Using Regression to Explain Cube Measures

Matteo Francia[1], Stefano Rizzi[1,*] and Patrick Marcel[2]

[1]DISI - University of Bologna, Bologna, Italy
[2]LIFAT - University of Tours, Blois, France

Abstract

The Intentional Analytics Model (IAM) has been devised to couple OLAP and analytics by (i) letting users express their analysis intentions on multidimensional data cubes and (ii) returning enhanced cubes, i.e., multidimensional data annotated with knowledge insights in the form of models (e.g., correlations). Five intention operators were proposed to this end; of these, describe and assess have been investigated in previous papers. In this work we enrich the IAM picture by focusing on the explain operator, whose goal is to provide an answer to the user asking "why does measure m show these values?". Specifically, we propose a syntax for the operator and discuss how enhanced cubes are built by (i) finding the polynomials that best approximate the relationship between m and the other cube measures, and (ii) highlighting the most interesting one. Finally, we test the operator implementation in terms of efficiency.

Keywords

Data cube, OLAP, Analytics, Correlation

1. Introduction

Despite the huge success of the OLAP paradigm, it is now clear that this paradigm, alone, does no longer meet the sophisticated requirements of new-generation decision makers. Among the directions taken by research to enhance OLAP, the *Intentional Analytics Model* (IAM) suggests to couple it with analytics [1]. The IAM approach relies on two main ideas: (i) users explore the data space by expressing their analysis *intentions* and (ii) in return they receive both multidimensional data and knowledge insights in the form of models. To achieve (i) five intention operators were proposed, namely, describe (describes one or more cube measures at some aggregation level, possibly focused on some level members), assess (judges one or more cube measures with reference to some benchmark), explain (reveals the reason behind the values of a measure, for instance by correlating it with other measures), predict (shows data not in the original cubes, derived for instance with regression), and suggest (shows data similar to those the current user, or similar users, have been interested in). As to (ii), first-class citizens of the IAM are *enhanced cubes*, defined as multidimensional cubes coupled with *highlights*, i.e., interesting components of models automatically extracted from cubes. An overview of the approach is shown in Figure 1. Noticeably, having different models automatically computed and evaluated in terms of their interest relieves the user from the time-wasting effort of trying different possibilities.

SEBD 2023: 31st Symposium on Advanced Database System, July 02–05, 2023, Galzignano Terme, Padua, Italy
*Corresponding author.
m.francia@unibo.it (M. Francia); stefano.rizzi@unibo.it (S. Rizzi); patrick.marcel@univ-tours.fr (P. Marcel)
0000-0002-0805-1051 (M. Francia); 0000-0002-4617-217X (S. Rizzi); 0000-0003-3171-1174 (P. Marcel)

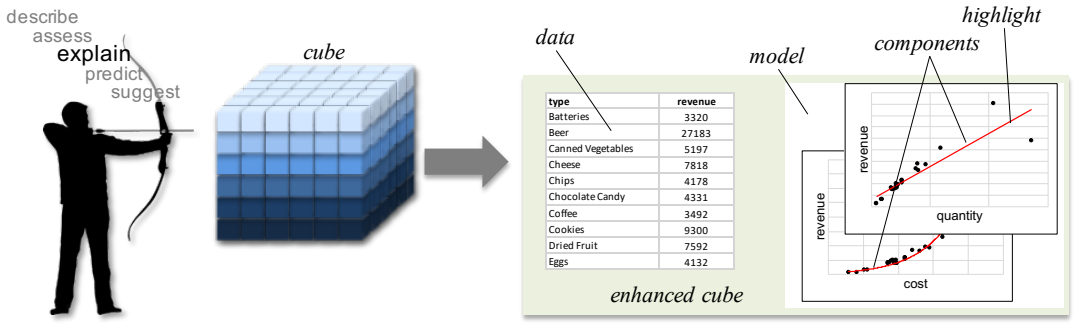

Figure 1: The IAM approach

Among the five intention operators, describe and assess have been investigated in previous papers [2, 3]. In this paper we enrich the IAM picture by focusing on the explain operator. An *explanation* is essentially a description of causation for an observed phenomenon; in practice, it answers the *why?* question for that phenomenon by providing a causal model for it [4]. In our context, we concentrate on providing explanation models for a measure the user is observing; thus, the goal of the explain operator will be to provide an answer to the user asking "why does measure m show these values?".

