

XINSIGHT: eXplainable Data Analysis Through The Lens of Causality

PINGCHUAN MA, Hong Kong University of Science and Technology, Hong Kong SAR

RUI DING*, Microsoft Research, China

SHUAI WANG, Hong Kong University of Science and Technology, Hong Kong SAR

SHI HAN, Microsoft Research, China

DONGMEI ZHANG, Microsoft Research, China

In light of the growing popularity of Exploratory Data Analysis (EDA), understanding the underlying causes of the knowledge acquired by EDA is crucial. However, it remains under-researched. This study promotes a transparent and explicable perspective on data analysis, called *eXplainable Data Analysis* (XDA). For this reason, we present XINSIGHT, a general framework for XDA. XINSIGHT provides data analysis with qualitative and quantitative explanations of causal and non-causal semantics. This way, it will significantly improve human understanding and confidence in the outcomes of data analysis, facilitating accurate data interpretation and decision making in the real world. XINSIGHT is a three-module, end-to-end pipeline designed to extract causal graphs, translate causal primitives into XDA semantics, and quantify the quantitative contribution of each explanation to a data fact. XINSIGHT uses a set of design concepts and optimizations to address the inherent difficulties associated with integrating causality into XDA. Experiments on synthetic and real-world datasets as well as a user study demonstrate the highly promising capabilities of XINSIGHT.

CCS Concepts: • **Information systems** → **Data analytics**; **Data management systems**; • **Mathematics of computing** → **Bayesian networks**.

ACM Reference Format:

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* 1, 2, Article 156 (June 2023), 26 pages. <https://doi.org/10.1145/3589301>

1 INTRODUCTION

Exploratory data analysis (EDA) is key to acquiring insight from data and facilitating analysis towards decision making [Ma, Ding, Han, et al. 2021; Milo and Somech 2020]. With the advent of the digital age, the information explosion phenomenon [Buckland 2017] makes it difficult for users to justify and rely on knowledge and conclusions from EDA. To ease the cognitive process, data explanations are proposed to deliberate data facts (e.g., query outcomes) and enhance user comprehension [Glavic et al. 2021]. In this paper, we term such a process as *eXplainable Data Analysis* (XDA), which advances data analysis by providing users with effective explanations.

*Corresponding author.

Authors' addresses: Pingchuan Ma, Hong Kong University of Science and Technology, Kowloon, Hong Kong SAR, pmaa b@cse.ust.hk; Rui Ding, Microsoft Research, Beijing, China, juding@microsoft.com; Shuai Wang, Hong Kong University of Science and Technology, Kowloon, Hong Kong SAR, shuaiw@cse.ust.hk; Shi Han, Microsoft Research, Beijing, China; Dongmei Zhang, Microsoft Research, Beijing, China.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.

2836-6573/2023/6-ART156 \$15.00

<https://doi.org/10.1145/3589301>

By suggesting and justifying choices to alter outcomes, XDA helps users comprehend and trust phenomena emerging from data; as a result, it facilitates real-world decision making.

Explanations can be categorized as either causal or non-causal [Lange 2016]. Causal explanations seek causal factors to explain an outcome. Fig. 1 depicts a hypothetical lung cancer dataset. Here, a patient’s location (indicating regional tobacco control policy) and amount of stress have an impact on whether they would smoke. Then, smoking influences lung cancer’s severity. The degree of severity further affects whether they would undergo surgery and the five-year survival rate. Here, smoking explains why a patient has high lung cancer severity (see Fig. 1(f)). In contrast, a non-causal explanation shows the results merely by statistical correlations. For example, surgery “explains” (more precisely, is relevant to) lung cancer severity (see Fig. 1(g)). Despite being helpful, this is not a causal explanation [Povich and Craver 2018].

Existing data explanation tools (e.g., Tableau’s Explain Data [Discover Insights Faster with Explain Data 2022] in industry, Scorpion [E. Wu and Madden 2013] and DIFF [Abuzaid et al. 2021] in academia) often provide non-causal explanations [Glavic et al. 2021]. Although valuable for data analysis, they may mislead users who want causal explanations. A well-known confusion, as noted in [Law et al. 2021], is that Tableau’s Explain Data reports that Massachusetts’ low teenage pregnancy rate may explain this state’s high ACT Math score. Such explanations are questionable. In comparison, causal explanations play a central role in human cognition [Keil 2006; Murphy and Medin 1985]. They enable users to make counterfactual thinking and actionable decisions. For instance, quitting smoking reduces lung cancer severity whereas cancelling surgery does not.

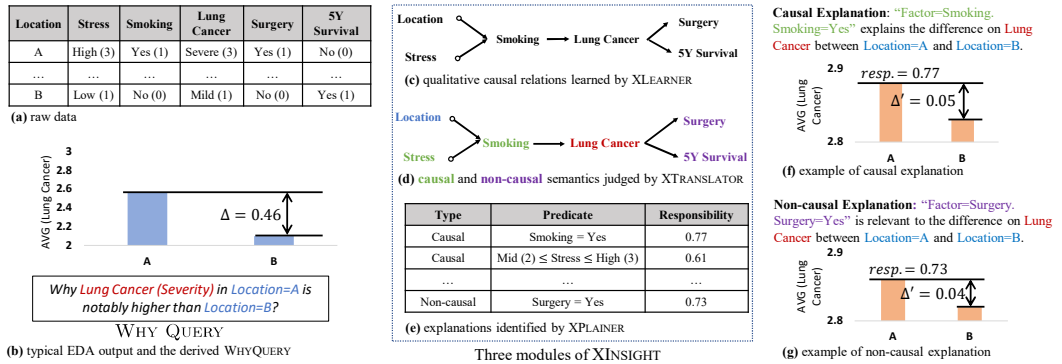


Fig. 1. Illustrative example of XINSIGHT.

According to Pearl’s general causality [Pearl 2009a], causal knowledge is typically represented by causal graphs. Each node in a causal graph represents a random variable in data, and each directed edge between parent and child nodes denotes a cause-effect relation. Causal knowledge primarily conveys *qualitative explanations* [Scheines et al. 1991], such as smoking causes lung cancer. To further enable *quantitative explanations*, it is necessary to quantify the contribution of each input to the output. This way, we quantify smoking’s contribution to lung cancer and compare it with other factors. Halpern’s actual causality [Halpern 2016; Halpern and Pearl 2005], and essentially its adaptation, DB Causality [30], provide an elegant formulation of this concept.

This paper proposes XINSIGHT as a unified, causality-based XDA framework that qualitatively and quantitatively answers WHY QUERY raised by users. Considering the following WHY QUERY:

Example 1.1: The dataset in Fig. 1(a) depicts the patient information in a country. An analyst observes an interesting data fact: “the average severity of lung cancer of patients in location A is much higher than in location B” and then raises a WHY QUERY (Fig. 1(b)).

for which Fig. 1(f) and (g) illustrate two explanations provided by XINSIGHT. Each explanation is flagged as either a “causal” or “non-causal” explanation, and is composed of both qualitative and quantitative sub-explanations.

Example 1.2: An explanation (Fig. 1(f)) deems “Smoking” to be a qualitative causal factor of lung cancer severity and highlights “Smoking=Yes” and its responsibility as a quantitative sub-explanation.

XINSIGHT includes three modules, XLEARNER, XTRANSLATOR, and XPLAINER to gradually form explanations. XLEARNER first automatically discovers a causal graph \mathcal{G} from data (Fig. 1(c)). Then, given a WHY QUERY (Fig. 1(b)) with the target (i.e., the measure “Lung Cancer”) and the context (i.e., the breakdown dimension “Location”), XTRANSLATOR enumerates each remaining variable on \mathcal{G} and decides if it is causal or non-causal to the target under the context. Fig. 1(d) shows causal variables (i.e., those that can potentially provide causal explanations) in green and non-causal variables (i.e., those that can potentially provide non-causal explanations) in purple. Last, XPLAINER quantifies how well each variable answers WHY QUERY by searching possible predicates on the variables that are the most responsible, as shown in Fig. 1(e). Despite the promising capability of XINSIGHT, concretizing each module is challenging. We brief the challenges and our solutions in the following.

XLEARNER. Most real-world datasets are collected irrespective of causal sufficiency [Peters, Janzing, and Schölkopf 2017]. In other words, not all causally relevant variables are available in the dataset. Furthermore, real-world datasets often contain deterministic relations in the form of Functional Dependency (FD), especially when they have materialized from relational databases. These FDs may violate the faithfulness assumption [Ding, Liu, et al. 2020], which is crucial for many causal discovery algorithms. To address these challenges, we establish a theory to propose an FD-induced graph \mathcal{G}_{FD} . XLEARNER uses \mathcal{G}_{FD} to select a subset of variables for standard causal discovery where the selected variables do not trigger faithfulness violations induced by FDs. It adopts FCI [J. Zhang 2008] to address causal insufficiency and synergistically combine the result of FCI with the causal relations entailed by \mathcal{G}_{FD} .

XTRANSLATOR. The translation from causal primitives (the structural relations in the causal graph) into XDA semantics (e.g., whether a variable provides causal explanations) is under-explored. Given a WHY QUERY (with a target and a context), it is unclear how to determine if a variable X can explain the target given the context, and, moreover, if X provides causal or non-causal explanations. XTRANSLATOR characterizes various causal primitives (e.g., m -separation, ancestor/descendant relations) from a causal graph and provides a taxonomy to translate them into XDA semantics.

XPLAINER. DB causality is primarily designed for data provenance, which usually provides tuples as explanations. Contrarily, we note that predicate-level explanations shall be more desirable for XDA scenarios. Moreover, computing the responsibility of explanations with DB causality is NP-complete in general [Meliou, Gatterbauer, Moore, et al. 2010]. XPLAINER adapts DB causality to XDA by using predicate-level explanations with the conciseness consideration and also significantly reduces the computing cost with theoretical guarantees. In summary, we make the following contributions:

- We propose XINSIGHT, a unified and causality-based framework for XDA. XInsight features adequate (by distinguishing causal from non-causal) and comprehensive (with qualitative and quantitative) explanations.
- XINSIGHT consists of three modules, XLEARNER, XTRANSLATOR and XPLAINER, each of which is meticulously designed to address technical challenges and deliver efficient analysis. XLEARNER learns the causal graph from causally insufficient data in the presence of FD-induced faithfulness

violations, XTRANSLATOR translates causal primitives into XDA semantics, and XPLAINER efficiently provides quantitative explanations via an adaptation of DB causality to meet the needs of XDA scenarios.

- Empirically, we conduct thorough experiments on public data, production data, and synthetic data via quantitative experiments and human evaluations. The results are very encouraging.

Open Source and Real-world Adoption. We release our code at [Microsoft 2022a]. XPLAINER has been integrated into Microsoft Power BI to explain increase/decrease in data [Microsoft 2022b].

2 PRELIMINARY

2.1 Data Model and Query

Multi-Dimensional Data. Let $D := \{X_1, \dots, X_n\}$ represents multi-dimensional data comprising n attributes. In XINSIGHT, we assume that records of D are drawn independently from an identical distribution without selection biases (i.e, i.i.d. assumption) such that each attribute is a (random) variable. Here, selection bias is a preferential selection of units in data analysis [Bareinboim and Pearl 2012]. A variable is either categorical or numerical. In accordance with previous works [Ding, Han, et al. 2019; Ma, Ding, Han, et al. 2021], we denote a categorical variable as *dimension* and a numerical variable as *measure*. Multi-dimensional data is commonly represented as a spreadsheet in our context. For relational data, we anticipate taking a materialized provenance table [C. Li et al. 2021] as input.

Aggregation and Discretization on Measure. Given a measure M , users may perform aggregation operations (such as SUM and AVG in SQL) over a set of realizations of M . In some cases, measures are processed in the form of a dimension (e.g., use measures for explanations), which necessitates discretization. It transforms numerical values into several discrete bins that form a categorical variable.

Filter. In this paper, filter is the basic unit of data operations. Given a multi-dimensional data D and a dimension X , a filter $p_i = \{X = x_i\}$ (e.g., “Smoking=Yes”) implies an equality assertion to X such that the value of X shall equal x_i .

Predicate. The disjunction of filters applied on the same dimension is a predicate. Given the dimension X , the predicate $P(x_1, \dots, x_k)$ is a set containment assertion $\{X = x_1 \vee \dots \vee X = x_k\} \equiv \{p_1, \dots, p_k\}$. On a discretized measure, a predicate is an assertion on ranges. A filter is a special case of a predicate. For clarity, we represent a general predicate with a capital P and a filter with a lower-case p .

Subspace. A subspace is a conjunction of filters on disjointed dimensions. Given multi-dimensional data D , a subspace corresponds to a subset of rows satisfying the conditions of all filters. If two subspaces only differ in one filter, they are regarded as *siblings*. The term *Context* refers to the variables of two sibling subspaces, where the *background variables* are the variables with the shared filters and the *foreground variables* are the variables with the different filters. In the following example, we provide a simple instantiation.

