The impact of rewriting on coverage constraint satisfaction

Chiara Accinelli, Barbara Catania, Giovanna Guerrini, Simone Minisi
DIBRIS - University of Genoa, Genoa - Italy
name.surname@dibris.unige.it

ABSTRACT

Due to the impact of analytical processes on our life, an increasing effort is being devoted to the design of technological solutions that help humans in measuring the bias introduced by such processes and understanding its causes. Existing solutions can refer to either back-end or front-end stages of the data processing pipeline and usually represent bias in terms of some given diversity or fairness constraint. In our previous work [1], we proposed an approach for rewriting filtering and merge operations in pre-processing pipelines into the "closest" operations so that protected groups are adequately represented (i.e., *covered*) in the result. This is relevant because any under-represented category in an initial or intermediate dataset might lead to an under-representation of that category in any subsequent analytical process. Since many potential rewritings exist, the proposed approach is approximate and relies on a sample-based cardinality estimation, thus introducing a trade-off between the accuracy and the efficiency of the process. In this paper, we investigate this trade-off by first presenting various measures quantifying the error introduced by the rewriting, due to the applied approximation and the selected sample. Then, we (preliminarly) experimentally evaluate such measures on a real-world dataset.

1 INTRODUCTION

The impact of data on our society is getting higher and higher, with data about people being more and more often exploited as the basis to make decisions that might impact people's lives. Thus, it becomes crucial to ensure that, in the systems enabling such data-based decisions, data are dealt with in a *responsible* and *non-discriminating* way [23], in all the steps, from acquisition to analysis.

Non-discrimination can be addressed by considering specific *diversity* and *fairness constraints*. Diversity allows us to capture the quality of a collection of items with respect to the variety of its constituent elements. On the other hand, fairness can be broadly defined as the impartial treatment of individuals and of demographic groups inside data processing tasks.

Among all data processing steps, data pre-processing plays a relevant role when considering non-discrimination issues since it can introduce *technical bias* by exacerbating pre-existing bias that may exist in society, with an impact on the whole data lifecycle.

When considering pre-processing tasks, an often considered non-discriminating constraint is *coverage*. Coverage constraints guarantee that the input, or training, dataset includes enough examples for each (protected) category of interest, thus increasing diversity with the aim of limiting the introduction of bias during the next analytical steps. Coverage is quite relevant in the first data processing tasks, like data transformation, since the used transformations might change the, possibly initially satisfied, coverage of protected categories.

In this paper, we are interested in investigating solutions for detecting bias in data-preprocessing steps, defined in terms of coverage constraints, with a special reference to filtering and merge data transformations, mitigating it, and checking whether the mitigation was effective. Specifically, we focus on classical data transformation operations, often defined in terms of Selection-Projection-Join (SPJ) operations over tabular data, that can reduce the number of records related to some protected or disadvantaged groups, defined in terms of some sensitive attributes, even if such attributes are not directly used in the specification of the data transformation operation.

In this frame, the approach we developed [1] aims at supporting the user by minimally rewriting the transformation operation so that input coverage constraints are guaranteed to be satisfied in the transformation result. Through rewriting, the revised process is traced for further processing, thus guaranteeing *transparency*. Since many potential rewritings exist, we proposed a sample-based two-steps approach for detecting, under an approximate approach, the minimal (i.e., the optimal) rewriting of the original query. After transforming the SPJ query into a canonical form, first (in a *pre-processing step*), the search space of potential rewritings is discretized, so that an approximation of the optimal solution can be detected in the next *processing step*, by looking at the resulting finite set of points. The coverage-based rewriting of the input query can be obtained by visiting the grid, produced as the result of the pre-processing step, according to an order that guarantees the fast detection of the minimal rewriting, and by verifying constraint satisfaction through a sample-based approach. The coverage-based rewriting is *approximate* both because of the discretization of the search space and of the error in estimating cardinalities and constraint satisfaction on the sample.