As envisioned in [1], several types of models can be used to this end. To give a proof-of-concept for explain, in this paper we restrict to the simplest model type, the one that establishes a polynomial relationship between m and m'.

Example 1. *Let a* SALES *cube be given, and let the user's intention be*

with SALES explain revenue by type for year='2022'

First, the subset of facts for 2022 are selected from the SALES *cube and aggregated by product type (in OLAP terms, a slice-and-dice and a roll-up operator are applied). Then, regression analysis is used to compare the* revenue *measure with each other cube measure and find the polynomials that best approximates their relationship. Finally, a measure of interest is computed for the components (i.e., for the polynomials) obtained, and the most interesting one is shown to the user (in Figure 1, the one showing that* revenue *is roughly proportional to* quantity*).*

The paper outline is as follows. After introducing models and enhanced cubes in Section 2, in Section 3 we give the syntax of explain and illustrate how models are built. Then, in Section 4 we explain how enhanced cubes are visualized. Finally, in Section 5 we discuss the related literature and in Section 6 we test the operator implementation and draw the conclusions.

2. Enhanced cubes

Models are concise, information-rich knowledge artifacts that represent relationships hiding in the cube facts. The possible models range from simple functions and measure correlations to more elaborate techniques such as decision trees, clusterings, etc. A model is bound to (i.e.,

is computed over the levels/measures of) one cube, and is made of a set of components, each component being a specific relationship among cube facts.

Definition 1 (Hierarchy and Cube Schema). *A* hierarchy *is a pair $h = (L_h, \succeq_h)$ where L_h is a set of categorical* levels*, each coupled with a domain including a set of members, and $\succeq_h$ is a* roll-up *total order of L_h. The top level of $\succeq_h$ is called* dimension*. A* cube schema *is a pair $\mathcal{C} = (H, M)$ where H is a set of hierarchies and M is a set of numerical measures, each coupled with one aggregation operator.*

Example 2. *For our working example we will use the* SALES *cube, which includes three hierarchies and three measures. Formally,* $\mathsf{SALES} = (H, M)$ *with*

$$H = \{h_{\mathsf{Date}}, h_{\mathsf{Product}}, h_{\mathsf{Store}}\};\ M = \{\mathsf{quantity}, \mathsf{revenue}, \mathsf{cost}\};$$
$$\mathsf{date} \succeq \mathsf{month} \succeq \mathsf{year};\ \mathsf{product} \succeq \mathsf{type} \succeq \mathsf{category};\ \mathsf{store} \succeq \mathsf{city} \succeq \mathsf{country}$$

Aggregation is the basic mechanism to query cubes, and it is captured by the following definition of group-by set.

Definition 2 (Group-by Set and Coordinate). *Given cube schema $\mathcal{C} = (H, M)$, a* group-by set *of $\mathcal{C}$ is a set of levels, at most one from each hierarchy of H. The partial order induced on the set of all group-by sets of $\mathcal{C}$ by the roll-up orders of the hierarchies in H, is denoted with $\succeq_H$. A* coordinate *of group-by set G is a tuple of members, one for each level of G.*

Example 3. *Two group-by sets of* SALES *are $G_1 = \{\mathsf{date}, \mathsf{type}, \mathsf{country}\}$ and $G_2 = \{\mathsf{month}, \mathsf{category}\}$, where $G_1 \succeq_H G_2$. G_1 aggregates sales by date, product type, and store country, G_2 by month and category. Example of coordinates of the two group-by sets are, respectively, $\gamma_1 = \langle$2022-04-15, Fresh Fruit, Italy$\rangle$ and $\gamma_2 = \langle$2022-04, Fruit$\rangle$.*

The instances of a cube schema are called cubes and are defined as follows.