Example 2.1: Consider the preceding dataset in Fig. 1(a). $s = \{\text{Location} = A \wedge \text{Lung Cancer} = \text{Severe}\}$ represents the subspace denoting all patients in “Location=A” with severe lung cancer. All patients in “Location=A” with severe lung cancer and all patients in “Location=B” with severe lung cancer form a pair of sibling subspaces. Here, “Location” is the foreground variable and “Lung Cancer” is the background variable.

Selection. We use the following notation to represent the selection procedure over multi-dimensional data D . The subset of data after the selection operation is defined as D_{p_i} , D_P , or D_s , where p_i is a filter, P is a predicate, and s is a subspace. We define $D - D'$ as the rows remaining in D after removing those from D' .

WHY QUERY and Explanation. As illustrated in Fig. 1(b), the user would issue a WHY QUERY to XINSIGHT for explanation. We formally define WHY QUERY as follows.

DEFINITION 2.1 (WHY QUERY). *Given a multi-dimensional data D , a user launches aggregate query $agg()$ on a target measure M under two sibling subspaces s_1, s_2 . WHY QUERY is defined as $\Delta_{s_1, s_2, M, agg}(D) = agg_M(D_{s_1}) - agg_M(D_{s_2})$. For brevity, we use $\Delta(D)$ as the shorthand of $\Delta_{s_1, s_2, M, agg}(D)$. W.l.o.g., we assume Δ is always non-negative.*

Example 2.2: As shown in Fig. 1(b), we concretize the WHY QUERY Δ with the AVG aggregate on the target “Lung Cancer” over two sibling subspaces $s_1 = \{\text{Location} = A\}$, $s_2 = \{\text{Location} = B\}$, denoting the difference in average lung cancer severity in “A” and “B”.

Indeed, explaining the difference between two aggregate queries is one prevalent data analysis task. Identifying the cause in data difference constitutes the basis of many data explanation applications, such as outlier explanation and data debugging [Glavic et al. 2021]. In accordance with prior works [Abuzaid et al. 2021; Glavic et al. 2021; E. Wu and Madden 2013], we concretize the problem of XDA by concentrating on the explanation of *data difference*. The following form is used to provide explanations in response to WHY QUERY.

DEFINITION 2.2 (EXPLANATION). *Given a WHY QUERY, an explanation is represented by the following triplet*

$$\text{explanation} := \langle \text{type}, \text{predicate}, \text{responsibility} \rangle \quad (1)$$

where $\text{type} \in \{\text{causal}, \text{non-causal}\}$ denotes whether the explanation is causal or non-causal, the predicate is the content of the explanation, and responsibility, and a score ranging from 0 to 1 quantifies the extent to which the explanation explains the given WHY QUERY.

Example 2.3: Fig. 1(e) lists several explanations to the WHY QUERY. Fig. 1(f)-(g) visualize two of them. Fig. 1(f), as a causal explanation, depicts that “Smoking=Yes” causes the lung cancer severity difference in Location A and B with a responsibility of 0.77.

Single- vs. Multi-Dimensional Explanation. For conciseness and clarity, we anticipate that each explanation reflects one aspect contributing to the outcome when explaining the WHY QUERY. We recommend adopting a single-dimensional explanation in XINSIGHT due to its unambiguous causal semantics, although it is feasible to extend an explanation as multi-dimensional using the Cartesian product. The joint causal semantics of several variables, however, could be obscure. Furthermore, multiple single-dimensional explanations (e.g., Fig. 1(e)) suffice to represent a multi-dimensional case.

Functional Dependency (FD). Functional dependency relations are common in multi-dimensional data. In a relational database, among the attributes, there may exist primary keys and foreign keys. Therefore, after materialization, the resulting multi-dimensional data may have functional dependencies. A functional dependency between X and Y is represented by $X \xrightarrow{\text{FD}} Y$. FD, as a deterministic relation among two variables, deems a form of reliable knowledge. This research focuses on one-to-one and one-to-many FDs. We present a simple exemplary dataset that contains FDs.

Example 2.4: Let **CityInfo** be a dataset with three attributes (i.e., City, State, Country). It has three FDs, namely, $\text{City} \xrightarrow{\text{FD}} \text{State}$, $\text{State} \xrightarrow{\text{FD}} \text{Country}$, and $\text{City} \xrightarrow{\text{FD}} \text{Country}$.

FD-Induced Graph. Given a multi-dimensional data D and its functional dependencies, the FD-induced Graph $\mathcal{G}_{\text{FD}} := (V, E)$, where $V := \{X_i \mid \forall X_i \in D\}$ and $E := \{(X_i, X_j) \mid \text{if } X_i \xrightarrow{\text{FD}} X_j\}$. We assume \mathcal{G}_{FD} to be acyclic. Cycles in \mathcal{G}_{FD} imply redundant attributes; in such cases, we retain only one of them to ensure acyclicity.

2.2 Causal Discovery with Latent Variables

This section presents terminology essential to causal discovery with latent variables, such as the representation of causal graphs under causal insufficiency and typical assumptions in causal discovery.

Causal Sufficiency. Causal discovery aims to learn the causal relations from the observational data. Most causal discovery algorithms assume a sufficient observation of the underlying data generating process [Spirtes et al. 2000]. Formally, a set of variables X is said to be causally sufficient if there is no hidden variable $Z \notin X$ that is causing more than one variable in X . In other words, it assumes that latent confounders — the shared causes among two or multiple variables — do not exist. However, the process used to acquire real-world data does not provide such guarantees, thereby often yielding causally insufficient observations. Hence, the causal discovery procedure is compromised by the spurious association between two variables sharing a latent confounder. We present an example below.

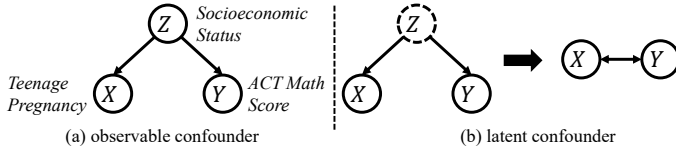


Fig. 2. Examples of observable and latent confounders.

Example 2.5: Consider the hypothetical causal graph in Fig. 2(a) where the socioeconomic status (Z) simultaneously causes teenage pregnancy (X) and their ACT math scores (Y). The socioeconomic status, however, does not appear in the dataset. This absence yields an insufficient observation (the left-side causal graph in Fig. 2(b)), which further results in a spurious association [Pearl 2009b] of teenage pregnancy and ACT math scores (the bidirected edge in Fig. 2(b)).

Hence, popular directed acyclic graphs are not expressive enough to represent these subtle relations. This necessitates the Maximum Ancestral Graph [Spirtes et al. 2000], which is introduced shortly.

Notation and Terminology. Recall that we assume the dataset is *i.i.d.* with potential latent confounders and does not contain selection bias. Maximal Ancestral Graph (MAG) forms the standard representation of causal graphs in this setting. We now introduce important concepts of graphical models and properties of MAG.

A directed mixed graph \mathcal{G} is a graphical model that contains nodes X and two types of edges, including directed (\rightarrow) and bidirected (\leftrightarrow). There is at most one edge between any two nodes. For each directed edge $X \rightarrow Y$, X is a *parent* of Y and Y is a *child* of X . X and Y are *adjacent* if there is an edge (either directed or bidirected) between them. A *path* \mathcal{P} is a sequence of distinct nodes (X_1, \dots, X_k) where X_i and X_{i+1} are adjacent in \mathcal{G} for all $1 \leq i < k$. A path $\mathcal{P} = (X_1, \dots, X_k)$ is directed if X_i is a parent of X_{i+1} for all $1 \leq i < k$. X is an ancestor of Y if there exists a directed path from X to Y and Y is a descendant of X accordingly. Given a path (X_1, \dots, X_k) , a non-endpoint node X_i is a collider if there are arrowheads pointing to X_i from both X_{i-1} and X_{i+1} . Below, we list all possible cases of a collider.

Example 2.6: Given (X_{i-1}, X_i, X_{i+1}) , X_i is a collider if and only if a) $X_{i-1} \rightarrow X_i \leftarrow X_{i+1}$, or b) $X_{i-1} \leftrightarrow X_i \leftarrow X_{i+1}$, or c) $X_{i-1} \rightarrow X_i \leftrightarrow X_{i+1}$, or d) $X_{i-1} \leftrightarrow X_i \leftrightarrow X_{i+1}$. In Fig. 1(c), Smoking is a collider of Location and Stress since “Location $\circ \rightarrow$ Smoking $\leftarrow \circ$ Stress”, where \circ represents an undetermined edge endpoint.

A path (X, W_1, \dots, W_k, Y) is said to be *blocked* by $Z \subseteq X \setminus \{X, Y\}$ if there exists a node $W_i \in \{W_1, \dots, W_k\}$ such that a) W_i is not a collider but a member of Z , or b) W_i is a collider but not an ancestor of any nodes of Z . We now introduce *m-separation* and MAG.

DEFINITION 2.3 (M-SEPARATION [J. ZHANG 2008]). X, Y are m -separated by Z (denoted by $X \perp\!\!\!\perp_{\mathcal{G}} Y \mid Z$) if all paths between X, Y are blocked by Z .

Example 2.7: Consider the causal graph in Fig. 1(c) where “Smoking” blocks the only path between “Location” and “Lung Cancer”. Hence, “Smoking” m -separates “Location” and “Lung Cancer” (denoted by $\text{Lung Cancer} \perp\!\!\!\perp_{\mathcal{G}} \text{Location} \mid \text{Smoking}$).

DEFINITION 2.4 (MAXIMAL ANCESTRAL GRAPH [J. ZHANG 2008]). A directed mixed graph is called a MAG if a) it contains no directed cycles or almost directed cycles and b) for each pair of non-adjacent nodes, there exists a set of nodes that m -separates them. A directed cycle refers to the case where $X \rightarrow Y \rightarrow \dots \rightarrow X$ and an almost directed cycle refers to the case where $X \rightarrow Y \rightarrow \dots \rightarrow Z \leftrightarrow X$.

Then, the Global Markov Property (GMP) is developed to provide a probabilistic interpretation of m -separation.

DEFINITION 2.5 (GLOBAL MARKOV PROPERTY [SPIRITES ET AL. 2000]).

$$X \perp\!\!\!\perp_{\mathcal{G}} Y \mid Z \Rightarrow X \perp\!\!\!\perp Y \mid Z \quad (2)$$

As aforementioned, m -separation indicates that all paths between X and Y are “blocked” by Z . Hence, it is intuitive that, if X and Y are m -separated, their statistical correlation is also “blocked” by Z . The term *conditional independence* (i.e., $X \perp\!\!\!\perp Y \mid Z$) depicts this absence of statistical correlation. Statistically, $X \perp\!\!\!\perp Y \mid Z$ implies that $P(X, Y \mid Z) = P(X \mid Z)P(Y \mid Z)$, which can be empirically examined using statistical hypothesis tests (e.g., χ^2 tests).

Example 2.8: Consider the dataset in Fig. 1(a). According to GMP, the m -separation in Ex. 2.7 implies that, for the dataset in Fig. 1(a), “Location” and “Lung Cancer” are conditionally independent given “Smoking” in a statistical sense.

With GMP, we can deduce statistical conditional independence in data from m -separations. Note that only data is available when performing causal discovery. Hence, we need to invert GMP and establish a connection from data distribution to the graphical structure. Faithfulness assumption establishes such connection.

DEFINITION 2.6 (FAITHFULNESS [SPIRITES ET AL. 2000]).

$$X \perp\!\!\!\perp Y \mid Z \Rightarrow X \perp\!\!\!\perp_{\mathcal{G}} Y \mid Z \quad (3)$$

According to faithfulness, if we observe that two variables are conditionally independent by a set of variables in data, then they are m -separated by the same set of variables on the causal graph. Faithfulness and GMP together establish the equivalence between conditional independence and m -separation and they form the key to causal discovery. In addition, we define *skeleton* as follows.

DEFINITION 2.7 (SKELETON). The skeleton \mathcal{S} of a MAG \mathcal{G} is the undirected graph obtained by removing all arrowheads from \mathcal{G} .

Constraint-based Causal Discovery. Constraint-based approaches are the standard solution to causal discovery. With the faithfulness assumption, these methods exploit the conditional independence derived from data and gradually establish a MAG \mathcal{G} . \mathcal{G} is consistent with all m -separations entailed by conditional independence. However, there may exist multiple MAGs that are equally consistent with the m -separations and not distinguishable, which is called Markov equivalence class, denoted by $[\mathcal{G}]$. It is worth noting that these feasible MAGs share the same skeleton while differing in direction on certain edges. These MAGs are therefore summarized into a compact representation called Partial Ancestral Graph (PAG) with some undetermined edge endpoints.

