(\psi_g \wedge \psi_{g_1})(D)$. Formally,

$$\Delta(\phi) = \left|CATE_{\mathcal{G}_D}(\psi_e, O|\psi_g \wedge \psi_{g_1}) - CATE_{\mathcal{G}_D}(\psi_e, O|\psi_g \wedge \psi_{g_2})\right|$$

Definition 3.3 (Normalized Disparity Score (NDScore)).

$$NDScore(\phi) = 1 - \frac{1}{e^{\lambda \times \Delta(\phi)}}$$

In the above expression, λ is a scaling parameter that controls how sensitive the mapping is. For example, a low value of λ will make the NDScore slowly increase from 0 to 1 as the disparity score Δ increases from 0 to ∞. In contrast, a high value will cause NDScore to saturate to 1 very quickly as Δ increases. The parameter λ should be tuned by a domain expert who has a good understanding of the expected range of the CATE values.

Example 3.4. *Continuing from Example 3.3, the support for the disparity explanation, defined as the fraction of white males working as analysts or back-end developers among the respondents of the Stack Overflow Developer Survey, is $\frac{19{,}180}{47{,}702} = 40.21\%$. The disparity score for this explanation is 23,940. By setting $\lambda = 5.0 \times 10^{-7}$, for instance, we obtain an NDScore of 0.0119 (1.19%). Note that this is an actual disparity explanation that we found during our empirical analysis over the SO dataset (Table 3 Row 4).*

In order to prioritize disparity explanations that cover a large portion of the given database, we use the notion of *support*. The support of a disparity explanation $\phi = (\psi_g, \psi_e)$ is defined by the fraction of tuples $\in D$ that take part in the explanation, namely, tuples that satisfy the patterns in disparity explanation. Formally,

$$support(\phi) = \frac{|\psi_{g \wedge g_1}(D) \cup \psi_{g \wedge g_2}(D)|}{|D|}$$

Intuitively, the higher the support of a disparity explanation, the more interesting it is, as it applies to a larger portion of the population. We prefer disparity explanations with high support.

Finally, we define the explainability score *IScore* of a set of disparity explanations as the average NDScore of each disparity explanation within the set, weighted by its support.

Definition 3.4 (Explainability Score (IScore)). *For a set of disparity explanations $\Phi = \{\phi_1, \phi_2, \phi_3, \ldots\}$ defined by the same outcome variable O and groups g_1 and g_2,*

$$IScore(\Phi) = \sum_{\phi \in \Phi} NDScore(\phi) \times support(\phi)$$

Disparity explanations overlap. We are interested in a set of disparity explanations to reveal and explain the difference in outcome for two groups of interest. Given two disparity explanations $\phi_1 = (\psi_g^1, \psi_e^1)$ and $\phi_2 = (\psi_g^2, \psi_e^2)$ defined by the same outcome variable O and groups g_1 and g_2, $Overlap(\phi_1, \phi_2)$ is quantified by the number of tuples shared between $\psi_g^1(D)$ and $\psi_g^2(D)$. Formally:

$$Overlap(\phi_1, \phi_2) = |\psi_g^1(D) \cap \psi_g^2(D)|$$

Since we prefer disparity explanations with low overlap, we define the notion of *non-overlap* to quantify the degree of exclusiveness among disparity explanations within a given set of such explanations. This definition is based on a similar notation of overlap between prediction rules presented in [32].

Definition 3.5 (Non-Overlap (NOverlap)). *Given a set of disparity explanations $\Phi = \{\phi_1, \ldots, \phi_l\}$ with cardinality l, defined by the same outcome variable O and groups g_1 and g_2, $NOverlap(\Phi)$ quantifies the number of tuples shared between pair-wise disparity explanations. Formally:*

$$NOverlap(\Phi) = \frac{|D| \cdot l^2 - \sum_{\phi_i, \phi_j \in \Phi} overlap(\phi_i, \phi_j)}{|D| \cdot l^2}$$

In this definition, the higher the overlap between various pairs of disparity explanations, the lower the non-overlap value. Since $|\Phi| = l$, the overlap value is upper-bounded by $|D| \cdot l^2$. We normalize the result by dividing by $|D| \cdot l^2$ to stick to the range $[0, 1]$

We are now ready to formally define the problem of selecting disparity explanations. At a high level, our goal is to select a bounded-sized set of disparity explanations such that the overlap is minimized (i.e., *NOverlap* is maximized) and the *IScore* is maximized. We use a parameter α to strike a balance between the two objectives.

Problem 1 (Disparity Explanation Selection). *Given a database instance D with schema $\mathbb{A}$, a causal model $\mathcal{G}_D$ on $\mathbb{A}$, outcome $O \in \mathbb{A}$, two groups of interest g_1 and g_2, a set of possible disparity explanations $\{\phi_1, \phi_2, \ldots, \phi_l\}$, a fixed budget $k \in \mathbb{N}^+$, and a balance parameter $\alpha \in [0, 1]$, select a disparity explanation set $\Phi \subseteq \{\phi_1, \phi_2, \ldots, \phi_l\}$, such that:*

*(1) **(size constraint)** $|\Phi| \le k$, and*
*(2) **(objective)** $f(\Phi) = \alpha \cdot IScore(\Phi) + (1 - \alpha) \cdot NOverlap(\Phi)$ is maximized.*

Intuitively, the first term in the objective function quantifies the overall explainability score of Φ, and the second term considers the coverage of the dataset by the elements of Φ. We can show that the problem of Disparity Explanation Selection is intractable. In particular, given a bound B over the minimal value of the objective function, the corresponding decision problem is NP-Hard.

Proposition 3.1. *Given a set of candidate disparity explanations Φ_c, a budget k, a balance parameter $\alpha \in [0, 1]$, and a bound B, determining whether $\exists \Phi \subseteq \Phi_c$ s.t $|\Phi| \le k \wedge f(\Phi) \ge B$ is NP-hard.*

Furthermore, we show that the above objective function $f(\Phi)$ is submodular, which justifies our choice of a greedy approach to select a set of k disparity explanations, as described in Section 4.3.

Proposition 3.2. *The objective function $f(\Phi) = \alpha \cdot IScore(\Phi) + (1 - \alpha) \cdot NOverlap(\Phi)$ is submodular.*

The proofs are in our technical report [17].

4 THE DISEX ALGORITHM

Recall that given a set of possible disparity explanations, finding a subset Φ of size k that maximizes the objective function $f(\Phi)$ in Problem 1 is intractable (Proposition 3.1). Moreover, the number of possible disparity explanations may be exponential in the number of attributes. Thus, simply considering the set of all possible disparity explanations is inefficient. DisEx avoids generating all possible disparity explanations and instead generates a set of potentially promising ones. Then it applies a greedy approach to select a subset of disparity explanations of size k.

The DisEx algorithm is outlined in Algorithm 1. It consists of three parts: (1) generating patterns to identify promising subpopulations by utilizing the DivExplorer algorithm [46] (line 2), (2) identifying promising treatment patterns for each candidate subpopulation by using the approach outlined in [74] (line 4), and (3) finding a set of disparity explanations using a greedy approach (line 6). We leverage existing solutions (e.g., [46, 74]) where applicable, and develop novel techniques to build on them. The first two steps leverage existing solutions to find candidate disparity explanations, while the third step offers a greedy approach to find a feasible solution.

Note that while DisEx avoids generating all possible disparity explanations (as their number grows exponentially with the number of attributes and their domain values), given all possible disparity explanations, a greedy approach for selecting a subset from them has an approximation guarantee w.r.t the optimal solution, as the objective function $f(\Phi)$ is a non-negative, monotone, and submodular (Proposition 3.2).