In this paper, we start from this approach and we investigate its effectiveness in the detection of the optimal coverage-based solution. To this aim, we introduce three main groups of measures quantifying the error that can be generated by the used approximated and sample-based approach. The first group of measures deals with the accuracy of the discretization applied in the pre-processing step. The second group deals with the error due to the usage of the sample for the grid generation and cardinality estimations, while the third group helps the user in quantitatively evaluating specific solutions obtained through the rewriting.

The generated rewritings are then (preliminarly) experimentally analyzed in terms of the proposed measures, on a real-world dataset. The obtained results provide hints on how to tune approximation and sample related parameters to achieve a good trade-off between accuracy and performance in the context of coverage-based rewriting, by combining new results related to accuracy presented in this paper with results related to performance, previously presented in [1].

Even if the proposed measures have been defined in the context of a specific coverage-based approach, we believe that they can be of more general value in understanding the role of approximation and sampling during data pre-processing.

The remainder of the paper is structured as follows. In Section 2, we present the overall approach to coverage-based rewriting. In Section 3, we introduce new measures to quantify, in terms of accuracy, the impact of rewriting. In Section 4, we experimentally analyze the impact of the rewriting according to the introduced measures. Section 5 discusses related work while Section 6 concludes the paper and outlines future work directions.

2 COVERAGE-BASED REWRITING

We focus on data to be transformed for further data analytical tasks. Data can refer specific protected (minorities or historically disadvantaged) groups and we aim at guaranteeing that each transformation step during the pre-processing pipeline, based on filtering or merge operations, produces a new dataset containing enough entries for each protected group of interest. In the following, we briefly describe input data and the proposed technique. Additional details can be found in [1, 3].

Datasets. Our rewriting approach can be applied over a collection of tabular datasets (e.g., relations in a relational database, Data Frames in the Pandas analytical environment) $I \equiv I_1, ..., I_r$. Among the attributes $A_1, ..., A_m$ of each input dataset, we assume that some discrete valued attributes $S_1, ..., S_n$ are of particular concern, since they allow the identification of protected groups, and we call them *sensitive attributes*. Examples of sensitive attributes are the `gender` (with values in $\{female, male\}$) and the `race` (with values in $\{asian, black, hispanic, white\}$).

Data pre-processing operations. The pre-processing operations we are interested in correspond to monotonic Select-Project-Join (SPJ) queries over input tabular data that might alter the representation (i.e., the coverage) of specific groups of interests, defined in terms of sensitive attribute values. To this aim, we focus on SPJ queries that return, among the others, at least one sensitive attribute (called *sensitive SPJ operations or queries*). For the sake of simplicity, we assume that selection conditions are defined over numeric attributes, even if the proposed approach can be easily extended to any other ordered domain. Thus, under the considered assumptions, sensitive attributes are not included in selection conditions (typical assumption in data processing).

In the following, when needed, we denote Q by $Q\langle v_1, ..., v_d\rangle$ or $Q\langle\overline{v}\rangle$, $\overline{v} \equiv (v_1, ..., v_d)$, where $v_1, ..., v_d$ are the constant values appearing in the selection conditions $sel_i \equiv A_i \theta_i v_i$ in Q.