Table 1. Four types of edges in PAG. Circle represents undetermined edge endpoint (can be either an arrowhead or tail).

Edge	Causal Semantics
$X \rightarrow Y$	X is a cause of Y .
$X \leftrightarrow Y$	neither X nor Y is a cause of each other but they share a latent common cause.
$X \circ \rightarrow Y$	1) X is a cause of Y ; or 2) neither X nor Y is a cause of each other but they share a latent common cause.
$X \circ \circ Y$	1) X may be a cause of Y ; or 2) Y may be a cause of X ; or 3) neither X nor Y is a cause of each other but they share a latent common cause.

DEFINITION 2.8 (PARTIAL ANCESTRAL GRAPH [J. ZHANG 2008]). Let $[\mathcal{G}]$ be a Markov equivalence class of a MAG \mathcal{G} . A PAG for $[\mathcal{G}]$ is a graph \mathcal{P} with three possible edge endpoints (namely, tail, circle and arrowhead; and hence four kinds of edges: \rightarrow , \leftrightarrow , $\circ \rightarrow$, $\circ \circ$) such that 1) \mathcal{P} shares the same adjacencies with \mathcal{G} (and any member of $[\mathcal{G}]$), and 2) every non-circle edge endpoint indicates an invariant edge endpoint in $[\mathcal{G}]$.

The second condition in Def. 2.8 implies that an edge associates a tail “ \leftarrow ” or arrowhead “ \rightarrow ” endpoint, if and only if it is invariant in all $\mathcal{G} \in [\mathcal{G}]$. Table 1 lists the semantics of edges.

Example 2.9: Location $\circ \rightarrow$ Smoking in Fig. 1 (c) implies that “Location” is a cause of “Smoking” or they share a latent confounder.

We clarify that the FCI algorithm [Spirtes et al. 2000], as a typical constraint-based approach, consists of two phases. The skeleton of $[\mathcal{G}]$ is first learned by assuming faithfulness (i.e., the FCI-SL phase of the FCI algorithm). Then, the undirected edges are subsequently oriented according to a set of orientation rules (i.e., the FCI-Orient phase of the FCI algorithm). Finally, the PAG is returned; see full details of the FCI algorithm in Supplementary Material. However, soon we will show that the faithfulness assumption can be violated by FD relations. In this paper, we focus on establishing a theory and proposing a solution to tackle this unique challenge that arises in data analysis scenarios. That is, our XLEARNER calibrates the FCI algorithm to correctly handling FDs (see details in Sec. 3.1).

3 XINSIGHT

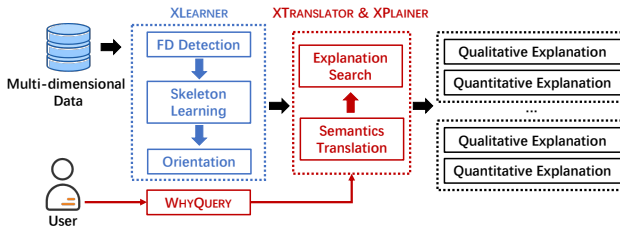


Fig. 3. Workflow of XINSIGHT. Offline phase is marked in blue and online phase is marked in red.

XINSIGHT delivers a unified framework for XDA with three modules. The workflow of XINSIGHT is shown in Fig. 3. First, given a multi-dimensional data D , XLEARNER pre-learns a causal graph \mathcal{G} from data in the offline phase (blue-annotated in Fig. 3). Then, in the online phase (red-annotated in Fig. 3), upon receiving a WHY QUERY, XTRANSLATOR identifies variables that have the potential to give either causal or non-causal explanations based on the causal primitives in \mathcal{G} . Finally, XPLAINER examines each identified variable with potential and decides the optimal explanation for the given WHY QUERY. After applying XPLAINER to all variables with potential, XINSIGHT yields a set of explanations (with qualitative sub-explanations and quantitative sub-explanations). By decoupling XINSIGHT into an offline phase and an online phase, heavy-weight computations are performed beforehand, and only light-weight computations are needed in the online phase, allowing for a

rapid response to a user’s query. In the following, we elaborate on the design of each module. Due to page limits, we present proofs and theoretical discussion in the Supplementary Material.

3.1 XLEARNER

XLEARNER aims to learn a causal graph \mathcal{G} from multi-dimensional data D in the presence of latent confounders. The primary obstacle is that learning the skeleton of \mathcal{G} requires the faithfulness assumption (see Sec. 2.2), which may be violated by FDs in D . Below, we show how contradictory causal structures can be induced when being agnostic to FDs.

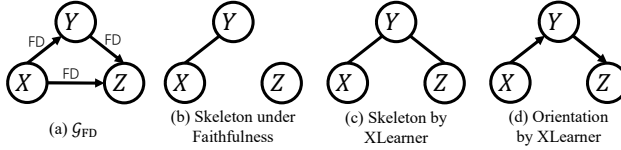


Fig. 4. Illustration of FD-induced faithfulness violation.

Example 3.1: We first consider the **CityInfo** dataset described in Ex. 2.4 and the corresponding FD-induced graph in Fig. 4(a), where X denotes city, Y denotes state, and Z denotes country. By definition of conditional independence (i.e., $X \perp\!\!\!\perp Y \mid Z \iff P(X, Y \mid Z) = P(X \mid Z)P(Y \mid Z)$), we have $Y \perp\!\!\!\perp Z \mid X$ and $X \perp\!\!\!\perp Z \mid Y$. The definition of faithfulness implies $Y \perp\!\!\!\perp_{\mathcal{G}} Z \mid X$ and $X \perp\!\!\!\perp_{\mathcal{G}} Z \mid Y$, given $Y \perp\!\!\!\perp Z \mid X$ and $X \perp\!\!\!\perp Z \mid Y$. Z is non-adjacent to both X and Y according to the m -separation definition (see the induced graph in Fig. 4(b)). Consequently, Z is an isolated node in Fig. 4(b). We have $Y \perp\!\!\!\perp_{\mathcal{G}} Z$ and GMP further implies that $Y \perp\!\!\!\perp Z$, which contradicts $Y \not\perp\!\!\!\perp Z$ entailed by $Y \xrightarrow{FD} Z$. Indeed, the skeleton is not consistent with any MAGs that are on the top of it.

Table 2. Comparing different causal discovery algorithms. \checkmark denotes “support” whereas \times denotes “no support”.

Alg.	Orientation	FD-induced Faithfulness Violation	Causal Insufficiency
PC [Spirtes et al. 2000]	\checkmark	\times	\times
FCI [J. Zhang 2008]	\checkmark	\times	\checkmark
REAL [Ding, Liu, et al. 2020]	\times	\checkmark	\times
XLEARNER	\checkmark	\checkmark	\checkmark

As aforementioned in Sec. 2, the violations of causal sufficiency and faithfulness (induced by FDs) are common in the data analysis scenarios. However, they are addressed separately in the literature, as reviewed in Table 2. XLEARNER focuses on addressing both challenges simultaneously. Fig. 4(c)-(d) show the skeleton and orientation by XLEARNER, which are compliant with intuition.

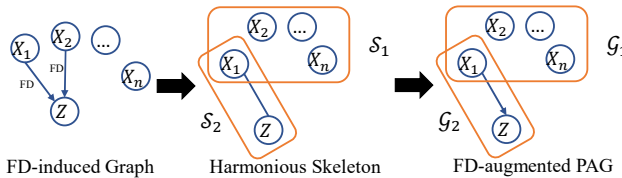


Fig. 5. Running example of XLEARNER.

Overall, XLEARNER tackles the problem in three stages. We outline the workflow of XLEARNER in Alg. 1 and present an example. Then, we elaborate on the design of XLEARNER.

Example 3.2: Consider the FD-induced graph \mathcal{G}_{FD} shown in Fig. 5. In the first stage, XLEARNER uses \mathcal{G}_{FD} to identify variables (e.g., X_1 and Z in Fig. 5) that may trigger faithfulness violations. Then, the skeleton \mathcal{S}_2 is built upon a harmonious assumption instead of faithfulness over X_1 and Z . In the second stage, the FCI algorithm (skeleton learning and orientation) is only conducted over

Algorithm 1: XLEARNER procedure.**Input:** Multi-dimensional Data D , FD-induced graph \mathcal{G}_{FD} **Output:** FD-augmented PAG \mathcal{G}

```

1 // stage 1: detect and preclude  $X_{\text{FD}}$  (Sec. 3.1.1)
2  $\mathcal{S}_2 \leftarrow (V, \emptyset)$ ;
3 Topologically sorting nodes in  $\mathcal{G}_{\text{FD}}$  and record depth as  $d(X_i)$ ;
4 while  $\mathcal{G}_{\text{FD}}$  has non-root nodes do
5    $X \leftarrow \operatorname{argmax}_{X \in \mathcal{G}_{\text{FD}}.V} d(X)$ ;
6    $Y \leftarrow \operatorname{argmin}_{Y \in \text{Pa}(\mathcal{G}_{\text{FD}}, X)} |Y|$ ;
7   add edge  $(X, Y)$  in  $\mathcal{S}_2$ ;
8   remove  $X$  and all connected edges from  $\mathcal{G}_{\text{FD}}$ ;
9 end
10 // stage 2: standard PAG learning
11  $\mathcal{S}_1 \leftarrow \text{FCI-SL}(D, \mathcal{G}_{\text{FD}}.V)$ ;
12  $\mathcal{G}_1 \leftarrow \text{FCI-Orient}(\mathcal{S}_1)$ ;
13 // stage 3: orient  $\mathcal{S}_1$  and generate  $\mathcal{G}$  (Sec. 3.1.2)
14 foreach  $(X \xrightarrow{\text{FD}} Y) \in \mathcal{G}_{\text{FD}}.E$  do
15   if  $X, Y$  is adjacent in  $\mathcal{S}$  then orient  $X \rightarrow Y$  on  $\mathcal{G}^2$ ;
16 end
17 generate  $\mathcal{G}$  concatenating  $\mathcal{G}^1$  and  $\mathcal{G}^2$ ;
18 return  $\mathcal{G}$ ;

```

variables that comply with the faithfulness assumption. Hence, the skeleton \mathcal{S}_1 and the PAG \mathcal{G}_1 are identified accordingly. In the third stage, we orient \mathcal{S}_2 to generate an FD-augmented PAG \mathcal{G}_2 . By concatenating \mathcal{G}_1 and \mathcal{G}_2 , the resultant (FD-augmented) PAG \mathcal{G} is obtained.

Comparison with FCI. Comparing with the FCI algorithm [Spirites et al. 2000; J. Zhang 2008], XLEARNER for the first time reconciles functional dependency (FD) and the faithfulness assumption for causally insufficient data within the harmonious skeleton framework. In that sense, it can learn causal graphs from real-world data adequately. As validated in Sec. 4.3, XLEARNER learns more accurate causal graphs than the FCI algorithm. Second, it uses FDs to provide a more complete orientation to the underlying causal graph. Hence, compared to the FCI algorithm, it leverages the knowledge from FDs to enforce a more precise causal graph with less undetermined edges. In sum, we deem that XLEARNER enhances the FCI algorithm from the theoretical perspective, and also addresses obstacles in the real-life adoption of the FCI algorithm.

3.1.1 Skeleton Learning with FD (lines 1–9 of Alg. 1). Ding et al. point out that in the presence of FD relations, faithfulness assumption can be violated thus we can at most obtain a *harmonious skeleton* [Ding, Liu, et al. 2020]. However, the original theory of *harmonious skeletons* is established under causal sufficiency. Here, we further generalize the *harmonious skeleton* for causally insufficient systems:

DEFINITION 3.1 (HARMONIOUS SKELETON). A skeleton \mathcal{S} is said to be *harmonious* w.r.t. a joint probability distribution P if 1) there exists a MAG \mathcal{G} sharing the same adjacencies of \mathcal{S} , 2) P satisfies GMP to \mathcal{G} , and 3) any subgraph of \mathcal{S} does not satisfy the previous two conditions.

Def. 3.1 entails three properties of \mathcal{S} . First, since there exists a MAG \mathcal{G} on top of the skeleton \mathcal{S} , there exists a set of nodes that m-separates any non-adjacent nodes. Second, if two nodes (e.g., X, Y) are m-separated by Z , then $X \perp\!\!\!\perp Y \mid Z$. These two conditions imply that X and Y are non-adjacent

in \mathcal{S} , if and only if there exists a set of nodes Z such that $X \perp\!\!\!\perp Y \mid Z$. The last condition implies the minimality of \mathcal{S} , which is commonly assumed [Peters, Janzing, and Schölkopf 2017]. When two graphs $\mathcal{G}, \mathcal{G}'$ are equally compatible with the data, we would prefer the simpler one. We now show the construction of \mathcal{S} , which begins with a basic case and generalizes to arbitrary structures.