4.1 Step 1: finding "where"

Our first goal is to find data regions where the disparity between the two groups g_1 and g_2 is significant, i.e., the difference in their aggregate outcome variable O is substantial. To this end, we leverage the DivExplorer algorithm [46]. DivExplorer is an efficient algorithm designed to analyze the behavior of classification models. Its primary goal is to identify regions in the data where the overall performance metric (e.g., false positive/negative rates) deviates significantly from its value over the entire dataset. Given a divergence metric (e.g., false positive rate), DivExplorer identifies subsets of the data, defined by patterns, where this metric shows a significant disparity compared to the overall population.

We adapt the concept of divergence to compare two groups, g_1 and g_2, rather than analyzing the entire dataset. Specifically, instead of comparing a subgroup to the overall dataset, we define

Algorithm 1: The DisEx algorithm

```
input  : A database instance $D$, an outcome $O$, two groups defined
         by the patterns $\psi_{g_1}$ and $\psi_{g_2}$, and a size-constraint $k$.
output : A set $\Phi$ of disparity explanations.
1 $\Phi \leftarrow \emptyset$
2 $\Psi \leftarrow$ GetSubpopPatterns$(D, O, \psi_{g_1}, \psi_{g_2})$          // Section 4.1
3 for $\psi_g \in \Psi$ do
4     $\psi_e \leftarrow$ GetTreatment$(D, O, \psi_g, \psi_{g_1}, \psi_{g_2})$    // Section 4.2
5     $\Phi \leftarrow \Phi \cup (\psi_g, \psi_e)$
6 $\Phi \leftarrow$ ApplyGreedy$(\Phi, k)$                                // Section 4.3
7 return $\Phi$
```

divergence as the difference in a performance metric within the intersection of each group and a subpopulation defined by a pattern. We utilize the difference in average outcome values as the divergence metric, enabling us to identify subpopulation patterns where the average outcome difference between g_1 and g_2 is significant.

More formally, given a subpopulation defined by the pattern ψ_g and two groups g_1 and g_2 defined by the patterns ψ_{g_1} and ψ_{g_2}, respectively, we calculate the divergence as the difference in means:

$$divergence_{\psi_g}(g_1, g_2) = |AVG(O \mid \psi_g \wedge \psi_{g_1}) - AVG(O \mid \psi_g \wedge \psi_{g_2})|$$

We limit the scope of DivExplorer exclusively to the immutable attributes, **I**, to generate candidate subpopulation patterns. The output of this step is a set of subpopulations where there is a significant disparity in the average outcome between g_1 and g_2.

4.2 Step 2: finding "why"

Our next objective is to explain the disparity within each subpopulation, generated in the previous step. Given a subpopulation pattern ψ_g, an outcome variable O, and two groups g_1 and g_2, our goal is to identify the treatment pattern ψ_e with the highest *NDScore*, as defined in Definition 3.3. To this end, we adapt CauSumX [74] to our setting. CauSumX is designed to provide causal explanations for the results of aggregate queries. An explanation pattern consists of a set of tuples from the aggregate view (i.e., the query results) and a treatment pattern is used to quantify the causal effect (in terms of the CATE value) of the treatment on the outcome within the relevant subview. In our setting, we aim to identify a treatment pattern that maximizes the difference between two CATE values. CauSumX employs a heuristic lattice traversal approach to identify promising treatment patterns with high CATE values. To make sure our explanations are concise (and, thus, interpretable), we limit the search for treatment patterns to a depth of 2. We use the same approach as CauSumX, but instead of estimating the CATE value, we estimate the IScore of the treatment pattern under consideration.

CauSumX operates on aggregate views, so the number of treatment patterns it considers is relatively small. In our setting, however, the potential number of treatment patterns is much larger, as we need to search for them for each subpopulation. To address this, we develop the following optimizations to improve runtime:

Parallelism. Since the treatment pattern for each subpopulation can be identified independently, we leverage parallelism across subpopulations. Additionally, within a subpopulation, we compute the *NDScore* of each treatment pattern in parallel to further reduce runtime.

Caching. Computing the *NDScore* for a given treatment pattern requires adding it as a node to the underlying causal DAG [74]. To avoid redundant computations, we implement caching to store previously modified causal DAGs.

Sampling. As was done in [74], instead of focusing on obtaining precise CATE values, we employ a strategy of sampling a random subset of the data for CATE estimation. We use a fixed sample size of 80,000 tuples, guided by our empirical findings, which indicate that this sampling size achieves highly accurate CATE estimations while maintaining relatively low runtime. However, the sampling ratio is a customizable system parameter and the user is free to tune it for more accurate result.

4.3 Step 3: finding the disparity explanation set

Given the set of candidate disparity explanations $\{\phi_i\}_{i=1}^{l}$obtained in the previous two steps , our goal is to identify a set of k disparity explanations with a high *IScore* and low overlap (high *NOverlap*).

To achieve this, we develop a greedy approach to iteratively select k disparity explanations to construct the disparity explanation set. At $j-th$ iteration, it selects the explanation ϕ^* such that:

$$\phi^* = \underset{\phi \in \{\phi_i\}_{i=1}^{l}}{\arg\max}\ f(\Phi_{j-1} \cup \{\phi\})$$

where Φ_{j-1} is the set of disparity explanations selected up until the j-th iteration.

An alternative approach is to formulate the problem as an Integer Linear Program (ILP), like [74]. However, given that our objective function is non-monotone and submodular, (Proposition 3.2), a greedy approach provides optimal approximation guarantees. Specifically, if steps 1 and 2 were replaced with a brute-force approach (all possible explanations were considered as candidates), the greedy algorithm guarantees to produce a solution Φ such that $f(\Phi) \geq (1-\frac{1}{e}) \times f(\Phi^*)$, where Φ^* is the optimal solution. This is the best achievable approximation guarantee by a polynomial-time algorithm [18, 45].

4.4 Usability extensions

As noted in the introduction, the DisEx framework is versatile and can be applied to various use cases, such as investigating surprising observations, debugging fairness issues, or identifying reverse trends. In each scenario, one may search for specific subpopulations where the average outcome for g_1 is either higher or lower than that for g_2. To accommodate this, we introduce a filtering step in Step 1 of DisEx, which returns only subgroups that meet the relevant condition.

Furthermore, if, for instance, we are exploring parts of the data where the average outcome for g_1 exceeds that of g_2, the explanation should elucidate this phenomenon—specifically, identifying treatments that favors g_1 compared to g_2. This can be addressed by filtering the examined treatments in Step 2 of DisEx.

4.5 Runtime complexity analysis

The maximum number of disparity explanations in a database D with attributes $\mathbb{A}$ is bounded by $|D|^{|\mathbb{A}|}$ (considering both subpopulation and treatment patterns), which is polynomial in terms of data complexity, assuming a fixed schema [67]. The final greedy step is also polynomial in the number of explanations considered. Additional operations, such as calculating CATE values, are polynomial in D, leading to worst-case polynomial data complexity. As we demonstrate in Section 5.4, DisEx is capable of efficiently handling large, high-dimensional datasets.

5 EXPERIMENTAL EVALUATION

We present an experimental evaluation that evaluates DisEx's effectiveness. We aim to address the following questions: **Q1**: How does the quality of DisEx-generated disparity explanations compare to that of existing methods? **Q2**: What is the efficiency of DisEx and how is it affected by various parameters? **Q3**: How do our proposed optimizations affect the runtime of DisEx?

5.1 Experimental setup

All experiments were performed on a Windows computer, Intel CPU, with 16 GB memory. DisEx was implemented in Python3, and is publicly available in [7]. CATE values computation was performed using the DoWhy library [62].

Datasets. We examine three commonly used datasets whose statistics are presented in Table 2. We construct the corresponding causal DAGs using [76]. To process numerical attributes, we apply equal-width binning to continuous attributes, using 10 bins.

Stack Overflow (SO): The Stack Overflow Developer Survey [1] dataset contains responses from developers worldwide, covering topics such as professional experience, education, technologies used, and employment-related information, such as annual salary.

MEPS: The Medical Expenditure Panel Survey (MEPS) [2] dataset provides detailed information on healthcare utilization, expenditures, insurance coverage, and demographic characteristics of individuals in the United States.