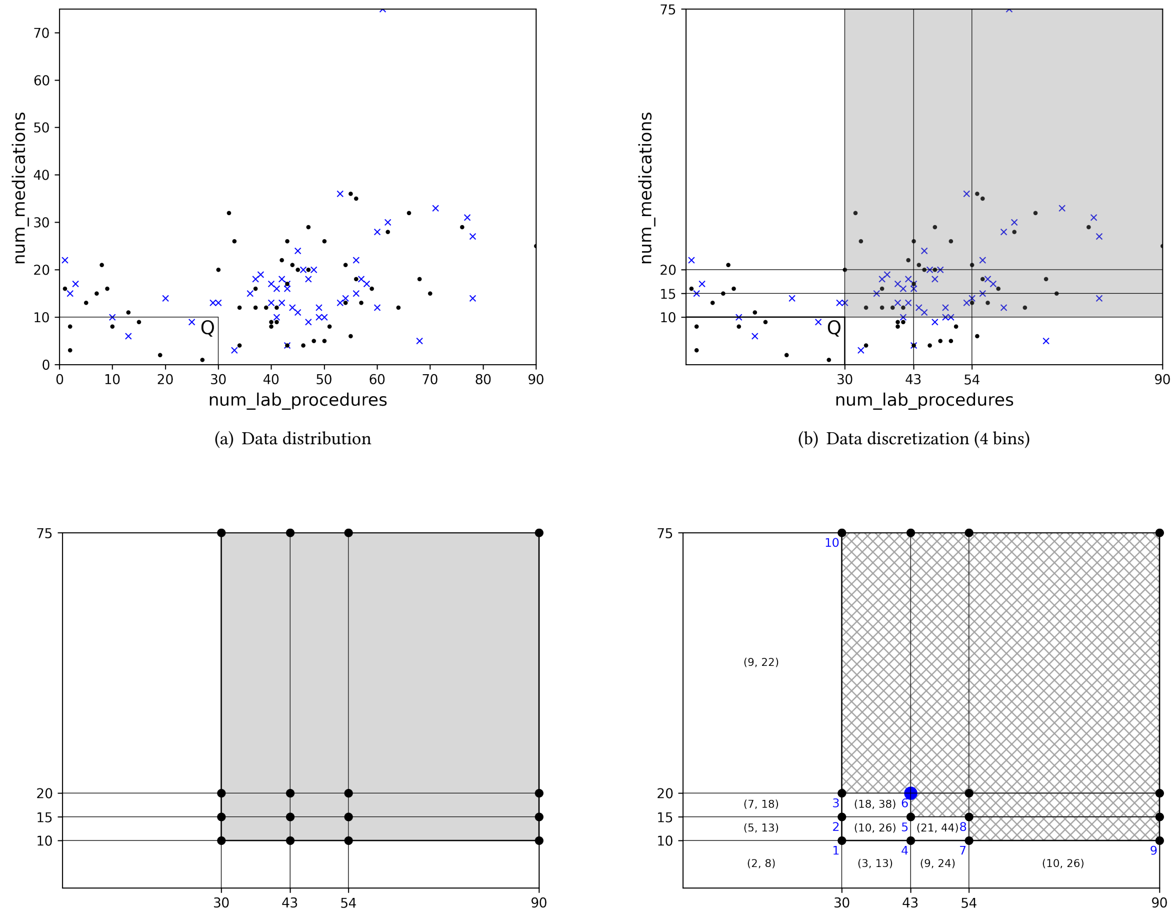
Coverage constraints. Conditions over the number of entries belonging to a given protected group of interest returned by the execution of SPJ queries can be specified in terms of *coverage constraints* [6, 27]. Given a sensitive SPJ query Q, with reference to a sensitive attribute S_i, and a value s_i belonging to the domain of S_i, $i \in \{1, ..., n\}$, a coverage constraint with respect to S_i and s_i is denoted by $Q \downarrow^{S_i}_{s_i} \geq k_i$ and it is satisfied by Q over the input dataset I when $card(\sigma_{S_i=s_i}(Q(I))) \geq k_i$ holds. For example, choosing `gender` as a sensitive attribute, a coverage constraint could be $Q \downarrow^{\texttt{gender}}_{female} \geq 10$, specifying that the result of Q must contain data related to at least 10 female individuals.

Coverage constraints can be provided together with Q or they can already be available in the system, as any other integrity constraint. This could be useful when they represent generally valid non-discrimination rules that must be satisfied by any query execution.