Definition 3 (Cube). *A* cube *over $\mathcal{C}$ is a triple $C = (G_C, M_C, \omega_C)$ where G_C is a group-by set of $\mathcal{C}$, $M_C \subseteq M$, and ω_C is a partial function that maps the coordinates of G_C to a numerical value for each measure $m \in M_C$.*

Each coordinate γ that participates in ω_C, with its associated measure values, is called a *fact* of C. With a slight abuse of notation, we will write $\gamma \in C$ to state that γ is a fact of C. The value taken by measure m in the fact corresponding to γ is denoted as $\gamma.m$. A cube whose group-by set G_C includes all and only the dimensions of the hierarchies in H and such that $M_C = M$, is called a *base cube*, the others are called *derived cubes*. In OLAP terms, a derived cube is the result of either a roll-up, a slice-and-dice, or a projection made over a base cube; this is formalized as follows.

Definition 4 (Cube Query). *A* query *over cube schema $\mathcal{C}$ is a triple $q = (G_q, P_q, M_q)$ where G_q is a group-by set of H, P_q is a (possibly empty) set of selection predicates each expressed over one level of H, and $M_q \subseteq M$.*

Example 4. *The cube query over* SALES *used in Example 1 is* $q = (G_q, P_q, M_q)$ *where* $G_q = \{\text{type}\}$, $P_q = \{\text{year} = \text{'2022'}\}$, *and* $M_q = \{\text{revenue}\}$. *A coordinate of the resulting cube is* $\langle \text{Batteries} \rangle$ *with associated value 3320 for* revenue.

Definition 5 (Model). *A* model *is a tuple* $\mathcal{M} = (t, alg, C, m, In, Out)$ *where: (i)* t *is the model type; (ii)* alg *is the algorithm used to compute* Out*; (iii)* C *is the cube to which the model is bound; (iv)* m *is the measure in* C *to be explained; (v)* In *is the tuple of levels/measures of* C *and parameter values supplied to* alg *to compute the model; (vi)* Out *is the set of model components.*

In this paper, to give a proof-of-concept of the explain operator, we restrict to consider a single type of model, namely, the one that establishes a polynomial relationship between two measures via regression analysis. In this case, In is the set of measures whose relationship with m is described; besides, each component $c_i \in Out$ shows the relationship of m with one measure $m_i \in In$.

Definition 6 (Component). *For a model of type polynomial, a* component c_i *is a triple* $c_i = (m_i, d_i, coeff_i)$ *where: (i)* m_i *is the measure in* C *whose relationship with* m *is described; (ii)* d_i *is the degree of the polynomial used to describe the relationship between* m *and* m_i*; (iii)* $coeff_i$ *is an array of the* $d_i + 1$ *coefficients of the polynomial* $\alpha^{d_i}(m_i)$ *that best approximates* m *with reference to the facts in* C.

Example 5. *A possible model over the* SALES *cube is characterized by*

$$t = \textit{regression};\ alg = \textit{Polyfit};\ C = \text{SALES};$$
$$m = \text{revenue};\ In = \{\text{quantity}, \text{cost}\};\ Out = \{c_1, c_2\};$$
$$c_1 = (\text{quantity}, 1, [0.98, 4909.52]);\ c_2 = (\text{cost}, 2, [1.1, 22.78, 1409.33])$$

According to this model, the relationships of revenue *with* quantity *and* cost *are described, respectively, as*

$$\text{revenue} = \alpha^1(\text{quantity}) = 0.98 \cdot \text{quantity} + 4909.52$$
$$\text{revenue} = \alpha^2(\text{cost}) = 1.1 \cdot \text{cost}^2 - 22.78 \cdot \text{cost} + 1409.33$$

As the last step in the IAM approach, cube C is enhanced by associating it with a set of models bound to C and with a *highlight*, i.e., with the most interesting model component:

Definition 7 (Enhanced cube). *An* enhanced cube E *is a triple of a cube* C*, a set of models* $\{\mathcal{M}_1, \ldots, \mathcal{M}_z\}$ *bound to* C*, and a highlight* $\overline{c} = argmax_{\{c_i \in \bigcup_{j=1}^{z} Out_j\}}(interest(c_i))$.