ACS: The American Community Survey (ACS) [66] dataset is a nationwide survey conducted by the U.S. Census Bureau, providing detailed demographic, social, economic, and housing data. We focused on seven states: California (CA), Texas (TX), Florida (FL), New York (NY), Pennsylvania (PA), Illinois (IL), and Ohio (OH).

Use cases. Throughout our experimental evaluation, we examine three scenarios that represent different use cases of DisEx (as mentioned in the Introduction). The description of the groups, the outcome variables and relevant statistics are depicted in Table 2:

(1) Investigating a surprising fact. We aim to investigate an intriguing observation: In the high-tech industry, it is generally expected that developers earn more than analysts. However, an analysis of SO data revealed a surprising trend: back-end developers (g_1) earn more on average compared to those working as data or business analysts (g_2). To analyze this phenomenon, we define the outcome variable as total compensation. *Our goal is to identify where this disparity is most pronounced and to understand the underlying reasons.* Namely, we are searching only for subpopulations where the average salary of g_1 members is higher than that of g_2 members, and explanations (i.e., treatments) that are more useful for g_1 members compared to g_2 members.

(2) Fairness Debugging. Using the ACS dataset, we focus on the relationship between occupation and its impact on health insurance coverage. Specifically, we define the outcome variable as the variable indicating whether an individual holds health insurance. g_1 represents individuals employed in manual labor occupations (such as cleaning, maintenance, farming, fishing, construction, etc.), where the health insurance coverage rate is only 78.6%. g_2 represents the entire dataset regardless of occupation, where the health insurance coverage rate is 91.5%. *Our objective is to explore the underlying factors contributing to this phenomenon.* We are searching for subpopulations where the average insurance coverage rate of g_1 is lower than that of g_2, together with a causal explanation for this disparity within this subpopulation.

(3) Finding opposite trends. We aim to investigate an intriguing observation in the MEPS dataset: while it is generally observed that men (g_1) have a higher likelihood of feeling nervous, our analysis reveals subpopulations where non-males (g_2) exhibit a greater likelihood compared to men. To examine this phenomenon, we define the outcome variable as the variable indicating whether an individual feeling nervous commonly. *Our goal is to pinpoint data parts where the trend between g_1 and g_2 is opposite from the overall data (with no-men feeling nervous more frequently) and to explore the underlying causes.*

Baselines. We consider the following baselines:

Brute Force. To evaluate the quality of our algorithm, we employ an exhaustive Brute Force algorithm, considering all possible k-sized disparity explanation sets.

Top-k. As a sanity check for the effectiveness of the non-overlap component in our objective function, we evaluate a variant of our algorithm that considers only IScore. This approach identifies the top-k disparity explanations with the highest IScore, where the first two steps of this baseline match our algorithm.

XInsight. The authors of [38] introduced XInsight, a system designed to identify both causal and non-causal patterns to explain disparities between two groups in aggregate SQL queries. Unlike our approach which focuses on subpopulations where the disparity is pronounced and finds specific explanations for those data parts, XInsight provides explanations that apply to the entire dataset. Since XInsight includes a causal discovery phase, we ensure a fair comparison under the same causal model by evaluating our results against this baseline as follows: for each treatment pattern (i.e., explanation) identified by DisEx, we also report its causal effect as assessed across the entire dataset. Our goal is to empirically demonstrate that *explanations specific to subpopulations where disparities occur are not necessarily valid explanations for the disparities observed in the overall dataset.*

DivExplorer. We consider the DivExplorer algorithm [46] as described in Section 4.1. The divergence function is set to be our IScore function (Def. 3.4), applying a constant treatment across all the subpopulations. We fixed this treatment as the one yielding the highest IScore across the overall dataset. Using this treatment, we compute the IScore for each subpopulation and select the top-k subpopulations identified by DivExplorer.

FairDebugger. The authors of [63] introduced FairDebugger, an algorithm that identifies training data subsets responsible for instances of fairness violations in the outcomes of a random forest classifier. The method measures the change in the model's behavior—such as prediction confidence or decision boundaries—caused by removing each tuple, to identify the most impactfull training samples that influence the model's output. To adapt this method to our work, we begin by calculating the IScore for the test dataset. Here again, we used a fixed treatment pattern, defined as the one yielding the highest IScore across the overall dataset. For each subpopulation, we remove the corresponding data from the training set. The difference between the original and updated IScores represents the influence of the subpopulation on the model's performance.

DivExplorer and FairDebugger serve as baselines for identifying subpopulations with disparities between the two groups of interest (g_1 and g_2). However, unlike DisEx, they do not provide causal explanations. To enable a comparison, we assigned them a fixed

Dataset	#Tuples	|I|	|M|	g_1	g_2	$\|g_1\|$	$\|g_2\|$	avg$_O(g_1)$	avg$_O(g_2)$	O
SO	47,702	4	6	Data or business analysts	Back-end developers	4088	28987	106,542$	96,609$	total compensation (TC)
MEPS	20,243	6	7	Males	Non-Males	9,220	11,023	37.58%	45.10%	likelihood of feeling nervous frequently
ACS	1,420,652	7	10	Manual Labor Occupations	Overall Data	99790	1420652	78.6%	91.5%	likelihood of having health insurance

Table 2: Details of the datasets we use for experiments and case studies.

treatment, which is expected to result in a low IScore. Nonetheless, this comparison focuses on evaluating the subpopulations identified by these baselines versus those selected by our method.

Default Parameters. Unless otherwise specified, we used the following default parameters: we set $\alpha = 0.65$ and $k = 5$. We set the support threshold to be 0.05 (i.e., we consider only groups that account for at least 5% of the data). The value of λ should differ for binary and numeric outcomes (e.g., continuous in SO vs. binary in MEPS). Thus, we used the following formula:

$$\lambda = \frac{1}{\max(O) - \min(O)}$$

We discuss this adjustment in further detail in Section 5.3.

5.2 Quality evaluation

In this part, we consider our three use cases and compare the output of DisEx with the baselines. The quantitative comparison of the objective function values for each baseline across all use cases is shown in Figure 2.

Results Summary

- The top-k baseline results in overlapping disparity explanations, demonstrating the need to consider the overlap among selected explanations.
- The output of DisEx closely matches that of Brute Force, demonstrating that our solution prioritizes efficiency without compromising quality.
- Explanations for disparity at the entire population level (as generated by XInsight) do not necessarily account for disparities within subpopulations, highlighting the need to find a specific explanation for each subpopulation.
- As expected, DivExplorer identified subpopulations with high disparity between g_1 and g_2, but the selected top-k subpopulations showed more overlap than DisEx.
- FairDebugger was less successful in identifying subpopulations with high disparity between g_1 and g_2. This is due to its top-down lattice traversal approach and the non-monotonic nature of IScore.

Use case 1 (SO): The explanations generated by DisEx are shown in Table 3. Notably, DisEx identifies subpopulations where the average salary of analysts is higher than that of back-end developers, oftentimes with this disparity being more pronounced than in the overall population (where the average salary for analysts is $106,542 and for back-end developers it is $96,609). For almost all cases, DisEx provides a different causal explanation to highlight the factor contributing to the disparity within the specific subpopulation.

In two out of the five disparity explanations, the causal effect of the chosen treatment on the entire population is not statistically significant, emphasizing the importance of providing specific explanations for each subpopulation. Additionally, in some cases, a treatment pattern can explain why the average outcome of one group is higher than another within a subpopulation. However, this relationship may not hold at the global population level, as the treatment may benefit the second group more overall. This is evident in the 4th and 5th disparity explanations. *These observations highlight the distinction between our approach and XInsight, demonstrating that subpopulation-level explanations differ from those provided at the entire population level.*

In this scenario, the output of the Brute-Force baseline is identical to ours, demonstrating that our algorithm efficiently identifies the best possible explanations. For the top-k baseline, four out of the five chosen disparity explanations matched ours. The fifth explanation selected by Top-k had a higher NDScore but low support and significant overlap with another chosen disparity explanation (as evident in its lower value of the No-Overlap objective, Figure 2).