The approach. Given a dataset I and a set of coverage constraints CC, each selected sensitive SPJ query Q can be rewritten into another query $Q^{opt}_{I,CC}$, according to what presented in [1], so that $Q^{opt}_{I,CC}$ is the minimal query relaxing Q guaranteeing coverage constraint satisfaction when evaluated over the input dataset. Relaxation is reasonable when the user is happy with the specified transformation and she wants to keep the result set of the original query after the rewriting. $Q^{opt}_{I,CC}$ must therefore satisfy the following properties: (i) $Q^{opt}_{I,CC} \equiv Q\langle\overline{u}\rangle$, thus $Q^{opt}_{I,CC}$ is obtained from Q by only changing the selection constants; (ii) $Q \subseteq Q^{opt}_{I,CC}$, thus $Q^{opt}_{I,CC}$ always contains the result of the input query; (iii) all coverage constraints associated with Q are satisfied by $Q^{opt}_{I,CC}(I)$. The rewriting should be *optimal*, i.e., the new query has to satisfy specific *minimality* properties with respect to the input query Q. In order to make the definition of minimality properties homogeneous with respect to all the selection attributes A_i in Q, we define them in a transformed unit space, in which the values for each attribute A_i in I are normalized between 0 and 1. We denote with $\underline{Q}, \underline{I}, \underline{A_i}, \underline{\overline{v}}$ a query, a dataset, an attribute, and a vector of values, respectively, in the unit space. Notice that properties (i), (ii), and (iii) are satisfied in the original space if and only if they are satisfied in the normalized one. Minimality can now be stated according to the following two properties: (iv) there is no other query Q' satisfying conditions (i), (ii), and (iii) such that $Q'(I) \subset Q^{opt}_{I,CC}(I)$ (thus, $Q^{opt}_{I,CC}$ is the *minimal* query on I satisfying (i), (ii), and (iii)); (v) $\underline{Q}^{opt}_{\underline{I},CC} \equiv \underline{Q}\langle\underline{\overline{u}}\rangle$ is the closest query to $\underline{Q}\langle\underline{\overline{v}}\rangle$ according to the Euclidean distance between $\underline{\overline{v}}$ and $\underline{\overline{u}}$ in the unit space, satisfying (i), (ii), (iii), and (iv) (thus, $\underline{Q}^{opt}_{\underline{I},CC}$ is the coverage-based rewriting syntactically closest to the input query, thus maximizing *proximity* and potentially user satisfaction).

In order to compute the optimal coverage-based rewriting of an SPJ query $Q\langle\overline{v}\rangle$, given a set of coverage constraints CC and an instance I, we follow the approach presented in [1], consisting of three steps shortly discussed in what follows. For the sake of simplicity in the notations, in presenting the approach and the related examples, we do not underline symbols referring to the unit space, even if proximity is always considered in that space.

Canonical form generation. We first translate the selected SPJ queries into a *canonical form*, in which each selection condition containing operators ($>$, $\geq$, $=$) is translated into one or more equivalent conditions defined in terms of operator $<$. For example, any predicate of the form $A_i > v_i$ can be transformed into the predicate $-A_i < -v_i$. When considering canonical forms, an optimal coverage-based rewriting query is obtained from the input query by replacing one or more selection predicates $sel_i \equiv A_i < v_i$ with a predicate $sel'_i \equiv A_i < u_i$ with $u_i \geq v_i$. sel'_i is called a *relaxation* of sel_i. Relaxed queries generated through coverage-based rewriting starting from $Q\langle\overline{v}\rangle$, I, and CC have the form $Q\langle\overline{u}\rangle$, with $\overline{u} \geq \overline{v}$, and can be represented as points $\overline{u}$ in the d-dimensional space defined over the selection attributes. Taking into account the features of the canonical form and property (ii) of the optimal rewriting, it is simple to show that the query point corresponding to the optimal rewriting must be contained in the upper right region of the reference space with respect to the point represented by the input query.



(a) Data distribution

(b) Data discretization (4 bins)

(c) Multi-dimensional grid

(d) Multi-dimensional grid visit

Figure 1: Data representation and processing

Example 2.1. Consider the `Diabetes US` dataset[1] and let *gender* be the sensitive attribute. Suppose we are interested in finding people whose number of medications is less than 10 and the number of performed lab tests is less than 30. Additionally, suppose we would like to guarantee that at least 15 females are present in the query result (coverage constraint). The corresponding SPJ query is $Q\langle 30, 10\rangle$, defined in SQL as `SELECT * FROM Diabetes WHERE num_lab_procedures < 30 AND num_medications < 10`. Figure 1(a) shows the data distribution corresponding to a small sample of size 100 taken from the `Diabetes` relation, projected over the attributes referenced in the query selection conditions (points are colored and shaped according to the sensitive attribute values: blue crosses for females and black dots for males). The query corresponds to point $(30, 10)$ in such a space and the region boxed at the bottom left of point $(30, 10)$ contains the result of Q.