In our scenario only polynomial models are considered, so an enhanced cube includes a single model with one component for each measure in In. Let c_i be the component associated to m_i; we evaluate the interest of c_i, $interest(c_i)$, as the *coefficient of determination* R2 [5], which measures how well the values of m are replicated by the model in m_i via the variation in the dependent variable m that is predictable from the independent variable m_i. The better the model, the closer the value of R2 to 1.

Example 6. *With reference to Example 5, it is* $interest(c_1) = 0.61$ *and* $interest(c_2) = 0.99$. *Thus, the highlight is* c_2.

3. The explain operator

The explain operator provides an answer to the user asking "why is this happening?" "why does measure m show these values?" by describing the relationship between m and the other cube measures, possibly focused on one or more level members, at some given granularity. The cube is enhanced by showing the polynomials that best approximate these relationships, with a highlight on the most interesting one.

3.1. Syntax

Let C_0 be a base cube over cube schema $\mathcal{C} = (H, M)$. The syntax for explain is (optional parts are in brackets):

with C_0 explain m [for P] by $l_1, \ldots, l_n$ [against m_1 [degree d_1], …, m_r [degree d_r]]

where $m \in M$ is a measure of $\mathcal{C}$; P is a set of selection predicates, each over one level of H; $\{l_1, \ldots, l_n\}$ is a group-by set of H; $m_1, \ldots, m_r$ are measures of M (different from m); the d_i's ($d_i > 0$) are integers denoting, for each m_i, the degree of the polynomial to be computed.

Example 7. *Examples of* explain *intentions on the* SALES *cube are, besides the one in Example 1,*

```
with SALES explain cost by date, product against quantity degree 1
with SALES explain revenue by year against cost
```

3.2. Semantics

The execution plan corresponding to a fully-specified intention, i.e., one where all optional clauses have been specified, is as follows:

1. Execute query $q = (G_q, P_q, M_q)$, where $G_q = \{l_1, \ldots, l_n\}$, $P_q = P$, and $M_q = \{m, m_1, \ldots, m_r\}$. Let $C = q(C_0)$ be the cube resulting from the execution of q over C_0.
2. Compute model $\mathcal{M} = (\text{polynomial}, \text{Polyfit}, C, m, \{m_1, \ldots, m_r\}, \{c_1, \ldots, c_r\})$, where $c_i = (m_i, d_i, \mathit{coeff}_i)$. The best approximating polynomial of degree d_i is determined via *ordinary least squares* [6], which finds —with a complexity of $O(d_i^2|C|)$, where $|C|$ is the number of facts in C— the polynomial coefficients that minimize the sum of squared errors between each m_i (independent variables) and m (dependent variable).
3. For each c_i compute $interest(c_i)$.
4. Find the highlight $\overline{c} = argmax_{\{1 \le i \le r\}}(interest(c_i))$.
5. Return the enhanced cube E consisting of C, $\{\mathcal{M}\}$, and $\overline{c}$.

Partially-specified intentions are interpreted as follows:

- If the for clause is not specified, we consider $P_q = TRUE$.
- If the against clause is not specified, a component is created for each measure in M (except m).
- If the degree clause is not specified for one or more measures, the value of d_i is determined automatically by polynomial fitting [7].

Example 8. *The first intention in Example 7 is executed by first computing the derived cube C that aggregates* SALES *by* {date, product} *and projects on measures* cost *and* quantity. *Then, a model $\mathcal{M}$ including a single component c (a linear polynomial approximating* cost *in function of* quantity*) is determined. Finally, the enhanced cube including C, $\mathcal{M}$, and the highlight c is returned.*

4. Visualizing enhanced cubes

As previously done for the describe and assess IAM operators, to give an effective visualization of the enhanced cubes built for explain intentions we couple a text-based representation (a pivot table and a ranked component list) with a graphical one (a chart) and with an ad-hoc interaction paradigm. Specifically, the visualization of enhanced cube $E = (C, \mathcal{M}, \overline{c})$ relies on three distinct but inter-related areas: a *table* area that shows the facts of C using a pivot table; a *component* area that shows a list of model components (i.e., approximating polynomials) sorted by their interest, with $\overline{c}$ at the top; a *chart* area that uses a scatter chart to display, for each component c_i of $\mathcal{M}$, the relationship between m and m_i as well as the function plotting the approximating polynomial.