As expected, the subpopulations identified by DivExplorer exhibited a high disparity between g_1 and g_2, which aligns with its purpose of identifying divergent subpopulations. However, the subpopulations selected by DivExplorer showed some overlap. Additionally, the fixed treatment used to explain the disparity within each subpopulation had a treatment effect that was not statistically significant (zero IScore in Figure 2), demonstrating that individual explanations are necessary for understanding disparity within different subpopulations.

FairDebugger failed to identify subpopulations with high disparity between g_1 and g_2 due to its top-down lattice traversal approach, which only drills down if the child patterns exhibit greater disparity than their parent (according to the pattern lattice). However, since IScore is non-monotonic (as CATE is non-monotonic [74]), this approach caused it to miss subpopulations with significant disparity. Further, the identified populations demonstrate significant overlap (as evident with relatively low No-Overlap value in Figure 2).

Use case 2 (ACS): The disparity explanations generated by DisEx for the second use case are show in Table 4. Surprisingly, DisEx identified subpopulations where the percentage of health insurance owners for labor workers was lower than in the entire population. Observe that most explanations (treatments) involved the education attribute, highlighting the strong causal link between education and the probability of having health insurance in the US. Notably, some treatments, such as `Education = elementary school`, have a negative effect on the outcome for both labor workers and the overall population when considering the entire data. However, within the subpopulation of natives from the southern region without disabilities, this trend no longer holds. This underscores the difference between the explanations provided by XInsight and DisEx.

Disparity Explanation	Support	Total Compensation (TC)				NDScore
		Subpopulation		Global		
		Average	CATE	Average	CATE	
For individuals aged 25–34, income growth is more influenced by the desire to remain in the same job and having 6–8 years of coding experience as an analysts compared to back-end developers.	31.94%	$96,984 $93,120	$90,389 ↑ $16,929 ↑	$106,542 $96,609	$54,424 ↑ $18,293 ↑	3.60%
For White individuals aged 25–34, income growth is more influenced by having 6–8 years of coding experience as an analysts compared to back-end developers.	22.14%	$115,777 $105,988	$44,513 ↑ $11,687 ↑	not statistically significant		1.62%
For White males aged 25–34, income growth is more influenced by having 6–8 years of coding experience as an analysts compared to back-end developers.	20.56%	$117,247 $107,115	$42,184 ↑ $10,520 ↑	not statistically significant		1.57%
For White men, income growth is more influenced by having 24-26 years of coding experience as an analysts compared to back-end developers.	40.21%	$122,605 $109,707	$89,048 ↑ $64,959 ↑	$106,542 $96,609	$72,999 ↑ $71,543 ↑	1.19%
For White individuals, income growth is more influenced by having 24-26 years of coding experience as an analysts compared to back-end developers.	43.07%	$122,766 $108,953	$87,039 ↑ $86,673 ↑	$106,542 $96,609	$72,999 ↑ $71,543 ↑	1.16%

Table 3: Disparity explanations by DisEx for the SO dataset. Patterns that form a subpopulation are colored in orange, while treatment patterns are colored in blue. We highlight the two groups of interest using yellow and pink. The first explanation highlights that for the specific subpopulation (individuals aged between 25-34 years old), the average TC for analysts observes an increase of $90,389 when the treatment hoping do the same job and having professional coding experience of 6-8 years is applied. In contrast, globally, the same treatment has an effect of only $54,424 increase for analysts.

Compared to the Brute-Force baseline, DisEx retrieved 4 out of the 5 disparity explanations, while exploring a much smaller search space. The Top-K baseline share 4 out of 5 explanation with DisEx The one explanation missed by Top-K resulted from not considering the overlap between the disparity explanations. DivExplorer and FairDebugger identified different subpopulations than DisEx, with FairDebugger finding subpopulations with smaller support and lower disparity, while DivExplorer's top subpopulations exhibited high overlap. In this case, the IScore for both solutions was low (though not zero), reinforcing the need for tailored explanations for each subpopulation.

Use case 3 (MEPS): The explanations generated by DisEx are shown in Table 5. DisEx identifies subpopulations where the average likelihood of experiencing nervous attacks very frequently for males (g_1) is lower than that of non-males (g_1), in contrast to the opposite trend observed in the overall population. The outcome generated by DisEx is identical to the Brute Force solution as well as the Top-K. Again, the subpopulations identified by FairDebugger and DivExplorer differed from those found by DisEx. The solution computed by DivExplorer sufferers from low IScore value and the results of FairDebugger were not statistically significant.

5.3 Parameters analysis

We investigate the impact of parameters on our objective function, aiming to gain insights into effective default parameter settings.

Size of the solution k: We examine the impact of solution size on our objective function. The results are shown in Figure 3(a). In all scenarios, we observe that increasing k leads to an increase in the objective function. This occurs because the IScore grows as the subpopulation size increases, and the impact of small no-overlap values becomes negligible compared to the gain in IScore. As a result, the objective function increases. *Based on our experiments, a desired balance between covering sufficient data and minimizing overlap was achieved with $k = 5$.*

Balancing IScore and non-overlap: We examine the impact of α, the parameter responsible for balancing IScore and non-overlap in our objective function on the value of the objective function. The results are shown in Figure 3(b). In the examined datasets, we observed that the non-overlap component in our objective function was higher than the IScore component. *To balance both components, our results suggest that a good default value is $\alpha = 0.65$.* Increasing α reduces the influence of the non-overlap component, leading to a lower overall score.

Scaling the IScore: We examine the impact of λ, the parameter responsible for scaling the NDScore (Definition 3.3) on our objective function. The results are shown in Figure 3(c). The NDScore is influenced by the domain of the outcome variable, as it is based on the normalized difference between two CATE values. Consequently, different values of λ may perform better depending on whether the outcome variable is binary or continuous. To provide a principled method for setting λ regardless of the outcome's domain, we found the formula described in Section 5.1 to be effective: It assigns high λ values for binary outcomes and lower values for continuous outcome variables.

5.4 Scalability

Next, we consider the effect of various parameters on runtime.

Step-by-step breakdown analysis: We start by presenting a step-by-step breakdown analysis for all datasets. The results are illustrated in Figure 5. Not surprisingly, Step 2, which focuses on identifying the causal explanation for each subpopulation, is the most computationally expensive, accounting for over 80% of the total runtime in all examined scenarios. Nevertheless, DisEx generates the solution within reasonable runtimes, even for large, high-dimensional datasets like ACS.

Disparity Explanation	Support	Likelihood of having a health insurance				NDScore
		Subpopulation		Global		
		Average	CATE	Average	CATE	
For White native men without disabilities, the percentage of health insurance ownership decreases for those with only an elementary school education in manual labor occupations, whereas it increases in the general population.	27.80%	84.87% 92.31%	30.44% ↓ 4.35% ↑	78.61% 91.58%	24.59% ↓ 0.45% ↓	29.38%
For White individuals who have never married and do not have disabilities, the percentage of health insurance ownership decreases for those without formal education in manual labor occupations, whereas it increases in the general population.	26.35%	71.90% 90.00%	24.74% ↓ 6.38% ↑	78.61% 91.58%	10.17% ↓ 0.07% ↑	26.75%
For White native men, the percentage of health insurance ownership decreases for those with only an elementary school education in manual labor occupations, whereas it increases in the general population.	32.59%	85.03% 92.65%	26.33% ↓ 4.06% ↑	78.61% 91.58%	24.59% ↓ 0.45% ↓	26.21%
Among White individuals who have never married, the percentage of health insurance ownership decreases for those without formal education in manual labor occupations, whereas it increases in the general population.	28.91%	72.30% 90.17%	21.20% ↓ 6.59% ↑	78.61% 91.58%	10.17% ↓ 0.07% ↑	24.26%
Among natives from the southern region without disabilities, the percentage of health insurance ownership decreases for those with only an elementary school education in manual labor occupations, whereas it increases in the general population.	23.41%	72.38% 87.91%	18.71% ↓ 8.49% ↑	78.61% 91.58%	24.59% ↓ 0.45% ↓	23.81%

Table 4: Disparity explanations by DisEx for the ACS dataset.