THEOREM 3.1. *Let Z be a sink node (i.e., all edges of Z are oriented to Z) in \mathcal{G}_{FD} . $\mathcal{S} = \mathcal{S}_1 \cup \mathcal{S}_2$ is a harmonious skeleton if 1) \mathcal{S}_1 is a harmonious skeleton over $X \setminus Z$ and \mathcal{S}_2 contains only one edge $X_i - Z$ where X_i can be any node connected to Z in \mathcal{G}_{FD} .*

Example 3.3: Fig. 5 presents an example of Thm. 3.1. Connecting the sink node Z with one of its parents X_1 in the \mathcal{G}_{FD} yields a skeleton \mathcal{S}_2 . If we can learn a harmonious skeleton \mathcal{S}_1 on the remaining nodes (X_1, \dots, X_n) , Thm. 3.1 ensures that concatenating $\mathcal{S}_1, \mathcal{S}_2$ produces a harmonious skeleton over all variables.

According to Thm. 3.1, if Z has more than one parent, multiple harmonious skeletons exist (note that Z can connect to any one of its parents). In practice, we connect Z to the parent node with the lowest cardinality. Given a FD-induced graph, we recursively apply Thm. 3.1 to identify sink nodes and derive the corresponding harmonious skeleton \mathcal{S}_2 until all FDs are properly resolved. Then, we can apply the standard skeleton learning algorithm over the remaining nodes. The procedure is shown in lines 1–9 in Alg. 1.

THEOREM 3.2. *The skeleton of Alg. 1 is harmonious.*

Alg. 1 first constructs an empty skeleton \mathcal{S} that shares the same nodes as \mathcal{G}_{FD} (line 2). At line 3, we topologically sort the \mathcal{G}_{FD} nodes (note that \mathcal{G}_{FD} is a DAG). In each iteration (lines 5–8), we pick the deepest node and apply Thm. 3.1 to connect X to one of its parents (in \mathcal{G}_{FD}) Y in the skeleton. We use the parent node with the lowest cardinality as Y (line 6), as it usually aligns with human intuition.

Example 3.4: Consider the **CityInfo** dataset in Ex. 2.4. Alg. 1 identifies the correct skeleton as City – State – Country in Fig. 4(c).

For root nodes, since there are no FDs and thus the faithfulness assumption holds, we employ the standard FCI algorithm (lines 10–12) to infer the PAG \mathcal{G}_1 . After \mathcal{S}_2 being oriented to \mathcal{G}_2 (see Sec. 3.1.2), we concatenate them to form \mathcal{G} (line 17). Thm. 3.2 proves that the skeleton of \mathcal{G} is also harmonious after the concatenation.

3.1.2 Orientation (lines 13–16 of Alg. 1). Classical constraint-based causal discovery algorithms decide the direction of edges based on a set of orientation rules. These rules orient undirected edges on skeletons (i.e., $\circ-\circ$) based on a set of criteria, including conditional independence and some graphical structural relations (e.g., discriminating path) [J. Zhang 2008]. These rules are applied iteratively until no more orientations can be made. However, we argue that an FD itself reflects a causal relation to a good extent, of which the reason is twofold.

ANM Perspective on FD-related Edges. We anticipate incorporating the discrete additive noise model (ANM) [Peters, Janzing, and Schölkopf 2011] for orienting FD-related edges. The main theory of ANM implies that if an asymmetric ANM $Y = f(X) + N_Y$ exists from X to Y and N_Y is independent of X , then X causes Y . By FD, we note that, if $X \xrightarrow{\text{FD}} Y$ in \mathcal{G}_{FD} , an ANM construction from X to Y naturally exist with noise term $N_Y = 0$. On the other hand, an ANM construction from Y to X exists only in very rare cases, as determined by the identifiability of the discrete ANM (see Thm. 4.6 in [Peters, Janzing, and Schölkopf 2017]). In light of this, we hypothesize that $X \xrightarrow{\text{FD}} Y$ in \mathcal{G}_{FD} implies causation of $X \rightarrow Y$.

FCI Perspective on FD-related Edges. The rules in FCI may be unreliable due to the faithfulness violations by FDs. However, an FD itself is generally more reliable, which describes deterministic

relations. More importantly, the directions from the FDs are compatible with the result of the FCI on the variables excluding FD-related variables. That is, incorporating ANM would not violate GMP.

We implement the above hypothesis in our orientation algorithm (lines 13–16 in Alg. 1). We examine, for each FD relation that is also adjacent in \mathcal{S}_2 , whether the edge is oriented as \rightarrow (lines 13–15). We note that, by incorporating ANM, the augmented graph is more informative and represents an overcomplete graph w.r.t. the ground-truth MAG’s Markov equivalence class, exhibiting greater precision than causal graphs learned only by rules.

3.2 XTRANSLATOR

A causal graph does not directly reveal if a variable adequately explains a WHY QUERY, nor does it directly reflect if the variable features a causal or non-causal explanation. Bridging this gap requires a translation from causal primitives to XDA semantics. To illustrate, we start with a WHY QUERY under AVG. We then show how to generalize the main result into SUM and other aggregates.

Principle of Explainability. Given a WHY QUERY Δ where $agg = \text{AVG}$, a variable X is said to have *No Explainability* if $X \perp\!\!\!\perp M \mid F \cup B$, where M is the target measure, F is the foreground variable, and B are background variable(s). In the subsequent discussion, we omit B for the ease of presentation without loss of generality.

A WHY QUERY in XDA requires us to observe the difference between aggregates on M within two subspaces. The conditional independence of $X \perp\!\!\!\perp M \mid F$ implies that $\mathbb{E}(M \mid F, X) = \mathbb{E}(M \mid F)$. Hence, $\Delta(D) = \Delta(D_{X=x})$ in the large sample limit for all feasible filters in X . If X is *conditionally independent* of M given F , X is simply impossible to offer explanations to the WHY QUERY. Thus, this principle imposes a restriction on possible variables that have the potential to provide explanations. In particular, we derive the following restriction.

PROPOSITION 3.1. *If X has explainability, M, X are not m -separated by F in the causal graph G .*

Proposition 3.1 illustrates the chance of pruning variables for which it is impossible to provide explanations. Table 3 further depicts the translation from causal primitives to XDA semantics. In XTRANSLATOR, a variable X is first confirmed to have explainability if X, M are not m -separated by F in G (1st row in Table 3). In addition, XTRANSLATOR also categorizes whether X is causal or non-causal according to Table 3. Overall, X provides a causal explanation if it is explainable and a cause (① and ② in Table 3) or a possible cause (③ and ④ rows in Table 3) of M . We show how the causal graph identified by XLEARNER is translated.

Example 3.5: Given the dataset in Fig. 1(a), XLEARNER identifies the corresponding causal graph in Fig. 1(c). With the WHY QUERY in Fig. 1(b), XTRANSLATOR translates the causal graph into the XDA semantics in Fig. 1(d). “Smoking” and “Stress Level” can be used to *causally* explain “Lung Cancer”. And, other variables (e.g., “Surgery”) are deemed non-causal explanations (last row in Table 3).

Table 3. Translating causal primitives to XDA semantics.

Rule	Path	Causal Primitive	XDA Semantics
①	$X \rightarrow F \rightarrow M, \dots$	m -separated	no explainability
②	$X \rightarrow M$	parent	causal explanation
③	$X \rightarrow \dots \rightarrow M$	ancestor	causal explanation
④	$X \circ \rightarrow M$	almost parent	causal explanation
⑤	$X \circ \rightarrow \dots \circ \rightarrow M$	almost ancestor	causal explanation
⑥	others	N/A	non-causal explanation

Extension to SUM. The above formulation over *explainability* is established on AVG aggregates. In the following, we discuss the implications of our formulation on SUM aggregates. If X has no explainability, $X \perp\!\!\!\perp M \mid F$. When we enforce $X = x$, $\Delta(D_{X=x})$ can merely be affected by the number

of rows where $X = x$ in two sibling subspaces (namely a COUNT-based explanation) instead of a causal relation between X and M (see detailed formulation in Supplementary Material). This may be valid for explanations; nevertheless, it is typically inconsistent with the common intuition of data analysis and may not align user expectations regarding explanations (i.e., a variable explains the target). Such COUNT-based explanation is unconventional and is thus less of a concern.

Semantics Consistency. Following the above discussion, we clarify that a variable may play different roles in various aggregates. However, in our current design, XTRANSLATOR focuses primarily on variables with strong connections to M , which are more likely to provide desirable explanations. Therefore, the semantics of a variable are consistent across different aggregates. As clarified in **Principle of Explainability** above, it is appropriate for pruning uninformative variables from general aggregates, and we do not observe notable issues in practice. We leave designing more comprehensive translation rules for future research.

3.3 XPLAINER

XLEARNER and XTRANSLATOR together provide a coarse-grained, variable-level qualitative explanation to a WHY QUERY. For instance, “Smoking” is a causal explanation for the differences in severity of “Lung Cancer” in Locations A and B. To go one step further, XPLAINER provides predicate-level quantitative explanations to answer WHY QUERY (e.g., “Smoking=Yes” explains the difference with the responsibility of 0.77 in Fig. 1(f)). XPLAINER is on the basis of a well-establish framework, DB causality [Meliou, Gatterbauer, Halpern, et al. 2010] (an extension of actual causality). To ease reading, below we first provide a recap of the notations defined in Sec. 2.1. We then rewrite the formulation of DB causality in the context of XINSIGHT in Sec. 3.3.1.

Recap of Notations. We refer to a dataset as D , a filter as a lowercase p , and a set of filters as an uppercase P . The subset of D satisfying p (or P) is represented by D_p (or D_P , respectively). We use $D - D_p$ as the complement of D_p in D . By default, $\Delta(D)$ represents the WHY QUERY over the dataset D . Likewise, for arbitrary $D' \subseteq D$, $\Delta(D')$ represents the difference between the aggregated values of two sibling subspaces inside D' .

DEFINITION 3.2 (DB CAUSALITY [MELIOU, GATTERBAUER, HALPERN, ET AL. 2010]). *Given a multi-dimensional data D and WHY QUERY Δ , let t be a tuple in D . t is called a counterfactual cause to Δ , if $\Delta(D) > \epsilon$ and $\Delta(D - \{t\}) \leq \epsilon$, where ϵ is a user-defined threshold. t is called an actual cause to Δ , if there exists a contingency $\Gamma \subseteq D$ such that t is a counterfactual cause for $D - \Gamma$ (i.e., $\Delta(D - \Gamma - \{t\}) \leq \epsilon < \Delta(D - \Gamma)$).*

DEFINITION 3.3 (DB RESPONSIBILITY [MELIOU, GATTERBAUER, HALPERN, ET AL. 2010]). *Suppose P is an actual cause to WHY QUERY Δ and Γ ranges over all valid contingencies for P . The responsibility of P is defined as $\rho_P = \frac{1}{1 + \min_{\Gamma} |\Gamma|}$, where $|\Gamma|$ denotes the number of tuples in the contingency.*

DB causality is appealing as it offers both a normalized measure (responsibility $\in (0, 1]$) and a contingency. First, when the responsibility is close to 1, it implies that the tuple is more accountable for the outcome, and when it hits 1, it is totally responsible. Second, the minimal contingency reflects the additional influential factors that, together with the tuple, are fully responsible for the outcome. The two elements form a quantitative explanation and it is useful for users to understand why the difference exists.

3.3.1 Adaption. DB causality was originally designed for data provenance. As pointed out in [Meliou, Roy, et al. 2014], tuple-level explanations are usually too fine-grained for data analysis scenarios. An individual tuple usually has too little effect on the highly aggregated outcome of a large dataset. Recalling the example in Fig. 1, users would expect to know that “Smoking=Yes” causes high

“Lung Cancer” severity rather than an individual patient being the cause of the high “Lung Cancer” severity. This necessitates predicate-level explanations which are easier to understand and frequently used in data analysis scenarios, and by many data explanation tools [Abuzaid et al. 2021; E. Wu and Madden 2013]. Motivated by this, we make three adaptations over DB causality (namely, W-CAUSALITY, W-RESPONSIBILITY and conciseness) to support XDA. We formulate W-CAUSALITY as follows.