The interaction paradigm we adopt is component-driven: clicking on one component c_i in the component area leads to show the corresponding approximating polynomial in the chart area. The highlight is selected by default.

Example 9. *Figure 2 shows the visualization obtained when the intention in Example 1 is formulated. On the left, the table area; on the right, the chart area; in the middle, the component area. The highlight is a quadratic polynomial that approximates* revenue *in function of* cost*, so the chart area shows the relationship between these two measures and the approximating parabola.*

5. Related work

The idea of coupling data and analytical models was born in the 90's with inductive databases, where data were coupled with patterns meant as generalizations of the data [8]. Later on, data-to-model unification was addressed in MauveDB [9], which provides a language for specifying model-based views of data using common statistical models. More recently, Northstar [10] has been proposed as a system to support interactive data science by enabling users to switch between data exploration and model building. Finally, the coupling of data and models is at the core of the IAM vision [1], on which this paper relies.

The coupling of the OLAP paradigm and data mining to create an approach where concise patterns are extracted from multidimensional data for user's evaluation, was the goal of some

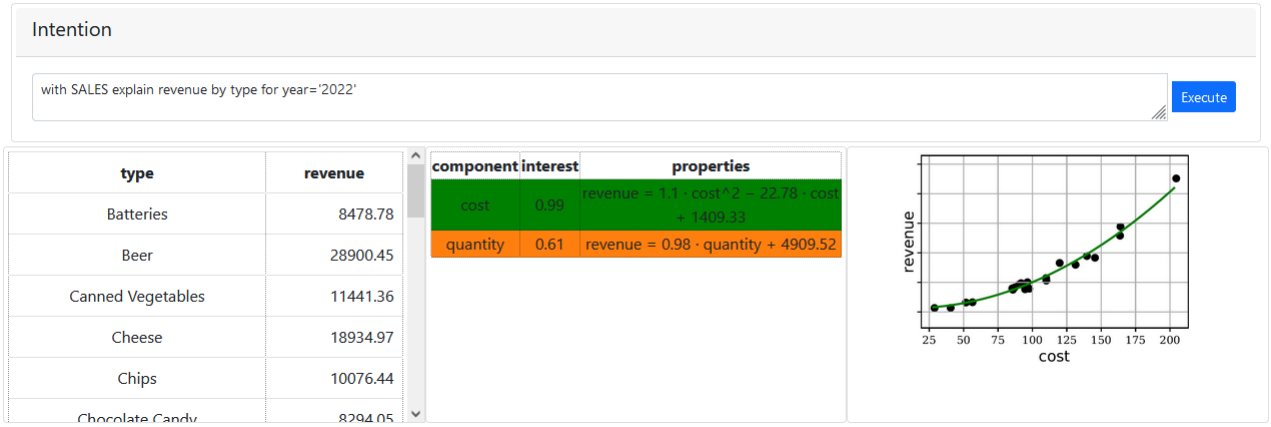

Figure 2: The visualization obtained for the intention in Example 1

approaches commonly labeled as OLAM [11]. In this context, k-means clustering is used in [12] to dynamically create semantically-rich aggregates of facts other than those statically provided by dimension hierarchies. Other operators that enrich data with knowledge extraction results are DIFF [13] and RELAX [14]. Finally, in [15] the OLAP paradigm is reused to explore prediction cubes, i.e., cubes where each fact summarizes a predictive model trained on the data corresponding to that fact.