The query is already in canonical form, so no preliminary rewriting is required. The search space for detecting coverage-based rewritings of the input query corresponds to the grey region in Figure 1(b). ◇

Pre-processing. According to property (ii) of the coverage-based rewriting, given an input query $Q\langle\overline{v}\rangle$, any coverage-based rewriting is located in the upper right portion of the space defined by $\overline{v}$. Thus, the (unit) search space contains infinite possible coverage-based rewritings among which the optimal one should be identified. During the *pre-processing step*, such search space is discretized, so that an approximation of the optimal solution can be detected in the next *processing step* by looking at the resulting finite set of points. To this aim, we first organize the search space as a multi-dimensional grid. The grid has d axes, one for each selection attribute in the canonical form of $Q\langle\overline{v}\rangle$, and each axis, starting from query values, is discretized into a fixed set of bins, by using a binning approach (e.g, equi-width, dividing each axis in a fixed number of bins of equal size, or equi-depth, in which each bin contains an equal number of instances), typical of histogram generation. Each point $\overline{v}$ at the intersection of hyperplanes corresponding to bin values corresponds to a sensitive SPJ query containing $Q\langle\overline{v}\rangle$, thus satisfying condition (ii) of the reference problem. The set of grid points identified in this way is called *discretized search space*. The approach is *approximate* because a smaller query, in terms of minimality and proximity, than that identified by the algorithm, corresponding

[1] https://archive.ics.uci.edu/ml/datasets/Diabetes+130-US+hospitals+for+years+1999-2008

to a coverage-based rewriting of the input query, might exist but, if it lies inside one grid cell, it cannot be discovered by the algorithm. Notice that the grid is computed starting from I and Q (CC is not used).

Example 2.2. During the pre-processing step, by applying, as an example, the equi-depth binning approach to each axis of our reference example, we obtain the grid represented in Figure 1(b), considering 4 bins for each axis. The points of the resulting discretized search space correspond to the grid intersection points. Each point corresponds to a sensitive SPJ query, obtained from the input one by replacing selection constants with the grid point coordinates (see Figure 1(c)). ◇

Processing. During the *processing step*, we visit the discretized search space returned by the pre-processing step starting from the grid point corresponding to the input query. The discretized search space is visited one point after the other, at increasing distance from Q. For each point $\overline{u}$, we check whether the associated query $Q\langle\overline{u}\rangle$ is a coverage-based rewriting of $Q\langle\overline{v}\rangle$ by estimating the query result cardinality, for each protected group referenced by coverage constraints CC, and the query cardinality $card(Q(I))$.

The properties of the discretized search space and of the used canonical form are taken into account for pruning cells that cannot contain the solution and for further improving the efficiency of the process, by iteratively refining the size and the number of cells during the visit and, as a consequence, by working with a discretized search space at varying granularity [1].

Example 2.3. Figure 1(d) illustrates the processing approach for the considered example. Starting from the grid point corresponding to the input query Q, we estimate the cardinality of $Q \downarrow^{\text{gender}}_{female}$ on I, needed for checking constraint satisfaction, and of $Q(I)$, by relying on a sample-based approach, obtaining $(2, 8)$. Since the constraint is not satisfied, we further visit the other points of the discretized search space at increasing distance from Q, checking constraint satisfaction and looking for the minimum rewriting (property (iv)). The visit proceeds as pointed out in Figure 1(d) (the order of the visit is represented by the blue numbers associated with the top right vertex of each cell). Shaped cells are not visited thanks to the pruning effect: if $Q\langle\overline{u}\rangle(I)$ satisfies the coverage constraints, all the points in the upper right portion of space defined by $\overline{u}$ will not satisfy conditions (iv) and (v) of the reference problem, thus they can be discarded. The optimal coverage-based rewriting corresponds to query $Q\langle 43, 20\rangle$ (big blue dot in Figure 1 (d)) since $(43, 20)$ is the point at the minimum distance from the origin such that the corresponding query satisfies the considered coverage constraint ($Q \downarrow^{\text{gender}}_{female} \geq 15$). The input query is thus rewritten into `SELECT * FROM Adult WHERE num_lab_procedures < 43 AND num_medications < 20.` ◇