DEFINITION 3.4 (W-CAUSALITY). *Given a multi-dimensional data D , an attribute of interest X and WHY QUERY Δ , let $P \subseteq \bigcup p_i$ be a predicate in D , where $\bigcup p_i$ denotes the set of all possible filters on X . P is called a counterfactual cause of Δ , if $\Delta(D) > \epsilon$ and $\Delta(D - D_P) \leq \epsilon$, where ϵ is a user-defined threshold. P is deemed to be an actual cause of Δ , if there is a contingency $\Gamma \subseteq \bigcup p_i$ such that P is a counterfactual cause for $D - D_\Gamma$ (i.e., $\Delta(D - D_\Gamma - D_P) \leq \epsilon < \Delta(D - D_\Gamma)$), where $P \cap \Gamma = \emptyset$.*

From Tuples to Predicates. Def. 3.4 transforms the tuple and contingency into two predicates. This way, explanations as well as contingencies constitute a form of intervention over the multi-dimensional data. When a contingency Γ is applied, it indicates that, if the events related to Γ do not happen, then the events related to P are fully responsible for Δ . This adaptation in turn entails another adaptation to the responsibility for predicate-level explanations.

DEFINITION 3.5 (W-RESPONSIBILITY). *Suppose P is an actual cause to WHY QUERY Δ and Γ range over all valid contingencies for P . The responsibility of P is defined as $\rho_P = \frac{1}{1 + \min_\Gamma |\Gamma|_W}$, where $|\Gamma|_W$ is defined as $\max(\frac{\Delta(D - D_P) - \Delta(D - D_P - D_\Gamma)}{\Delta(D)}, 0)$. We let $\rho_P = 0$ if P is not an actual cause.*

W-RESPONSIBILITY. Instead of using the number of rows in D_Γ as $|\Gamma|_W$, Def. 3.5 employs the truncated difference in Γ over Δ to measure the importance of P . In particular, $\Delta(D - D_P) - \Delta(D - D_P - D_\Gamma)$ can be deemed as *first-order finite backward difference* to the function $\Delta(\cdot)$ at the point of $D - D_P$ and Γ is the step size. This supplies a simple and intuitive way to understand to what extent Γ plays an important role in reducing the difference. The large difference imposed by Γ implies a low importance of explanation P to Δ ; because the reduction in Δ is primarily caused by Γ instead of P itself. Therefore, the responsibility of P is measured by a valid contingency Γ^* (such that $\Delta(D - D_P - D_\Gamma) \leq \epsilon$ and $\Delta(D - D_\Gamma) > \epsilon$) with minimal difference on Δ .

Conciseness. Using responsibility as the sole criterion is not sufficient in practical data analysis scenarios [Glavic et al. 2021]. Typically, a *concise* explanation is preferable. Therefore, given an attribute of interest X , we formulate the optimal explanation of X as follows.

$$\operatorname{argmax}_{P \subseteq \bigcup p_i} \rho_P - \sigma |P| \quad (4)$$

where $\bigcup p_i$ is the set of all possible filters in X , $|P|$ is the number of filters in P , and $\sigma |P|$ ($\sigma > 0$) forms a conciseness regularization. In practice, we would prefer $\sigma = 1/m$ such that when all filters are picked, the score is zero.

Table 4. Different search solutions in XPLAINER. FP is false positive and FN is false negative.

Solution	Complexity	Optimality
Brute-force Search	$O(2^m)$	Optimal
Approx. Search (SUM)	$O(m \log m)$	Moderated FP; Negligible FN
Approx. Search (AVG)	$O(m^2)$	Moderated FP&FN

3.3.2 Optimization. As pointed out in [Bertossi 2020; Bertossi et al. 2020], computing responsibility (i.e., ρ_P) is intractable. Furthermore, solving the optimization problem in Eqn. 4 is itself difficult given 2^m search space (m is the number of filters in X). We characterize the performance of different solutions in Table 4. First, the brute-force search is the most accurate and general method for arbitrary aggregates, despite being very slow. The explanation discovered by brute-force search is

exactly the optimal explanation. In this paper, we design two approximate solutions for SUM and AVG, respectively. In particular, we first show the existence of a linearithmic approximated solution when the aggregation is SUM. This solution also has negligible false negatives, as theoretically guaranteed by a lemma on the completeness. Furthermore, we present a heuristics-based solution for AVG with quadratic complexity. Moreover, this solution should be applicable for other aggregate functions with a mild downgrade in optimality. Our evaluation (Sec. 4.4) shows that both approximations are tight and efficient in comparison to brute-force search.

Optimization for SUM. Given the additive property of SUM (i.e., $\Delta(D_{P_1} + D_{P_2}) = \Delta(D_{P_1}) + \Delta(D_{P_2})$), we obtain the following proposition to prune the search space.

PROPOSITION 3.2. *If P^* is the optimal explanation of Eqn. 4, $\forall p \in P^*, \Delta(D_p) > 0$.*

According to Proposition 3.2, the search algorithm can omit filters with a non-positive Δ_i (i.e., $\Delta(D_{p_i})$). Recall that Eqn. 4 seeks the optimal explanation. When the aggregate function is SUM, we only need to focus on filters with a reasonably high Δ_i without losing optimality and we define such filters as *canonical filters*.

DEFINITION 3.6 (CANONICAL FILTER AND PREDICATE). *Without loss of generality (w.l.o.g.), given a WHY QUERY Δ and an attribute of interest X , let filters $\{p_1, \dots, p_m\}$ of X be ordered by Δ_i (i.e., $\Delta(D_{p_i})$) such that $\Delta_1 \geq \dots \geq \Delta_m$. We let p_1, \dots, p_j be canonical filters if*

$$\Delta(D) - \sum_{i=1}^j \Delta_i \leq \epsilon < \Delta(D) - \sum_{i=1}^{j-1} \Delta_i \quad (5)$$

$P^C = \{p_1, \dots, p_j\}$ is called a *canonical predicate* and $\tau = \sum_{i=1}^j \Delta_i$.

With canonical filters and a corresponding canonical predicate P^C , we observe that P^C manifests good properties. First, P^C is the minimal counterfactual cause entailed by Eqn. 5. Our construction of canonical predicates guarantees completeness.

PROPOSITION 3.3 (COMPLETENESS). *For SUM, given a WHY QUERY Δ , an attribute of interest X and corresponding canonical predicate P^C , there exists an optimal explanation $P^* \subseteq P^C$.*

The completeness proposition (Proposition 3.3) allows us to only focus on canonical filters when searching for the optimal explanation without loss of optimality. More importantly, the canonical predicate also allows us to efficiently identify actual causes and the corresponding valid contingencies.

THEOREM 3.3. *For SUM, given a WHY QUERY Δ , an attribute of interest X and corresponding canonical predicate P^C , $\forall P \subset P^C$, P is an actual cause and $\bar{P} = P^C - P$ is a valid contingency.*

The advantages of Thm. 3.3 are twofold. First, we can directly confirm valid explanations without exhaustive enumerations. Second, by the property of \bar{P} , we bound P 's responsibility (ρ_P).

THEOREM 3.4. *For SUM, given a WHY QUERY Δ , an attribute of interest X and corresponding canonical predicate P^C , the W-RESPONSIBILITY ρ_P of $P \subset P^C$ satisfies*

$$\frac{1}{1 + \frac{\tau - \Delta(D_P)}{\Delta(D)}} \leq \rho_P \leq \frac{1}{2 - \frac{\Delta(D_P) + \epsilon}{\Delta(D)}} \quad (6)$$

When $\Delta(D_P) \ll \tau$ and $0 < \tau \leq \Delta(D)$, $\frac{1}{1 + \tau - \Delta(D_P)} \approx \frac{1 + \tau + \Delta(D_P)}{(1 + \tau)^2}$ and the corresponding approximation error rate $E < 0.25$.

Thm. 3.4 provides a way to efficiently approximate responsibility with theoretical guarantees. In that sense, we can compute responsibility immediately and alleviate searching the minimal contingency. Let $\hat{\rho}_P = \frac{1+\tau+\Delta(D_P)}{(1+\tau)^2}$, we can rewrite the objective function in the following form.

$$\hat{\rho}_P - \sigma|P| = C_1 + C_2 \times \sum_{p_i \in P} (\Delta_i - C_3) \quad (7)$$

Here, C_1, C_2, C_3 are constants. Then, the optimal explanation to Eqn. 7 is straightforward:

$$P^* = \{p_i \mid \Delta_i > C_3\} \quad (8)$$

where $C_3 = \frac{\sigma\Delta(D)}{(1+\frac{\tau}{\Delta(D)})^2}$. The complexity is $\mathcal{O}(m \log(m))$ (primarily in sorting filters for generating canonical predicates).

Optimization for AVG. In terms of AVG, it is generally much more challenging due to the absence of the additive characteristics on $\Delta(D_P)$. Therefore, the majority of the preceding propositions are not applicable. Having said that, we find the causal graph gives considerable opportunities to prune unnecessary computations.

DEFINITION 3.7 (HOMOGENEOUS SIBLING SUBSPACE). *Given sibling subspaces s_1, s_2 (with foreground variable F and background variables \mathbf{B}), an attribute X and the causal graph G , s_1, s_2 are homogeneous on X if X, F are m -separated given \mathbf{B} on G .*

PROPOSITION 3.4. *For a homogeneous AVG, given a WHY QUERY Δ , an attribute of interest X , a predicate $P \subseteq \bigcup p_i$ and a filter $p \in P$, if $\Delta(D_p) > \Delta(D_P)$, then $\Delta(D_P - D_p) < \Delta(D_P)$.*

To practically address the search problem of AVG (Eqn. 4), we rely on greedy-based heuristics with a pruning strategy enabled by Proposition 3.4. The algorithm is outlined in Alg. 2.

The high-level idea behind Alg. 2 is similar to the one for SUM, which attempts to construct a canonical predicate P^C such that P^C forms a counterfactual cause, each subset $P \subset P^C$ of the canonical predicate constitutes an actual cause, and the complement set $P^C - P$ is a valid contingency. Unlike SUM, however, Alg. 2 does not ensure the optimality of the resulting explanation, primarily due to the incompleteness of the canonical predicate (Proposition 3.3) under AVG. Recall that ρ_P ranges from 0 to 1 in Eqn. 4. The optimal explanation contains at most $1/\sigma$ filters (otherwise, $\rho_P - \sigma|P| < 0$). Hence, the canonical predicate P^C shall contain at most $1/\sigma$ or m (i.e., the number of filters in the attribute).

Alg. 2 employs a greedy strategy to construct P^C progressively. It starts with an empty canonical predicate (line 1) and inserts one filter in each iteration (lines 2–13). Before insertion, it checks whether P^C is a canonical predicate (line 3) and terminates the loop if P^C is already valid. Otherwise, it picks the remaining filters that were not chosen in earlier iterations as candidates (line 5) and inserts the filter that minimizes the difference $\Delta(D - D_{P^C} - D_{p_i})$ at the highest magnitude into P^C (lines 6–12). When homogeneity holds and $\Delta_i \leq \Delta(D - D_{P^C})$, Alg. 2 prunes p_i according to Proposition 3.4 (lines 7–8). Note that Δ_i is invariant throughout the loop; thus it only needs to be queried once. In general cases where homogeneity does not hold, Alg. 2 has to enumerate all possible filters in \bar{P} (line 10). If we cannot obtain a valid canonical predicate (i.e., a counterfactual cause to Δ) after the loop, Alg. 2 terminates and outputs \perp , indicating that it fails to find the optimal explanation within the attribute (line 15). Empirically, we do not observe such rare cases. When the canonical predicate P^C is obtained, $\forall k = 1, \dots, |P^C| - 1$, the top- k filters of $P_k \subseteq P^C$ is a valid actual cause and the complement set $\Gamma_k = P^C - P_k$ is a valid contingency. According to the termination condition in the above loop (line 3), $\Delta(D - D_{P_k}) > \epsilon$. In addition, according to the definition of canonical predicate $\Delta(D - D_{P^C}) = \Delta(D - D_{P_k} - D_{\Gamma_k}) \leq \epsilon$, Γ_k is a valid contingency to P_k . Therefore, we compute the approximated responsibility $\hat{\rho}_{P_k}^*$ by using its lower bound deduced by Γ_k (line

Algorithm 2: XPLAINER FOR AVG**Input:** WHY QUERY Δ , threshold ϵ , consisteness parameter σ **Output:** (near) optimal explanation P^*

```

1  $P^C \leftarrow \emptyset$ ;
2 foreach  $r = 1, \dots, \min(m, \frac{1}{\sigma})$  do
3   if  $\Delta(D - D_{PC}) \leq \epsilon$  then break;
4   else
5      $\bar{P} \leftarrow \{p_1, \dots, p_m\} - P^C$ ;
6     if homogeneous then
7        $S \leftarrow \{p_i \mid p_i \in \bar{P}, \Delta_i > \Delta(D - D_{PC})\}$ ;
8        $p^* \leftarrow \operatorname{argmin}_{p \in S} \Delta(D - D_{PC} - D_p)$ ;
9     else
10       $p^* \leftarrow \operatorname{argmin}_{p \in \bar{P}} \Delta(D - D_{PC} - D_p)$ ;
11     end
12      $P^C \leftarrow P^C \cup \{p^*\}$ ;
13   end
14 end
15 if  $\Delta(D - D_{PC}) > \epsilon$  then return  $\perp$ ;
16 foreach  $k \in 1, \dots, |P^C|$  do
17    $P_k \leftarrow$  top-k filters of  $P^C$ ;
18    $\Gamma_k \leftarrow P^C - P_k$ ;
19   compute  $\rho_{\hat{P}_k}$  with  $\Gamma_k$ .
20 end
21 return  $\operatorname{argmax}_k \rho_{\hat{P}_k} - \sigma|P_k|$ ;

```

19). After enumerating each k , Alg. 2 returns P_k such that $\rho_{\hat{P}_k} - \sigma|P_k|$ is maximized (line 21). In summary, the first loop (lines 2–14) is of quadratic complexity regardless of homogeneity and the second loop is linear (lines 16–20). The total complexity is $O(m^2)$.