Disparity Explanation	Support	Likelihood of feeling nervous frequently				NDScore
		Subpopulation		Global		
		Average	CATE	Average	CATE	
For divorced individuals aged 51–63 who were born in the USA, do not have an asthma diagnosis, and have a doctor's recommendation to exercise, the likelihood of feeling nervous frequently increases for males who do not currently smoke compared to non-males.	1.45%	48.62% 44.86%	25.69% ↑ 16.39% ↓	37.58% 45.10%	3.39% ↓ 3.94% ↓	34.35%
For divorced individuals aged 51–63 who do not have an asthma diagnosis, and have a doctor's recommendation to exercise, the likelihood of feeling nervous frequently increases for males who do not currently smoke compared to non-males.	1.84%	46.03% 42.10%	22.54% ↑ 18.37% ↓	37.58% 45.10%	3.39% ↓ 3.94% ↓	33.57%
For divorced individuals aged 51–63 who were born in the USA, and have a doctor's recommendation to exercise, the likelihood of feeling nervous frequently increases for males who do not currently smoke compared to non-males.	1.76%	50.00% 45.56%	27.16% ↑ 13.06% ↓	37.58% 45.10%	3.39% ↓ 3.94% ↓	33.12%
For divorced individuals aged 51–63 who have a doctor's recommendation to exercise, the likelihood of feeling nervous frequently increases for males who do not currently smoke compared to non-males.	2.19%	47.48% 42.95%	21.45% ↑ 14.14% ↓	37.58% 45.10%	3.39% ↓ 3.94% ↓	29.95%
Among individuals from the Midwest region who were born in the USA, are under 29, have never married, and do not have an asthma diagnosis, the likelihood of feeling nervous frequently increases more for males than non-males when they have health insurance.	2.42%	46.63% 45.84%	20.61% ↑ 13.74% ↑	37.58% 45.10%	4.71% ↑ 2.45% ↑	6.64%

Table 5: Disparity explanations by DisEx for the MEPS dataset.

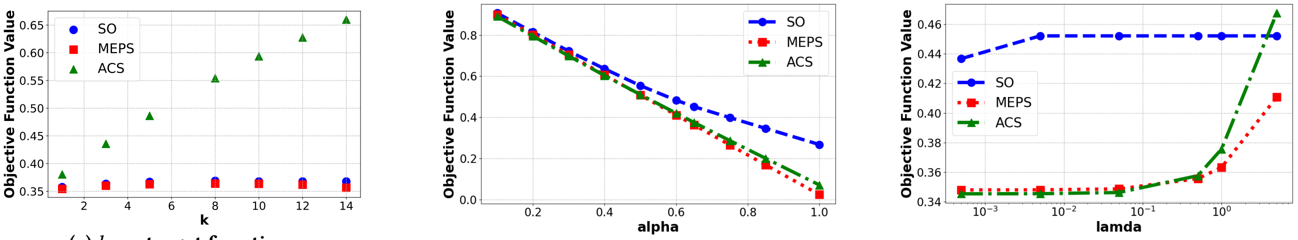
Next, we analyze how various parameters influence runtime. Since parameter variations involve sampling, we repeat each experiment 5 times and report the average runtime across all runs.

Runtime vs. the solution size: We vary the solution size k to examine its effect on the runtime. The results are shown in Figure 4(a). We note that k only affects the last step (Step 3) when selecting the solution from the candidates mined in the previous two steps. To account for the non-overlap objective, the pairwise intersection between the subpopulations corresponding to disparity explanations is evaluated. Therefore, as k increases, the number of pairwise comparisons grows quadratically and therefore the runtime grows quadratically as well.

Runtime vs. the number of attributes: We vary the number of attributes in the dataset to analyze its impact on runtime. To

Baseline	SO			ACS			MEPS		
	IScore	NO-Overlap	Target Function	IScore	NO-Overlap	Target Function	IScore	NO-Overlap	Target Function
Brute Force	0.0281	0.999782	0.3682	0.4056	0.999944	0.6133	0.0252	0.99999764	0.366379
DisEx	0.0281	0.999782	0.3682	0.3635	0.999939	0.5863	0.0252	0.99999764	0.366379
Top-K	0.0241	0.999875	0.3656	0.3488	0.999958	0.5767	0.0252	0.99999764	0.366379
DivExplorer	0	0.999724	0.3499	0.0663	0.999975	0.3931	0.0087	0.99999762	0.355686
FairDebugger	0	0.963276	0.3371	0.1013	0.995154	0.4141	0	0.98653168	0.345286

Figure 2: Objective function (broken down to IScore and non-overlap) values for different baselines across the three examined use cases.



(a) k **vs. target function.**

(b) α **vs. target function.**

(c) λ **(log scale) vs. target function.**

Figure 3: Comparison of system parameters with the target function.

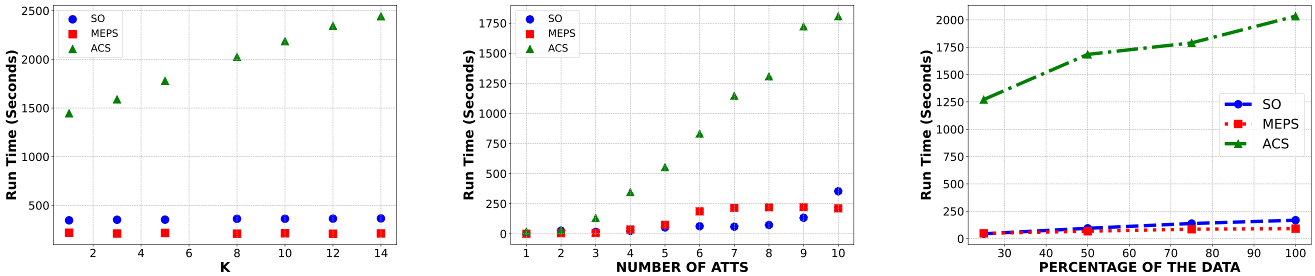


Figure 4: Effects of various parameters on runtime: (left) the set-size parameter k**, (center) number of attributes, and (right) percentage of data.**

Dataset	Step 1	Step 2	Step 3
SO	0.6	187	5.2
MEPS	0.4	91	1.8
ACS	5.0	1702	562.0

Figure 5: **Breakdown by step runtime analysis (seconds).**

do so, we randomly sampled subsets of attributes and removed them from the datasets. The results are presented in Figure 4(b). The number of attributes influences the size of the search space since more attributes lead to a larger set of subpopulations and treatment patterns to consider. Theoretically, the runtime should grow exponentially with the number of attributes. However, due to variations in the number of values per attribute and the fact that each experiment was repeated five times, this worst-case behavior was not observed.

Runtime vs. the number of tuples: We varied the number of tuples in the dataset to analyze its impact on runtime. To achieve this, we randomly sampled subsets of tuples and removed them from the dataset. Figure 4(c) illustrates that the runtime is directly influenced by the number of tuples (linearly), primarily due to the CATE computations handled by the DoWhy package. For the ACS data, where the sampling optimization was applied, the runtime remains consistent after a specific threshold of tuples. Beyond this point, all computations are performed on a random fixed-size sample of the data (as explained in Section 4.2).

Results Summary

- Step 2 of our algorithm, which focuses on identifying the causal explanation for each subpopulation, takes the longest time, accounting for over 80% of the total runtime.
- The runtime grows quadratically with the solution size k because of the pair-wise overlap computation.
- The runtime is highly influenced by the number of attributes in the data, as it affects the size of the search space.
- The runtime grows linearly with the number of tuples in the data, mainly due to CATE computations.