Sample-based estimation. The processing step, as well as coverage constraint checking, requires fast and accurate cardinality estimates. To make the processing more efficient, similarly to [16], we rely on sampling based estimators based on samples (uniform, independent, and without replacement) of the input dataset [9], dynamically constructed during the rewriting phase, and on the approach in [18] for generating the sample of joined tables. As well known, a sample of a given size can be generated so that query selectivity can be estimated with an error e and confidence δ [4]. As an example, if the error is 1% and the confidence is 95%, the sample size should be 9604: this means that 95 samples with size 9604 out of 100 will lead to an estimation error equal to 1%.

Performance evaluation. The analysis of the efficiency of the proposed coverage-based query rewriting approach has been investigated in a previous work [1]. The experiments demonstrated that the time complexity of the proposed algorithms depends on: the number of bins, the used binning approach, the number of selection conditions in the query, the coverage constraint threshold, and the sample size. Specifically, by varying the number of selection conditions or the number of bins, and therefore the dimensionality of the multi-dimensional grid and the size of the discretized search space, the execution time rapidly increases; the impact of the curve of dimensionality can however be reduced by applying the designed optimizations. Equi-depth binning approaches often lead to a more efficient processing than equi-width approaches since the distribution of points in the discretized search space follow data distribution, thus reducing the number of grid cells with no dataset points and, as a consequence, the number of cardinality estimations. The execution time also depends on the chosen coverage constraint thresholds and the number of coverage constraints. Finally, the sample size influences the cardinality estimation time and therefore the total execution time.

3 IMPACT EVALUATION

The coverage-based rewriting of the input query can be obtained by visiting the grid, produced as the result of the pre-processing step, and by verifying constraint satisfaction relying on a sample-based approach. The optimal coverage-based rewriting is therefore approximate since: (i) the grid, that corresponds to the discretized search space, might have an impact on the accuracy of the selected coverage-based rewriting; (ii) the estimation error related to the sample usage has an impact on query cardinality estimation and on constraint satisfaction.

It is therefore important to introduce some measures quantifying the error that can be generated. To this aim, in the following we discuss three groups of measures: the first group deals with the approximation error related to the usage of the grid for the discretization of the query search space; the second group deals with the approximation error related to the usage of a sample during the pre-processing and processing phases; the third concerns the error related to the detected optimal rewriting.

3.1 Grid-based accuracy

Due to the usage of a discretized search space, the optimal coverage-based rewriting identified by the proposed approach is the best approximation of an optimal rewriting, given the considered grid. The accuracy related to the usage of a given grid for the detection of the optimal rewriting thus corresponds to the error introduced by the discretization process.

As pointed out before, given a query $Q\langle\overline{v}\rangle$, the visit proceeds at increasing distance from the query point $\overline{v}$. When an optimal rewriting $Q\langle\overline{u}\rangle$ is reached, this means that the neighbours of $\overline{u}$ in the discretized search space cannot be optimal rewritings (otherwise the search would have stopped before). On the other hand, another point $\overline{z}$ might exist that is not included in the search space but $Q\langle\overline{z}\rangle(I)$ satisfies CC and either it is closer to the query point $\overline{v}$ than $\overline{u}$ or $Q\langle\overline{z}\rangle(I) \subseteq Q\langle\overline{u}\rangle(I)$. Such point is an optimal rewriting but, due to the approximation, it cannot be identified by the proposed approach. The approximation error