4 EVALUATION

In this section, we evaluate XINSIGHT to answer the following three research questions (RQs):

- (1) **RQ1: End-To-End Performance.** How can XINSIGHT facilitate end users in explainable data analysis?
- (2) **RQ2: XLEARNER Evaluation.** Does XLEARNER effectively recover causal relations from observational data?
- (3) **RQ3: XPLAINER Evaluation.** Does XPLAINER accurately and efficiently yield explanations?¹

4.1 Datasets & User Study Setup

To the best of our knowledge, there is no real-world benchmark with manually labeled query/explanation pairs. To deliver a scientific evaluation, we conduct experiments on ① public datasets collected from previous works, ② real-world data collected from a production environment for user study and human evaluation, and ③ synthetic data with ground-truth explanations. The detailed steps for generating synthetic datasets are given in the Supplementary Material. We make necessary preprocessing before feeding to XINSIGHT (e.g., remove missing values).

① **Flight Delay (FLIGHT).** We use the flight delay dataset from [Salimi, Cole, et al. 2017] to explore the causes of flight delays in US airports. After preprocessing, the resulting dataset contains 17

¹The correctness of XTRANSLATOR has been rigorously discussed in Sec. 3.2.

variables, including the weather conditions of departure airports (temperature, humidity, visibility, rain, etc.), flight carrier, flight time (month, quarter, year, day of week and hour) and two variables indicating flight delays, DelayMinute (continuous) and Delay>15min (binary).

① **Hotel Booking (HOTEL).** The hotel booking dataset [Antonio et al. 2019] is frequently used for demonstrating data analysis methods. It contains 119,390 observations from two hotels. Each observation depicts the booking status (e.g., “room type”, “reservation status”, and “is canceled”) of a guest.

② **Web Service Behavior Dataset (WEB).** The dataset is collected from a web service’s production environment. It contains 29 columns and 764 rows, where each row is a list of binary values. The first 28 columns describe user behaviors on the web service (e.g., whether he clicks a specific button), which are collected by a logging module. The last column indicates whether the user was blocked for publishing malicious content (i.e., “IsBlocked”), which is annotated by cybersecurity experts. These behaviors are known to exhibit strong and clear causal relations, making it appropriate for testing XINSIGHT in real-world scenarios.

③ **Synthetic Data A (SYN-A).** Ground-truth causal graphs are unattainable in practice and it is common to generate random graphs and then sample observational data from this graphical model. We generate MAGs with 10 to 150 variables (141 distinct scales in total). For each scale, we synthesize five random graphs and the associated datasets, resulting in 705 (141 × 5) datasets. Each dataset is injected with different amount of FDs.

③ **Synthetic Data B (SYN-B).** We follow the approach in Scorpion [E. Wu and Madden 2013] to synthesize datasets for assessing XPLAINER. Each dataset includes a valid WHY QUERY and a ground-truth explanation to this difference. We generate 18 datasets with different difficulties.

User Study Setup. In addition to the experiments that will be launched shortly, we intend to determine the extent to which the results on WEB is correct and reasonable. Nonetheless, rendering professional judgments on explanations and causal claims require sufficient expertise in this domain, which makes gathering a large number of participants difficult. In this study, we recruit six domain experts for the WEB dataset; we confirm each expert can evaluate the explanations and causal claims with professionalism and high confidence. We organize the user study as follows:

- (1) **Participant Education.** We organize an education session for participants and demonstrate how to discern between causation and correlation. Then, a pilot study is conducted to confirm that participants have an adequate sense of causality.
- (2) **Explanation Assessment.** We raise four WHY QUERY and ask XINSIGHT to generate two explanations for each WHY QUERY. We then ask participants to give each explanation a score (between 0 to 5) based on their domain knowledge.
- (3) **Causal Claim Assessment.** Following [Law et al. 2021], we collect eight edges connected to “IsBlocked”, transform these causal relations into human-comprehensible causal claims and ask participants to independently evaluate them (by labeling them as “reasonable”, “not reasonable”, or “unsure”).
- (4) **Follow-up Discussion.** Participants explain their decision and discuss the aggregated results.

4.2 RQ1: End-To-End Performance

We show that XINSIGHT generates plausible and intuitive explanations for diverse datasets (FLIGHT, HOTEL and WEB) and invite experts to assess the quality of explanations generated for WEB. In this experiment, we manually discover noticeable differences to form WHY QUERY, and ask XINSIGHT to supply the explanations. We also compare XINSIGHT’s outputs with naive correlation-based explanations. To ease presentation, we describe a WHY QUERY in human-readable natural language in the following paragraphs.

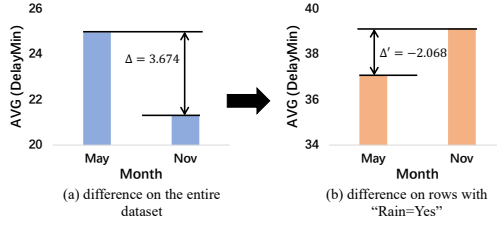


Fig. 6. Explanation of WHY QUERY on the FLIGHT dataset.

FLIGHT & HOTEL. We ask the following WHY QUERY on ①:

- (1) FLIGHT: *why* AVG(DelayMinute) in May (24.95 min) is notably higher than the one in November (21.28 min)?
- (2) HOTEL: *why* AVG(IsCanceled) (cancellation rate) in July (0.37) is notably higher than the one in January (0.30)?

For the first WHY QUERY, we observe that the duration of flight delay differs by month, particularly for May and November, which motivates us to ask XINSIGHT for explanations – what the cause of the flight delay difference is. XINSIGHT first learns a causal graph from data and identifies “rain” as a direct cause of DelayMinute. Then, XPLAINER finds that the difference is reversed ($\Delta = 3.674$ vs. $\Delta' = -2.068$) when the condition “rain=Yes” is enforced (Fig. 6). Thus, it returns “rain=Yes” as an explanation. We interpret the explanation as correct because 1) rain is a typical reason for flight delay, and 2) for most states, monthly precipitation in May is usually higher than in November. Hence, when only counting the rainy cases (by enforcing “rain=Yes”), the difference is eliminated.

For the second WHY QUERY, we observe that the cancellation rate varies by month of arrival. For instance, the cancellation rate in July is 0.37, which is higher than in January. Thus, we ask XINSIGHT for explanations. XINSIGHT identifies “LeadTime” (number of days between booking data and arrival date) as an indirect cause of “IsCanceled”. It also discovers that when enforcing “LeadTime \leq 133”, the difference is reduced. This is intuitive. A longer “LeadTime” results in greater uncertainty about guests’ future schedules, leading to a higher cancellation rate. In January, LeadTime of most reservations is less than 133 days (91%). In contrast, there are far more early reservations (> 133 days) in July (48%), resulting in a higher cancellation rate. When these early reservations are excluded, the difference becomes much smaller.

Table 5. Results of explanation assessment. E_i and P_i stand for the i th explanation and the i th participant, respectively.

	E1	E2	E3	E4	E5	E6	E7	E8
P1	4	4	5	4	4	4	5	3
P2	4	4	4	4	3	4	3	4
P3	5	3	4	5	3	5	5	5
P4	3	4	5	4	4	3	3	4
P5	4	2	5	3	5	4	3	3
P6	5	4	5	5	5	4	5	5
mean	4.16	3.50	4.67	4.17	4.00	4.00	4.00	4.00
std	0.69	0.76	0.47	0.69	0.82	0.58	1.00	0.82

WEB. We report the results of the second phase in the user study (i.e., Explanation Assessment in Sec. 4.1) in Table 5. We view the results as encouraging, since nearly all responses are positive (≥ 3). Moreover, the average scores for seven out of eight explanations are ≥ 4 . We also investigated the explanation with the lowest score (E2 in Table 5). We find this is counter-intuitive but reasonable in retrospect. In the follow-up session, the discussion among participants also confirmed our finding. During the assessments, experts find many explanations inspiring and insightful, despite their familiarity with the dataset. It continuously increases their knowledge and help them design a better criteria for detecting malicious behavior.

Answer to RQ1: *XINSIGHT shows a promising end-to-end performance in explaining data differences. The user study validates that XINSIGHT achieves a respectable level of agreement with experts.*

4.3 RQ2: XLEARNER Evaluation

As a cornerstone of XINSIGHT, XLEARNER is crucial to the effectiveness of the entire pipeline. To answer **RQ2**, we run XLEARNER on **SYN-A** which has ground truth causal graphs and on a real-world dataset **WEB**. Since **WEB** does not associate a ground-truth causal graph, we assess the quality of causal relations by the user study.

Table 6. Overall comparison between XLEARNER and FCI.

Algo.	F1-Score	Precision	Recall
XLEARNER	0.88 ± 0.04	0.95 ± 0.03	0.82 ± 0.06
FCI	0.72 ± 0.05	0.92 ± 0.04	0.59 ± 0.06

Table 6 provides an overall comparison between XLEARNER and FCI on **SYN-A** datasets. We find that XLEARNER is more accurate than FCI in the presence of FDs. In particular, while FCI has comparable precision, XLEARNER has a much higher recall. This confirms our discussion on the implications of FDs in Sec. 3.1. The faithfulness violations mislead FCI to incorrectly refute true edges (thus yield a lower recall) while XLEARNER is aware of such faithfulness violations and handles them with the procedure in Sec. 3.1.

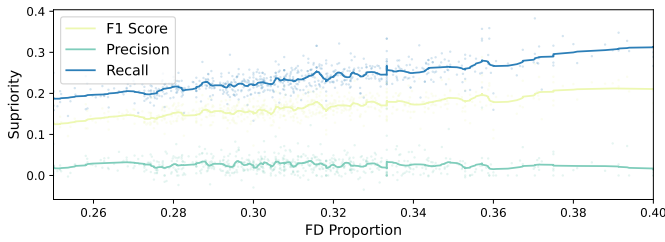


Fig. 7. Comparison by FD Proportion. The x-axis is the proportion of FDs in the causal graph. The y-axis is the superiority (determined by subtracting the FCI’s score from the XLEARNER’s score) of XLEARNER over FCI.

Since XLEARNER focuses primarily on FDs as the opposite of FCI, we further study how varying proportions of FDs in the causal graph affect XLEARNER’s performance. We report the superiority in terms of varied amounts of FDs in Fig. 7. Overall, we observe an increasing trend in XLEARNER performance (particularly for F1 and recall) as the FD proportion increases. More importantly, we observe that “superiority” increases as FDs increase. Recall, as noted in the caption of Fig. 7, that the superiority is computed by subtracting the FCI’s score from the XLEARNER’s score. Thus, we interpret that XLEARNER gradually outperforms the FCI algorithm with greater degree as the proportion of FDs grows.

In addition to the experiments on synthetic datasets, we also evaluate XLEARNER with the **WEB** dataset (Sec. 4.1). As aforementioned, this real-world dataset lacks a ground-truth causal graph. Evaluating the accuracy of an estimated causal graph is thus challenging, if not impossible. At this step, we involve human experts to assess the correctness of the identified causal relations in the third phase of our user study (i.e., Causal Claim Assessment in Sec. 4.1).

We report the results of our user study in Table 7: first, out of 48 responses (6 participants \times 8 questions), only three (6.3%) suggest that the causal claims are “Not Reasonable,” while 40 responses (83.3%) mark the causal claims as “Reasonable.” It indicates that the causal relations identified by XLEARNER correspond with expert knowledge in the majority of instances. Second, we investigate the claims that have been deemed “Not Reasonable” or “Unsure.” Encouragingly, we find that a notable proportion of causal claims are counter-intuitive yet *correct*. For instance, one causal claim states that “malicious intent would lead to more frequent configuration changes than benign intent.” In

Table 7. User study. C_i stands for the i th causal claim.