In an attempt to develop tools for helping users understand data, there have been several efforts in the research community to devise techniques to model explanations for observations made on data; see [16] for a comprehensive analysis of the literature and of the trends in explanation. A common way to give an explanation is to identify the actual *cause* of the observed outcome. Given the result of a database query, which database tuple(s) caused that output to the query? One way to answer this question is to quantify the contribution that each tuple has to the result and identify the tuples with the highest contributions [17, 18]; the intuition is that tuples with high contribution tend to be interesting explanations to query answers. Similarly, in [19] causality is defined in terms of *intervention*: an input is a cause to an output if we can affect the output by changing the value of that input. Causality poses additional challenges when the query contains aggregates [17], as in our scenario. The DIFF operator [13] tells users why a given aggregated quantity is lower or higher in one cube fact than in another by returning the set of rows that best explains the observed increase or decrease at the aggregated level. In Scorpion [20], outliers are explained in terms of properties of the tuples used to compute these outliers, while [21] explains outliers in aggregation queries through counter-balancing. LensXPlain [22] explains why some measure value is high or low by identifying subsets of facts that contributed the most toward such observation. A different approach to query explanation is taken in [23]. The authors focus on multidimensional data where a binary dimension is present, and explain query results by building *explanation tables* which provide an interpretable and informative summary of the factors affecting the binary dimension. Finally, regression is used to explain query results in the XAXA approach [24]. The authors focus on aggregate

queries with a center-radius selection operator, and give explanations using a set of parametric piecewise-linear functions acquired through a statistical learning model.

The approach we propose is not competing with the ones mentioned above, but should rather be seen as a modular framework where any approach to explanation of aggregate data could be plugged. The added value lies in the IAM paradigm, i.e., in giving users the possibility of explicitly expressing intentions, in letting the system select the most interesting/suitable explanations, and showing these explanations together with data.

6. Discussion

In this paper we have given a proof-of-concept for explain intentions formulated inside the IAM framework. The explain syntax is flexible enough to suit users who wish to verify a specific hypothesis they made about an inter-measure relationship, as well as users who have no clue so they will let the system find the most interesting relationship.

The prototype we developed to test our approach relies on the MySQL DBMS to execute queries on a star schema based on multidimensional metadata. The algorithms used for regression analysis are imported from the Scikit-Learn Python library. Finally, the web-based visualization is implemented in JavaScript and exploits the D3 library for chart visualization.

To verify the feasibility of our approach from the computational point of view, we made some scalability tests. Two main factors affect performances: the cardinality of the cube to which a model is bound, $|C|$ (which determines the time required to compute a single model component), and the number of cube measures, $|M|$ (which determines the number of model components to be computed).

To evaluate scalability with reference to cube cardinality, we populated the SALES cube using the FoodMart data (https://github.com/julianhyde/foodmart-data-mysql) and considered 10 intentions with increasing cardinalities; in each intention we explained the revenue measure against both quantity and cost. The tests were run on an Intel(R) Core(TM)i7-6700 CPU@3.40GHz CPU with 8GB RAM; each intention was executed 10 times and the average results are considered. Remarkably, it turns out that less than one second is necessary to explain a cube of almost 87000 facts. Additionally, we measured the complexity (as the number of characters) of writing explain intentions vs. the underlying cube query. It turns out that our approach saves 85% of complexity with respect to writing cube queries (and without considering the complexity of extracting regression models, which would make our approach even more convenient).

To evaluate scalability with reference to the number of measures, we created a cube with $|C| = 10^6$ facts and $|M| = 10$ measures. As expected, our approach scales linearly in the number of measures, and given 9 measures and 10^6 facts, the computation of an explanation takes less than 7 seconds, thus fulfilling the requirement of near-real-time response typical of analytical workloads. The detailed results of the tests can be found in [25].

We close the paper by mentioning that the main direction for future research we wish to pursue is to generalize the definition of model to cope with additional model types.

References

[1] P. Vassiliadis, P. Marcel, S. Rizzi, Beyond roll-up's and drill-down's: An intentional analytics model to reinvent OLAP, Information Systems 85 (2019) 68–91.

[2] M. Francia, M. Golfarelli, P. Marcel, S. Rizzi, P. Vassiliadis, Assess queries for interactive analysis of data cubes, in: Proceedings of EDBT, Nicosia, Cyprus, 2021, pp. 121–132.