5.5 Ablation study

In this section, we evaluate the impact of our proposed optimizations on runtime. To do this, we tested variations of our algorithm by using each optimization individually (e.g., only parallelism and

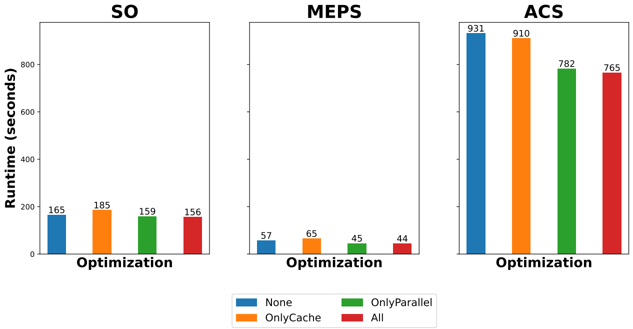


Figure 6: Effect of optimization on runtime across datasets.

only caching) as well as all optimizations together and no optimizations at all. The results are presented in Figure 6. The results indicate that the parallelism optimization has the most significant impact on runtime. In contrast, the caching optimization, which avoids redundant computations, has a relatively marginal effect. In fact, for the SO and MEPS datasets, using caching slightly decreases performance. This is because the generated graphs are relatively small, and editing them is cheaper than reading and writing them.

6 RELATED WORK

Identifying interesting subgroups in high-dimensional data. Previous research has introduced methods aiming to identify the most intriguing data subsets for exploration [5, 9, 22, 29, 36, 43, 46, 58–61, 73]. Other studies have focused on uncovering compelling data visualizations [68, 78], or pinpointing data subsets where models underperform [11]. A key part of our objective is identifying subpopulations where there is a significant disparity in the average outcomes between two groups of interest. To achieve this, we leverage existing methods when suitable—such as the DivExplorer algorithm [46] for subpopulation identification—and develop new techniques where required. The authors of [63] introduce FairDebugger, an algorithm that identifies data subsets responsible for fairness violations in the outcomes of a random forest classifier. FirDebugger identifies the most impactful data samples that significantly influence the model's output. As we demonstrated in our experiment, FairDebugger was unable to identify subpopulations with a significant disparity in the average outcome of two groups.

Query results explanation. Extensive research has been devoted to explaining the results of aggregate SQL queries. Multiple works have leveraged *data provenance* to generate explanations for query results [6, 10, 12, 34, 35, 40, 41, 65]. Other methods have explored non-causal interventions [8, 13, 14, 50, 51, 64, 72], entropy-based techniques [15], and identifying counterbalancing patterns [42]. This line of work is different than ours, as our goal is to explain the disparity among two, possibly overleaping, groups of interest via a small set of causal explanations.

Recent works [38, 54, 74, 75] have used causal inference to explain aggregate query results. The authors of [54, 75] proposed methods to find confounding variables that explain the correlation relationship between the grouping attribute and the outcome in group-by-average queries. In both the same explanation is provided for all groups in the query results. The work in [74] focuses on providing causal explanations for group-by-average queries, aiming to explain the overall aggregate view by identifying factors influencing the outcome within each group in the query results. In contrast, our objective is to identify subpopulations within the data where the disparity between two groups is most pronounced and to offer specific causal explanations for the observed disparity within these subpopulations. While our goals differ, we adapt their treatment mining algorithm for our approach, as detailed in Section 4.2. The framework introduced in [38] identifies both causal and non-causal patterns to explain disparities between two groups in aggregate queries. A key distinction is that our approach supports overlapping groups and focuses on identifying specific causal explanations within different subpopulations rather than seeking a single treatment for the entire dataset. We argue that, in many cases, no universal explanation suffices, and disparities are better understood through localized causal insights. This observation is supported by our experiments (see discussed in Section 5.2).

Causality in data management research. Causality has been used in different contexts in data management research [40, 41, 49, 55]. This includes data discovery [20, 25, 28, 57, 76], data cleaning [48, 56], query result explanation [38, 51, 54, 75], hypothetical reasoning [19], and large system diagnostics [4, 26, 39]. We leverage causal inference on observational data [47] to define our disparity explanations, identifying factors that differentially impact the aggregate outcomes of two groups of interest.

7 CONCLUSIONS AND FUTURE WORK

We have presented in this paper, DisEx, a framework for discovering causal explanations for disparities between two groups of interest. DisEx identifies subpopulations where disparities are most pronounced (or reversed), and associates specific factors that causally contribute to the disparity. We acknowledge that the quality of generated disparity explanations can be influenced by several factors, including data quality, the quality of the underlying causal model, and system parameters. Regarding causal DAG quality, prior work [19, 74] shows that meaningful results can still be obtained even with imperfect DAGs. For system parameters, as discussed in Section 5.3, we provide insights on tuning them to achieve satisfactory results across different use cases and datasets.

DisEx currently operates on single-relation databases, assuming no dependencies between tuples to align with the SUTVA assumption [53]. As explained in [74], while treatment and grouping patterns are straightforward in single-table scenarios, extending these concepts to multi-table databases introduces significant challenges. Supporting multi-relation datasets with dependencies among tuples remains an important avenue for future research. Notably, most prior work applying causal inference for explanations [38, 54, 74, 75] has also been limited to single-table databases.

Future work will focus on extending the framework to support comparisons among more than two groups, enabling more comprehensive analyses. Additionally, we aim to incorporate a broader range of aggregation functions beyond the average, such as median, sum, and max, to provide insights across diverse use cases.

REFERENCES

[1] 2021. 2021 Stackoverflow Developer Survey. https://insights.stackoverflow.com/survey/2021.

[2] Agency for Healthcare Research and Quality (AHRQ). 2024. Medical Expenditure Panel Survey (MEPS) - Data Overview. https://meps.ahrq.gov/mepsweb/data_stats/data_overview.jsp Accessed: 2024-01-30.

[3] Shunit Agmon, Amir Gilad, Brit Youngmann, Shahar Zoarets, and Benny Kimelfeld. 2024. Finding Convincing Views to Endorse a Claim. *arXiv preprint arXiv:2408.14974* (2024).

[4] Abdullah Alomar, Pouya Hamadanian, Arash Nasr-Esfahany, Anish Agarwal, Mohammad Alizadeh, and Devavrat Shah. 2023. CausalSim: A Causal Framework for Unbiased Trace-Driven Simulation. In *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*. 1115–1147.

[5] Abolfazl Asudeh, Zhongjun Jin, and HV Jagadish. 2019. Assessing and remedying coverage for a given dataset. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 554–565.

[6] Nicole Bidoit, Melanie Herschel, and Katerina Tzompanaki. 2014. Query-based why-not provenance with nedexplain. In *Extending database technology (EDBT)*.

[7] Tal Blau. 2024. Causal Explanation for Disparity. https://github.com/TalBl/CausalExplanationforDisparity GitHub repository.

[8] Pierre Bourhis, Daniel Deutch, and Yuval Moskovitch. 2020. Equivalence-Invariant Algebraic Provenance for Hyperplane Update Queries. In *Proceedings of the 2020 International Conference on Management of Data, SIGMOD Conference 2020, online conference [Portland, OR, USA], June 14-19, 2020*. ACM, 415–429. https://doi.org/10.1145/3318464.3380578

[9] Ángel Alexander Cabrera, Will Epperson, Fred Hohman, Minsuk Kahng, Jamie Morgenstern, and Duen Horng Chau. 2019. FAIRVIS: Visual Analytics for Discovering Intersectional Bias in Machine Learning. In *14th IEEE Conference on Visual Analytics Science and Technology, IEEE VAST 2019, Vancouver, BC, Canada, October 20-25, 2019*, Remco Chang, Daniel A. Keim, and Ross Maciejewski (Eds.). IEEE, 46–56. https://doi.org/10.1109/VAST47406.2019.8986948

[10] Adriane Chapman and HV Jagadish. 2009. Why not?. In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*. 523–534.