	C1	C2	C3	C4	C5	C6	C7	C8
# Reasonable	6	4	4	6	6	4	5	5
# Not Sure	0	2	1	0	0	0	1	1
# Not Reasonable	0	0	1	0	0	2	0	0

the causal claim assessment phase, one expert deemed it “Not Reasonable” and presumed that malicious users would keep a default configuration. In the follow-up session where they shared the independent assessments, this participant was persuaded and confirmed this causal relation as “Reasonable.”

Answer to RQ2: *As reflected by the carefully-designed quantitative experiments and the user study, XLEARNER generates plausible causal graphs that are consistent with expert knowledge.*

4.4 RQ3: XPLAINER Evaluation

Recall the descriptions of **SYN-B** in which different parameters result in datasets with varying degrees of difficulty. In this experiment, we explore the accuracy of XPLAINER on **SYN-B**.

Baseline. We compare XPLAINER with three baselines, namely, Scorpion [E. Wu and Madden 2013], RSExplain [Roy and Suciú 2014] and BOExplain [Lockhart et al. 2021], which use predicates as explanations. Scorpion is an explanation engine for explaining outliers. It uses a metric called influence score to rank explanations, which considers the effect of the explanation between the outlier region and the hold-out region. RSExplain uses the concept of intervention to measure the effectiveness of an explanation. BOExplain is originally designed for explaining black-box machine learning models. When explaining **WHY QUERY**, it employs the inference score and the Bayesian optimization to find the optimal explanation. To launch an apple-to-apple comparison, all baselines are enforced to search over a set of pre-defined causal filters $\{p_1, \dots, p_m\}$ derived from the generation procedure of **SYN-B** (see details in Supplementary Material); all these filters have been confirmed to constitute legitimate causal explanations.

Metric. We use the top-ranked explanation of each baseline as its optimal explanation. We mark a method as “N/A” denoting timeouts (more than one hour to process). We report the F1 Score of filters in the explanation over the ground-truth explanation.

Different Dataset Sizes. To study the scalability of XPLAINER, we generate datasets of varying sizes and report the results in Table 8. Overall, we observe that XPLAINER is more accurate and efficient than all baselines across all the studied settings. This is encouraging and also reasonable, as XPLAINER uses many distinct characteristics of aggregation functions to optimize the search process, while other methods primarily treat them as a “black-box.” Scorpion and BOExplain often produce incomplete explanations, while RSExplain may frequently find extra spurious filters. We presume that this is because the objective function of Scorpion (and also BOExplain) is for explaining anomalies instead of **WHY QUERY**, whereas RSExplain is primarily designed for data provenance. In contrast, explanations provided by XPLAINER are seen as consistent with the ground truth.

XPLAINER is highly efficient particularly for high cardinality regimes, while both Scorpion and RSExplain run out of time when the cardinality exceeds 30 (see the bottom half of Table 8). BOExplain uses Bayesian optimization to search for explanations, and its accuracy downgrades as cardinality increases. Similarly, when iterating different #Rows (the top half of Table 8), XPLAINER also exhibits highly encouraging efficiency: XPLAINER takes on average 0.06 seconds to explain **WHY QUERY** whereas BOExplain (the second best) takes 13.17 seconds. In sum, we interpret from Table 8 that XPLAINER delivers highly encouraging accuracy and efficiency across different settings in comparison with the baseline methods.

Different $\mu^* - \mu$. The difference between μ^* , μ indicates the magnitude of Δ . To study the sensitivity of XPLAINER, we study how well XPLAINER performs under varying differences and compare it with

Table 8. XPLAINER and baselines under various settings. ✓ denotes that F1=1.0 and the best metric is **highlighted**.

#Rows (Cardinality=10)		10K	20K	50K	100K	500K	1M
XPLAINER (SUM)	F1 Score	✓	✓	✓	✓	✓	✓
	Time (sec.)	0.004	0.005	0.007	0.010	0.017	0.019
Scorpion (SUM)	F1 Score	0.5	0.5	0.5	0.5	0.5	0.8
	Time (sec.)	0.68	0.82	1.25	1.93	2.45	2.93
RSExplain (SUM)	F1 Score	0.75	0.75	0.75	0.75	0.75	0.75
	Time (sec.)	0.68	0.83	1.25	1.94	2.44	2.90
BOExplain (SUM)	F1 Score	0.8	0.8	0.5	0.5	0.5	0.8
	Time (sec.)	5.24	5.32	5.62	6.38	9.80	13.53
XPLAINER (AVG)	F1 Score	✓	✓	✓	✓	✓	✓
	Time (sec.)	0.016	0.019	0.026	0.038	0.052	0.063
Scorpion (AVG)	F1 Score	✓	✓	✓	✓	✓	✓
	Time (sec.)	0.59	0.67	0.90	1.29	1.69	2.01
RSExplain (AVG)	F1 Score	0.75	0.75	0.75	0.75	0.75	0.75
	Time (sec.)	0.58	0.66	0.90	1.28	1.68	1.95
BOExplain (AVG)	F1 Score	0.86	✓	0.86	✓	✓	0.8
	Time (sec.)	5.33	5.37	5.56	6.56	8.67	12.62
Cardinality (#Rows=100k)		10	15	20	30	50	100
XPLAINER (SUM)	F1 Score	✓	✓	✓	✓	✓	0.8
	Time (sec.)	0.010	0.014	0.018	0.025	0.040	0.077
Scorpion (SUM)	F1 Score	0.5	0.5	0.5	N/A	N/A	N/A
	Time (sec.)	1.96	16.50	75.72	N/A	N/A	N/A
RSExplain (SUM)	F1 Score	0.75	0.75	0.75	N/A	N/A	N/A
	Time (sec.)	1.95	16.61	75.82	N/A	N/A	N/A
BOExplain (SUM)	F1 Score	✓	0.86	0.86	0.46	0.27	0.15
	Time (sec.)	6.28	8.71	11.17	15.44	25.44	48.73
XPLAINER (AVG)	F1 Score	✓	✓	✓	✓	✓	✓
	Time (sec.)	0.038	0.060	0.082	0.124	0.211	0.426
Scorpion (AVG)	F1 Score	✓	✓	✓	N/A	N/A	N/A
	Time (sec.)	1.27	10.58	47.90	N/A	N/A	N/A
RSExplain (AVG)	F1 Score	0.75	0.75	0.75	N/A	N/A	N/A
	Time (sec.)	1.28	10.59	47.91	N/A	N/A	N/A
BOExplain (AVG)	F1 Score	✓	0.86	0.5	0.5	0.27	0.14
	Time (sec.)	5.87	8.23	10.44	15.00	24.35	46.35

Table 9. XPLAINER and baselines with different $\mu^* - \mu$. ✓ denotes that the result is identical to the ground truth (F1=1.0).

$\mu - \mu^*$	5	10	15	30	50	100
XPLAINER (SUM)	0.86	✓	✓	✓	✓	✓
Scorpion (SUM)	0.50	0.50	0.50	0.50	0.50	0.50
RSExplain (SUM)	0.75	0.75	0.75	0.75	0.75	0.75
BOExplain (SUM)	0.50	0.86	0.80	0.80	0.80	✓
XPLAINER (AVG)	✓	✓	✓	✓	✓	✓
Scorpion (AVG)	0.80	✓	✓	✓	✓	✓
RSExplain (AVG)	0.75	0.75	0.75	0.75	0.75	0.75
BOExplain (AVG)	0.80	✓	0.86	0.86	0.80	✓

baselines in Table 9. To clarify, $\mu^* - \mu = 5$ and $\mu^* - \mu = 10$ denote two relatively more challenging settings in Table 9, given the very subtle differences. On SUM aggregates, we find that all methods have difficulties in identifying explanations in those two challenging settings; still, XPLAINER yields the best results for both settings. Even on the most challenging setting ($\mu^* - \mu = 5$), XPLAINER finds highly accurate explanations whereas RSExplain is less accurate.

On AVG aggregates, XPLAINER and Scorpion both perform well on identifying the ground-truth explanations; XPLAINER is slightly better particularly for the most challenging setting when $\mu^* - \mu = 5$. Nevertheless, RSExplain and BOExplain are less accurate on AVG. Overall, we conclude that

XPLAINER is more robust to subtle data differences while all other methods have difficulties in such challenging settings. We omit reporting the processing time here, since it has already been evidently explored in Table 8.

Tightness of Approximation. In Sec. 3.3, we show the approximation of minimal contingency for computing responsibilities under SUM and AVG. In the following, we compare the tightness of the responsibility $\hat{\rho}$ computed by $\bar{P} = P^C - P$ to the true responsibility ρ computed by the minimal contingency P_{\min} via brute-force search. The approximation error is computed as $E = \frac{|\hat{\rho} - \rho|}{\rho}$. Recall that we craft three filters that form the counterfactual cause in each dataset. In SUM, we can craft six $\binom{3}{2}$ actual causes from the three filters. In AVG, since the canonical predicate of AVG only supports the first k filters as actual causes and the rest as contingencies, we pick the top-1 and top-2 filters as two actual causes and repeat the experiments on three datasets (2×3 in total). On the six actual causes of SUM aggregates, we find that the brute-force algorithm is $253.3\times$ slower than our approximated solution. More importantly, the approximation error is highly negligible with an average of 0.007. We also observe that the approximation error on AVG is slightly higher (0.066) and that our heuristics-based solution is $27.3\times$ faster. This result is reasonable, as the heuristics-based solution does not provide guarantees of accuracy and requires more queries than SUM.

Answer to RQ3: *XPLAINER shows high scalability to large datasets and also accurately generates explanations in very difficult settings. On a mild cost of precision, two approximation solutions of XPLAINER substantially improve efficiency.*

5 DISCUSSION

FD in Noisy Data. XINSIGHT only considers *deterministic FDs*. As illustrated in Ex. 3.1, taking deterministic FDs into account eliminates faithfulness violations. However, when the data is noisy, the FDs may be stochastic (e.g., probabilistic interpretation of FDs [Y. Zhang et al. 2020]), which is currently not considered in XINSIGHT. We clarify that considering only deterministic FDs deems a common setup shared by relevant works in this field [Ding, Liu, et al. 2020; Mabrouk et al. 2014]. It remains unclear how noisy FDs may impact faithfulness. We leave this for future exploration.

Acquiring Causal Knowledge. Inferring causal relations is difficult. Typically, it needs a combination of domain knowledge [Andrews et al. 2020], randomized experiments [Triantafillou and Tsamardinos 2015] and causal discovery [Dai et al. 2021; Ding, Liu, et al. 2020; Ma, Ding, Dai, et al. 2022; Spirtes et al. 2000; J. Zhang 2008]. XINSIGHT performs causal discovery from observational data due to its simplicity. Nevertheless, we envision users of XINSIGHT can combine additional sources for acquiring more accurate causal knowledge. In this paper, we explain several key obstacles of applying causal discovery to real-world data, including causal insufficiency [J. Zhang 2008] and FD-induced faithfulness violations [Ding, Liu, et al. 2020; Mabrouk et al. 2014]. XLEARNER, for the first time, simultaneously addresses all of them.

Other Forms of Explanations. Currently, XINSIGHT employs predicates as the content of explanations, which is general enough for common data analysis scenarios. However, in some cases, explanations may be formed by the number of records in a database [*Discover Insights Faster with Explain Data 2022*] or counterbalances [Miao et al. 2019]. Furthermore, when explaining changes in a whole time series [Chen and Huang 2021], XINSIGHT may be not applicable. We leave integrating XINSIGHT into these scenarios for future work.

6 RELATED WORK

Data Explanation. Explaining an unexpected query outcome in database is a crucial phase in the lifecycle of data analysis. In general, an explanation aims to provide certain forms of patterns that lead to the unexpected query outcome. Such patterns may be a set of predicates [Abuzaid et al. 2021;

Bailis et al. 2017; Roy and Suciu 2014; E. Wu and Madden 2013], tuples [Meliou, Roy, et al. 2014], or counterbalances [Miao et al. 2019]. Scorpion is the most relevant work for XINSIGHT, which also provides explanations to aggregated queries [E. Wu and Madden 2013]. In particular, it employs an influence score to quantify explanations and features a set of optimizations to reduce the cost of explanation search. Recently, many tools have attempted to enhance explanations with additional knowledge (e.g., join tables) about the underlying data. However, such additional knowledge does not imply causation – top ranked explanations could be rated low by human participants due to a lack of causal semantics [C. Li et al. 2021]. These observations evidently show the necessity of integrating causality into XDA.