[3] M. Francia, P. Marcel, V. Peralta, S. Rizzi, Enhancing cubes with models to describe multidimensional data, Inf. Syst. Frontiers 24 (2022) 31–48.

[4] G. R. Mayes, Theories of Explanation, Internet Encyclopedia of Philosophy (2018).

[5] R. G. D. Steel, J. H. Torrie, Principles and procedures of statistics, with special reference to the biological sciences, McGraw-Hill, New York, 1960.

[6] P.-N. Tan, M. Steinbach, V. Kumar, Introduction to data mining, Pearson Education India, 2016.

[7] E. Ostertagová, Modelling using polynomial regression, Procedia Engineering 48 (2012) 500–506.

[8] L. D. Raedt, A perspective on inductive databases, SIGKDD Explorations 4 (2002) 69–77.

[9] A. Deshpande, S. Madden, MauveDB: supporting model-based user views in database systems, in: Proceedings of SIGMOD, Chicago, IL, USA, 2006, pp. 73–84.

[10] T. Kraska, Northstar: An interactive data science system, Proceedings of VLDB Endow. 11 (2018) 2150–2164.

[11] J. Han, OLAP mining: Integration of OLAP with data mining, in: Proceedings of Working Conf. on Database Semantics, Leysin, Switzerland, 1997, pp. 3–20.

[12] F. Bentayeb, C. Favre, RoK: Roll-up with the k-means clustering method for recommending OLAP queries, in: Proceedings of DEXA, Linz, Austria, 2009, pp. 501–515.

[13] S. Sarawagi, Explaining differences in multidimensional aggregates, in: Proceedings of VLDB, Edinburgh, Scotland, 1999, pp. 42–53.

[14] G. Sathe, S. Sarawagi, Intelligent rollups in multidimensional OLAP data, in: Proceedings of VLDB, Rome, Italy, 2001, pp. 531–540.

[15] B. Chen, L. Chen, Y. Lin, R. Ramakrishnan, Prediction cubes, in: Proceedings of VLDB, Trondheim, Norway, 2005, pp. 982–993.

[16] B. Glavic, A. Meliou, S. Roy, Trends in explanations: Understanding and debugging data-driven systems, Found. Trends Databases 11 (2021) 226–318.

[17] A. Meliou, W. Gatterbauer, J. Y. Halpern, C. Koch, K. F. Moore, D. Suciu, Causality in databases, IEEE Data Eng. Bull. 33 (2010) 59–67.

[18] A. Meliou, W. Gatterbauer, K. F. Moore, D. Suciu, The complexity of causality and responsibility for query answers and non-answers, Proceedings of VLDB Endow. 4 (2010) 34–45.

[19] S. Roy, D. Suciu, A formal approach to finding explanations for database queries, in: Proceedings of SIGMOD, Snowbird, UT, USA, 2014, pp. 1579–1590.

[20] E. Wu, S. Madden, Scorpion: Explaining away outliers in aggregate queries, Proceedings of VLDB Endow. 6 (2013) 553–564.

[21] Z. Miao, Q. Zeng, B. Glavic, S. Roy, Going beyond provenance: Explaining query answers with pattern-based counterbalances, in: Proceedings of SIGMOD, Amsterdam, The Netherlands, 2019, pp. 485–502.

[22] Z. Miao, A. Lee, S. Roy, LensXPlain: Visualizing and explaining contributing subsets for aggregate query answers, Proceedings of VLDB Endow. 12 (2019) 1898–1901.
[23] K. E. Gebaly, P. Agrawal, L. Golab, F. Korn, D. Srivastava, Interpretable and informative explanations of outcomes, Proceedings of VLDB Endow. 8 (2014) 61–72.
[24] F. Savva, C. Anagnostopoulos, P. Triantafillou, Explaining aggregates for exploratory analytics, in: Proceedings of BigData, Seattle, WA, USA, 2018, pp. 478–487.
[25] M. Francia, S. Rizzi, P. Marcel, The whys and wherefores of cubes, in: Proceedings of DOLAP, Ioannina, Greece, 2023, pp. 43–50.