[11] Yeounoh Chung, Tim Kraska, Neoklis Polyzotis, Ki Hyun Tae, and Steven Euijong Whang. 2019. Automated data slicing for model validation: A big data-ai integration approach. *IEEE Transactions on Knowledge and Data Engineering* 32, 12 (2019), 2284–2296.

[12] Daniel Deutch, Nave Frost, and Amir Gilad. 2020. Explaining Natural Language query results. *VLDB J.* 29, 1 (2020), 485–508.

[13] Daniel Deutch, Amir Gilad, Tova Milo, Amit Mualem, and Amit Somech. 2022. FEDEX: An Explainability Framework for Data Exploration Steps. *Proc. VLDB Endow.* 15, 13 (2022), 3854–3868. https://www.vldb.org/pvldb/vol15/p3854-gilad.pdf

[14] Daniel Deutch, Amir Gilad, and Yuval Moskovitch. 2015. Selective Provenance for Datalog Programs Using Top-K Queries. *Proc. VLDB Endow.* 8, 12 (2015), 1394–1405. https://doi.org/10.14778/2824032.2824039

[15] Kareem El Gebaly, Parag Agrawal, Lukasz Golab, Flip Korn, and Divesh Srivastava. 2014. Interpretable and Informative Explanations of Outcomes. *Proc. VLDB Endow.* 8, 1 (sep 2014), 61–72. https://doi.org/10.14778/2735461.2735467

[16] Kareem El Gebaly, Parag Agrawal, Lukasz Golab, Flip Korn, and Divesh Srivastava. 2014. Interpretable and informative explanations of outcomes. *Proceedings of the VLDB Endowment* 8, 1 (2014), 61–72.

[17] Tal Blau et al. 2025. *Causal Explanations for Disparate Trends: Where and Why? (Technical Report)*. Technical Report. https://github.com/TalBl/CausalExplanationforDisparity/blob/main/techreport/disex_tech_report.pdf Technical Report.

[18] Uriel Feige. 1998. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)* 45, 4 (1998), 634–652.

[19] Sainyam Galhotra, Amir Gilad, Sudeepa Roy, and Babak Salimi. 2022. Hyper: Hypothetical reasoning with what-if and how-to queries using a probabilistic causal approach. In *Proceedings of the 2022 International Conference on Management of Data*. 1598–1611.

[20] Sainyam Galhotra, Yue Gong, and Raul Castro Fernandez. 2023. Metam: Goal-oriented data discovery. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2780–2793.

[21] Sainyam Galhotra, Romila Pradhan, and Babak Salimi. 2021. Explaining Black-Box Algorithms Using Probabilistic Contrastive Counterfactuals. In *SIGMOD*. ACM, 577–590.

[22] Floris Geerts, Bart Goethals, and Taneli Mielikäinen. 2004. Tiling databases. In *Discovery Science: 7th International Conference, DS 2004, Padova, Italy, October 2-5, 2004. Proceedings 7*. Springer, 278–289.

[23] Clark Glymour, Kun Zhang, and Peter Spirtes. 2019. Review of causal discovery methods based on graphical models. *Frontiers in genetics* 10 (2019), 524.

[24] Ricky Charles Godbolt. 2011. *Black and Blue: African Americans, Blue-Collar Bias, and the Construction Industry in Prince George's County, Maryland*. Ph. D. Dissertation. University of Phoenix.

[25] Yue Gong, Sainyam Galhotra, and Raul Castro Fernandez. 2024. Nexus: Correlation Discovery over Collections of Spatio-Temporal Tabular Data. *Proceedings of the ACM on Management of Data* 2, 3 (2024), 1–28.

[26] Helga Gudmundsdottir, Babak Salimi, Magdalena Balazinska, Dan RK Ports, and Dan Suciu. 2017. A demonstration of interactive analysis of performance measurements with viska. In *Proceedings of the 2017 ACM International Conference on Management of Data*. 1707–1710.

[27] Paul W Holland. 1986. Statistics and causal inference. *Journal of the American statistical Association* 81, 396 (1986), 945–960.

[28] Zezhou Huang, Jiaxiang Liu, Haonan Wang, and Eugene Wu. 2023. The Fast and the Private: Task-based Dataset Search. *arXiv preprint arXiv:2308.05637* (2023).

[29] Manas Joglekar, Hector Garcia-Molina, and Aditya G. Parameswaran. 2019. Interactive Data Exploration with Smart Drill-Down. *IEEE Trans. Knowl. Data Eng.* 31, 1 (2019), 46–60. https://doi.org/10.1109/TKDE.2017.2685998

[30] Amir-Hossein Karimi, Gilles Barthe, Bernhard Schölkopf, and Isabel Valera. 2023. A Survey of Algorithmic Recourse: Contrastive Explanations and Consequential Recommendations. *Comput. Surveys* 55, 5 (2023), 95:1–95:29.

[31] Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. 2021. Algorithmic Recourse: from Counterfactual Explanations to Interventions. In *FAccT*. ACM, 353–362.

[32] Himabindu Lakkaraju, Stephen H Bach, and Jure Leskovec. 2016. Interpretable decision sets: A joint framework for description and prediction. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 1675–1684.

[33] Laks VS Lakshmanan, Jian Pei, and Jiawei Han. 2002. Quotient cube: How to summarize the semantics of a data cube. In *VLDB'02: Proceedings of the 28th International Conference on Very Large Databases*. Elsevier, 778–789.

[34] Seokki Lee, Bertram Ludäscher, and Boris Glavic. [n. d.]. Approximate Summaries for Why and Why-not Provenance. *Proceedings of the VLDB Endowment* 13, 6 ([n. d.]).

[35] Chenjie Li, Zhengjie Miao, Qitian Zeng, Boris Glavic, and Sudeepa Roy. 2021. Putting Things into Context: Rich Explanations for Query Answers using Join Graphs. In *Proceedings of the 2021 International Conference on Management of Data*. 1051–1063.

[36] Jinyang Li, Yuval Moskovitch, and H. V. Jagadish. 2023. Detection of Groups with Biased Representation in Ranking. In *39th IEEE International Conference on Data Engineering, ICDE 2023, Anaheim, CA, USA, April 3-7, 2023*. IEEE, 2167–2179. https://doi.org/10.1109/ICDE55515.2023.00168

[37] Yin Lin, Brit Youngmann, Yuval Moskovitch, HV Jagadish, and Tova Milo. 2021. On detecting cherry-picked generalizations. *Proceedings of the VLDB Endowment* 15, 1 (2021), 59–71.

[38] Pingchuan Ma, Rui Ding, Shuai Wang, Shi Han, and Dongmei Zhang. 2023. XInsight: EXplainable Data Analysis Through The Lens of Causality. *Proc. ACM Manag. Data*, Article 156 (jun 2023), 27 pages.

[39] Markos Markakis, An Bo Chen, Brit Youngmann, Trinity Gao, Ziyu Zhang, Rana Shahout, Peter Baile Chen, Chunwei Liu, Ibrahim Sabek, and Michael Cafarella. 2024. Sawmill: From Logs to Causal Diagnosis of Large Systems. In *SIGMOD*. 444–447.

[40] Alexandra Meliou, Wolfgang Gatterbauer, Katherine F Moore, and Dan Suciu. 2009. Why so? or why no? functional causality for explaining query answers. *arXiv preprint arXiv:0912.5340* (2009).

[41] Alexandra Meliou, Wolfgang Gatterbauer, Katherine F Moore, and Dan Suciu. 2010. The Complexity of Causality and Responsibility for Query Answers and non-Answers. *Proceedings of the VLDB Endowment* 4, 1 (2010).

[42] Zhengjie Miao, Qitian Zeng, Boris Glavic, and Sudeepa Roy. 2019. Going beyond provenance: Explaining query answers with pattern-based counterbalances. In *Proceedings of the 2019 International Conference on Management of Data*. 485–502.