Causality in Database. Most works related to causality analysis in the database is on the basis of Halpern’s seminal framework on actual causality [Halpern 2016; Halpern and Pearl 2005]. It provides an elegant and natural way to reason about input-output relations. Its results not only highlight the output’s cause, but also provide a contingency describing how it is triggered. The adaption of actual causality in the database (i.e., DB causality) is widely used for data provenance [Meliou, Gatterbauer, Moore, et al. 2010], data explanation [Roy and Suciu 2014] and debugging [Fariha et al. 2020; Ji et al. 2022; Meliou, Gatterbauer, Nath, et al. 2011; Yoon et al. 2016]. However, it has limitations when applied alone. On the one hand, as noted in [Glavic et al. 2021], DB causality does not necessarily imply *true causation*. Indeed, it assumes that causal knowledge is already known, and focuses solely on quantitative explanations. On the other hand, considerable adaptations are required to make it applicable to XDA scenarios, as discussed in Sec. 3.3. We also notice other methods for quantifying explanations, such as sufficient score, necessity score, and average causal effect [Salimi, Parikh, et al. 2020; Watson et al. 2021]. Despite their usefulness, we design XPLAINER on top of actual causality because it is more understandable and general. Furthermore, without prior causal knowledge, none of these methods can imply true causation.

XDA vs. XAI. We note that XAI (explainable artificial intelligence) is parallel and complementary to XDA. Through the lens of data analysis, XAI aims to explain a prediction or model [Galhotra et al. 2021; Pradhan et al. 2022], while XDA enhances EDA for understanding data facts. In addition, we also observe a line of research [Flokas et al. 2022; Y. Li et al. 2022; Lin et al. 2022; W. Wu et al. 2020] that identifies a subset of (training) data that is responsible for a prediction. While this line of research shares similar output format with XINSIGHT, it is essentially for explaining how model predictions are influenced by training/test data, a scenario that is orthogonal to our research.

7 CONCLUSION

This paper advocates XDA, a concept that ships comprehensive and in-depth explainability toward EDA. XDA offers either causal or non-causal explanations for EDA outcomes, from both quantitative and qualitative perspectives. We have also presented the design of XINSIGHT, a production framework for XDA over databases. Experiments and human evaluations reveal that XINSIGHT manifests highly encouraging explanation capabilities. XPLAINER has been integrated into Microsoft Power BI to explain increase/decrease in data.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions. The author would also like to thank Siwen Zhu, Haidong Zhang, Zhitao Hou, Ruming Wang, Ziyu Wang, Long Ding, Kai Zhang, Jon Kay, and Dingkun Xie for helpful discussions and all our participants in the user study for their valuable feedback. The authors at HKUST were supported in part by RGC RMGS under the contract RMGS22EG02.

REFERENCES

- Firas Abuzaid et al. 2021. “Diff: a relational interface for large-scale data explanation.” *The VLDB Journal*, 30, 1, 45–70.
- Bryan Andrews, Peter Spirtes, and Gregory F Cooper. 2020. “On the completeness of causal discovery in the presence of latent confounding with tiered background knowledge.” In: *International Conference on Artificial Intelligence and Statistics*. PMLR, 4002–4011.
- Nuno Antonio, Ana de Almeida, and Luis Nunes. 2019. “Hotel booking demand datasets.” *Data in brief*, 22, 41–49.
- Peter Bailis, Edward Gan, Samuel Madden, Deepak Narayanan, Kexin Rong, and Sahaana Suri. 2017. “Macrobase: Prioritizing attention in fast data.” In: *Proceedings of the 2017 ACM International Conference on Management of Data*, 541–556.
- Elias Bareinboim and Judea Pearl. 2012. “Controlling selection bias in causal inference.” In: *Artificial Intelligence and Statistics*. PMLR, 100–108.
- Leopoldo Bertossi. 2020. “Score-Based Explanations in Data Management and Machine Learning.” In: *International Conference on Scalable Uncertainty Management*. Springer, 17–31.
- Leopoldo Bertossi, Jordan Li, Maximilian Schleich, Dan Suciu, and Zografoula Vagena. 2020. “Causality-based explanation of classification outcomes.” In: *Proceedings of the Fourth International Workshop on Data Management for End-to-End Machine Learning*, 1–10.
- Michael Buckland. 2017. *Information and society*. MIT Press.
- Yiru Chen and Silu Huang. 2021. “TSExplain: Surfacing Evolving Explanations for Time Series.” In: *Proceedings of the 2021 International Conference on Management of Data*, 2686–2690.
- Haoyue Dai, Rui Ding, Yuanyuan Jiang, Shi Han, and Dongmei Zhang. 2021. “ML4C: Seeing Causality Through Latent Vicinity.” *arXiv preprint arXiv:2110.00637*.
- Rui Ding, Shi Han, Yong Xu, Haidong Zhang, and Dongmei Zhang. 2019. “Quickinsights: Quick and automatic discovery of insights from multi-dimensional data.” In: *Proceedings of the 2019 International Conference on Management of Data*, 317–332.
- Rui Ding, Yanzhi Liu, Jingjing Tian, Zhouyu Fu, Shi Han, and Dongmei Zhang. 2020. “Reliable and Efficient Anytime Skeleton Learning.” In: *Proceedings of the AAAI Conference on Artificial Intelligence* 06. Vol. 34, 10101–10109.
- Discover Insights Faster with Explain Data*. https://help.tableau.com/current/pro/desktop/en-us/explain_data.htm. (2022).
- Anna Fariha, Suman Nath, and Alexandra Meliou. 2020. “Causality-guided adaptive interventional debugging.” In: *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 431–446.
- Lampros Flokas, Weiyuan Wu, Yejia Liu, Jiannan Wang, Nakul Verma, and Eugene Wu. 2022. “Complaint-Driven Training Data Debugging at Interactive Speeds.” In: *Proceedings of the 2022 International Conference on Management of Data*, 369–383.
- Sainyam Galhotra, Romila Pradhan, and Babak Salimi. 2021. “Explaining black-box algorithms using probabilistic contrastive counterfactuals.” In: *Proceedings of the 2021 International Conference on Management of Data*, 577–590.
- Boris Glavic, Alexandra Meliou, Sudeepa Roy, et al. 2021. “Trends in Explanations: Understanding and Debugging Data-driven Systems.” *Foundations and Trends® in Databases*, 11, 3, 226–318.
- Joseph Y Halpern. 2016. *Actual causality*. MIT Press.
- Joseph Y Halpern and Judea Pearl. 2005. “Causes and explanations: A structural-model approach. Part I: Causes.” *The British journal for the philosophy of science*, 56, 4, 843–887.
- Zhenlan Ji, Pingchuan Ma, and Shuai Wang. 2022. “PerfCE: Performance Debugging on Databases with Chaos Engineering-Enhanced Causality Analysis.” *arXiv preprint arXiv:2207.08369*.
- Frank C Keil. 2006. “Explanation and understanding.” *Annu. Rev. Psychol.*, 57, 227–254.
- Marc Lange. 2016. *Because Without Cause: Non-Casual Explanations In Science and Mathematics*. Oxford University Press.
- Po-Ming Law, Leo Yu-Ho Lo, Alex Endert, John Stasko, and Huamin Qu. 2021. “Causal Perception in Question-Answering Systems.” In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 1–15.
- Chenjie Li, Zhengjie Miao, Qitian Zeng, Boris Glavic, and Sudeepa Roy. 2021. “Putting Things into Context: Rich Explanations for Query Answers using Join Graphs (extended version).” *arXiv preprint arXiv:2103.15797*.
- Yanhui Li, Linghan Meng, Lin Chen, Li Yu, Di Wu, Yuming Zhou, and Baowen Xu. 2022. “Training data debugging for the fairness of machine learning software.” In: *Proceedings of the 44th International Conference on Software Engineering*, 2215–2227.
- Jinkun Lin, Anqi Zhang, Mathias Léculuyer, Jinyang Li, Aurojit Panda, and Siddhartha Sen. 2022. “Measuring the Effect of Training Data on Deep Learning Predictions via Randomized Experiments.” In: *International Conference on Machine Learning*. PMLR, 13468–13504.
- Brandon Lockhart, Jinglin Peng, Weiyuan Wu, Jiannan Wang, and Eugene Wu. 2021. “Explaining inference queries with bayesian optimization.” *Proceedings of the VLDB Endowment*, 14, 11, 2576–2585.
- Pingchuan Ma, Rui Ding, Haoyue Dai, Yuanyuan Jiang, Shuai Wang, Shi Han, and Dongmei Zhang. 2022. “ML4S: Learning Causal Skeleton from Vicinal Graphs.” In: *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*.

- Pingchuan Ma, Rui Ding, Shi Han, and Dongmei Zhang. 2021. "MetaInsight: Automatic Discovery of Structured Knowledge for Exploratory Data Analysis." In: *Proceedings of the 2021 International Conference on Management of Data*, 1262–1274.
- Ahmed Mabrouk, Christophe Gonzales, Karine Jabet-Chevalier, and Eric Chojnacki. 2014. "An efficient Bayesian network structure learning algorithm in the presence of deterministic relations." In: *ECAI 2014*. IOS Press, 567–572.
- Alexandra Meliou, Wolfgang Gatterbauer, Joseph Y Halpern, Christoph Koch, Katherine F Moore, and Dan Suciu. 2010. "Causality in databases." *IEEE Data Engineering Bulletin*, 33, ARTICLE, 59–67.
- 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, 34–45.
- Alexandra Meliou, Wolfgang Gatterbauer, Suman Nath, and Dan Suciu. 2011. "Tracing data errors with view-conditioned causality." In: *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, 505–516.
- Alexandra Meliou, Sudeepa Roy, and Dan Suciu. 2014. "Causality and explanations in databases." *Proceedings of the VLDB Endowment*, 7, 13, 1715–1716.
- 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.
- Microsoft. 2022a. *Microsoft/reliableAI*. <https://github.com/microsoft/reliableAI>. (2022).
- Microsoft. 2022b. *Use the Analyze feature to explain fluctuations in report visuals*. <https://learn.microsoft.com/en-us/power-bi/consumer/end-user-analyze-visuals>. (2022).
- Tova Milo and Amit Somech. 2020. "Automating exploratory data analysis via machine learning: An overview." In: *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 2617–2622.
- Gregory L Murphy and Douglas L Medin. 1985. "The role of theories in conceptual coherence." *Psychological review*, 92, 3, 289.
- Judea Pearl. 2009a. "Causal inference in statistics: An overview." *Statistics Surveys*, 3, none, 96–146.
- Judea Pearl. 2009b. *Causality*. Cambridge university press.
- Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2011. "Causal inference on discrete data using additive noise models." *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33, 12, 2436–2450.
- Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. 2017. *Elements of causal inference: foundations and learning algorithms*. The MIT Press.
- Mark Povich and Carl F Craver. 2018. *Because without Cause: Non-Causal Explanations in Science and Mathematics*. (2018).
- Romila Pradhan, Aditya Lahiri, Sainyam Galhotra, and Babak Salimi. 2022. "Explainable AI: Foundations, Applications, Opportunities for Data Management Research." In: *Proceedings of the 2022 International Conference on Management of Data*, 2452–2457.
- 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.
- Babak Salimi, Corey Cole, Dan RK Ports, and Dan Suciu. 2017. "Zaliql: causal inference from observational data at scale." *Proceedings of the VLDB Endowment*, 10, 12, 1957–1960.
- 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.
- Richard Scheines, Peter Spirtes, and Clark Glymour. 1991. "A qualitative approach to causal modeling." In: *Qualitative simulation modeling and analysis*. Springer, 72–97.
- Peter Spirtes, Clark N Glymour, Richard Scheines, and David Heckerman. 2000. *Causation, prediction, and search*. MIT press.
- Sofia Triantafyllou and Ioannis Tsamardinos. 2015. "Constraint-based causal discovery from multiple interventions over overlapping variable sets." *The Journal of Machine Learning Research*, 16, 1, 2147–2205.
- David S Watson, Limor Gultchin, Ankur Taly, and Luciano Floridi. 2021. "Local explanations via necessity and sufficiency: Unifying theory and practice." In: *Uncertainty in Artificial Intelligence*. PMLR, 1382–1392.
- Eugene Wu and Samuel Madden. 2013. "Scorpion: Explaining Away Outliers in Aggregate Queries." *Proceedings of the VLDB Endowment*, 6, 8.
- Weiyuan Wu, Lampros Flokas, Eugene Wu, and Jiannan Wang. 2020. "Complaint-driven training data debugging for query 2.0." In: *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 1317–1334.
- Dong Young Yoon, Ning Niu, and Barzan Mozafari. 2016. "Dbsherlock: A performance diagnostic tool for transactional databases." In: *Proceedings of the 2016 International Conference on Management of Data*, 1599–1614.
- Jiji Zhang. 2008. "On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias." *Artificial Intelligence*, 172, 16-17, 1873–1896.
- Yunjia Zhang, Zhihan Guo, and Theodoros Rekatsinas. 2020. "A statistical perspective on discovering functional dependencies in noisy data." In: *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*, 861–876.

Received October 2022; revised January 2023; accepted February 2023