[43] Yuval Moskovitch, Jinyang Li, and H. V. Jagadish. 2023. Dexer: Detecting and Explaining Biased Representation in Ranking. In *Companion of the 2023 International Conference on Management of Data, SIGMOD/PODS 2023, Seattle, WA, USA, June 18-23, 2023*. ACM, 159–162. https://doi.org/10.1145/3555041.3589725

[44] Zafeiria Moumoulidou, Andrew McGregor, and Alexandra Meliou. 2021. Diverse Data Selection under Fairness Constraints. In *24th International Conference on Database Theory, ICDT 2021, March 23-26, 2021, Nicosia, Cyprus (LIPIcs, Vol. 186)*, Ke Yi and Zhewei Wei (Eds.). Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 13:1–13:25. https://doi.org/10.4230/LIPICS.ICDT.2021.13

[45] George L Nemhauser, Laurence A Wolsey, and Marshall L Fisher. 1978. An analysis of approximations for maximizing submodular set functions—I. *Mathematical programming* 14 (1978), 265–294.

[46] Eliana Pastor, Luca De Alfaro, and Elena Baralis. 2021. Looking for trouble: Analyzing classifier behavior via pattern divergence. In *Proceedings of the 2021 International Conference on Management of Data*. 1400–1412.

[47] Judea Pearl. 2009. Causal inference in statistics: An overview. (2009).

[48] Alireza Pirhadi, Mohammad Hossein Moslemi, Alexander Cloninger, Mostafa Milani, and Babak Salimi. 2024. Otclean: Data cleaning for conditional independence violations using optimal transport. *Proceedings of the ACM on Management of Data* 2, 3 (2024), 1–26.

[49] Sudeepa Roy. 2022. Toward interpretable and actionable data analysis with explanations and causality. *Proc. VLDB Endow.* 15, 12 (2022), 3812–3820.

[50] Sudeepa Roy, Laurel Orr, and Dan Suciu. 2015. Explaining query answers with explanation-ready databases. *Proceedings of the VLDB Endowment* 9, 4 (2015), 348–359.

[51] Sudeepa Roy and Dan Suciu. 2014. A formal approach to finding explanations for database queries. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*. 1579–1590.

[52] Donald Bruce Rubin. 1971. *The use of matched sampling and regression adjustment in observational studies*. Ph. D. Dissertation. Harvard University.

[53] Donald B Rubin. 2005. Causal inference using potential outcomes: Design, modeling, decisions. *J. Amer. Statist. Assoc.* 100, 469 (2005), 322–331.

[54] Babak Salimi, Johannes Gehrke, and Dan Suciu. 2018. Bias in olap queries: Detection, explanation, and removal. In *Proceedings of the 2018 International Conference on Management of Data*. 1021–1035.

[55] Babak Salimi, Harsh Parikh, Moe Kayali, Lise Getoor, Sudeepa Roy, and Dan Suciu. 2020. Causal relational learning. In *Proceedings of the 2020 ACM SIGMOD international conference on management of data*. 241–256.

[56] Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. 2019. Interventional fairness: Causal database repair for algorithmic fairness. In *Proceedings of the 2019 International Conference on Management of Data*. 793–810.

[57] Aécio Santos, Aline Bessa, Fernando Chirigati, Christopher Musco, and Juliana Freire. 2021. Correlation sketches for approximate join-correlation queries. In *Proceedings of the 2021 International Conference on Management of Data*. 1531–1544.

[58] Sunita Sarawagi. 2000. User-adaptive exploration of multidimensional data. In *VLDB*. ResearchGate GmbH, 307–316.

[59] Sunita Sarawagi. 2001. User-cognizant multidimensional analysis. *The VLDB Journal* 10 (2001), 224–239.

[60] Sunita Sarawagi, Rakesh Agrawal, and Nimrod Megiddo. 1998. Discovery-driven exploration of OLAP data cubes. In *Advances in Database Technology—EDBT'98: 6th International Conference on Extending Database Technology Valencia, Spain, March 23–27, 1998 Proceedings 6*. Springer, 168–182.

[61] Gayatri Sathe and Sunita Sarawagi. 2001. Intelligent rollups in multidimensional OLAP data. In *VLDB*. 307–316.

[62] Amit Sharma and Emre Kiciman. 2020. DoWhy: An End-to-End Library for Causal Inference. *arXiv preprint arXiv:2011.04216* (2020).

[63] Tanmay Surve and Romila Pradhan. 2024. Example-based Explanations for Random Forests using Machine Unlearning. *CoRR* abs/2402.05007 (2024).

[64] Yuchao Tao, Amir Gilad, Ashwin Machanavajjhala, and Sudeepa Roy. 2022. DPXPlain: Privately Explaining Aggregate Query Answers. *Proc. VLDB Endow.* 16, 1 (2022), 113–126. https://www.vldb.org/pvldb/vol16/p113-tao.pdf

[65] Balder ten Cate, Cristina Civili, Evgeny Sherkhonov, and Wang-Chiew Tan. 2015. High-level why-not explanations using ontologies. In *Proceedings of the 34th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*. 31–43.

[66] U.S. Census Bureau. 2024. American Community Survey (ACS) - Data. https://www.census.gov/programs-surveys/acs/data.html Accessed: 2024-01-30.

[67] Moshe Y. Vardi. 1982. The Complexity of Relational Query Languages (Extended Abstract). In *Proceedings of the Fourteenth Annual ACM Symposium on Theory of Computing* (San Francisco, California, USA) *(STOC '82)*. ACM, New York, NY, USA, 137–146. https://doi.org/10.1145/800070.802186

[68] Manasi Vartak, Sajjadur Rahman, Samuel Madden, Aditya Parameswaran, and Neoklis Polyzotis. 2015. Seedb: Efficient data-driven visualization recommendations to support visual analytics. In *VLDB*, Vol. 8. NIH Public Access, 2182.

[69] Clifford H Wagner. 1982. Simpson's paradox in real life. *The American Statistician* 36, 1 (1982), 46–48.

[70] Yue Wang, Alexandra Meliou, and Gerome Miklau. 2018. RC-Index: Diversifying Answers to Range Queries. *Proc. VLDB Endow.* 11, 7 (2018), 773–786. https://doi.org/10.14778/3192965.3192969

[71] Yuhao Wen, Xiaodan Zhu, Sudeepa Roy, and Jun Yang. 2018. Interactive summarization and exploration of top aggregate query answers. In *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*, Vol. 11. NIH Public Access, 2196.

[72] Eugene Wu and Samuel Madden. 2013. Scorpion: Explaining away outliers in aggregate queries. (2013).

[73] Brit Youngmann, Sihem Amer-Yahia, and Aurélien Personnaz. 2022. Guided Exploration of Data Summaries. *Proc. VLDB Endow.* 15, 9 (2022).

[74] Brit Youngmann, Michael Cafarella, Amir Gilad, and Sudeepa Roy. 2024. Summarized Causal Explanations For Aggregate Views. *Proceedings of the ACM on Management of Data* 2, 1 (2024), 1–27.

[75] Brit Youngmann, Michael Cafarella, Yuval Moskovitch, and Babak Salimi. 2023. On Explaining Confounding Bias. *2023 IEEE 39th International Conference on Data Engineering (ICDE)* (2023).

[76] Brit Youngmann, Michael Cafarella, Babak Salimi, and Anna Zeng. 2023. Causal Data Integration. *Proceedings of the VLDB Endowment* 16, 10 (2023), 2659–2665.

[77] Cong Yu, Laks Lakshmanan, and Sihem Amer-Yahia. 2009. It takes variety to make a world: diversification in recommender systems. In *Proceedings of the 12th international conference on extending database technology: Advances in database technology*. 368–378.

[78] Xiaozhong Zhang, Xiaoyu Ge, Panos K Chrysanthis, and Mohamed A Sharaf. 2021. Viewseeker: An interactive view recommendation framework. *Big Data Research* 25 (2021), 100238.