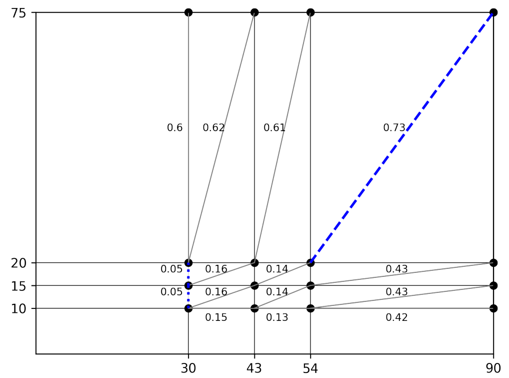


Figure 2: Grid-based accuracy (dashed blue line for the maximum diagonal, dotted blue line for the minimum diagonal)

for the identified approximate optimal rewriting $Q\langle\overline{u}\rangle$, also called *grid-based accuracy*, can therefore be defined as the maximum distance between $\overline{u}$ and all its neighbours on the grid, preceding it in the search; the accuracy is therefore lower than or equal to the diagonal of the grid cells having $\overline{u}$ as a vertex and closer to $\overline{v}$ than $\overline{u}$. By considering the entire grid, we can quantify the maximum and minimum grid-based accuracy in terms of the maximum and the minimum diagonal length of grid cells (see Figure 2 for a graphical explanation).

Definition 3.1 (Grid-based accuracy). Let G^b be the grid generated from a dataset I and a query Q using a certain binning approach b. The mimimum/maximum grid-based accuracy of G^b, denoted by $diag^{min}_{G^b}$ and $diag^{max}_{G^b}$, is defined as the minimum/ maximum diagonal length of grid cells in G^b, normalized between 0 and 1. □

Example 3.2. Figure 2 shows the normalized cell diagonal lengths for the grid created as discussed in Example 2.2. The grid-based accuracy for this grid varies between 0.05 (dotted blue line), corresponding to queries $Q\langle 30, 15\rangle$ and $Q\langle 30, 20\rangle$ and 0.73 (dashed blue line), corresponding to query $Q\langle 90, 75\rangle$. ◇

Different binning approaches might lead to a different grid-based accuracy. In particular, when fixing the reference data interval over each axis, binning approaches based on data distribution, like equi-depth, lead to a higher variability of grid-based accuracy, since they generate smaller buckets for dense regions and larger buckets for sparse ones, as stated by the following proposition.

Proposition 1. *Let G^w be the grid generated from a dataset I and a query Q using the equi-width approach and G^d that generated using the equi-depth approach, with n bins for each axis in both cases. Then: (i) $diag^{min}_{G^d} \leq diag^{min}_{G^w}$; (ii) $diag^{max}_{G^d} \geq diag^{max}_{G^w}$.* □

3.2 Sample-based accuracy

The accuracy of the query rewriting approach depends on the sample data distribution for two main reasons: (i) different sample distributions might lead to the generation of different buckets and therefore of different grids, with an impact on both preprocessing and processing steps; (ii) the sample is used for query cardinality estimation and, as a consequence, the estimation error has an impact on the minimality property, in addition, it might lead to a wrong assessment of constraint satisfaction.

Impact on (pre-)processing In order to evaluate the impact of the sample usage on grid generation, we compare the data distribution of the dataset I with the data distribution of the sample S, both projected over the attributes appearing in the considered query Q. The more similar the two distributions, the lower is the impact of the sample selection in the detection of the optimal coverage-based rewriting.

Several metrics have been proposed for quantifying the distance between two multivariate datasets. Many of them, e.g., the Wasserstein metric [17], the Kullback-Leibler [13, 14] and the Jensen-Shannon divergence [10], quantify the distance between the corresponding probability distributions and sometimes, as for the Wasserstein metric, the result might tend to infinity [17]. Probability distributions are not directly available under the considered scenarios and, even if computed, they introduce a further level of approximation in the computation. The Kolmogorov-Smirnov (KS) distance, defined for arbitrary univariate distributions [7, 22], measures the maximum distance, between 0 and 1, between the cumulative distributions of two datasets. Due to its generality, it is quite used and many, very complex, extensions to the multivariate case exist.

In our work, we consider a very simple extension of the univariate KS metric, obtained by averaging the KS distance computed for each query attribute. The distance computed between two datasets I and S is denoted by $d^{KS}(I, S)$. This approach is suitable since, similarly to what we have proposed for the search space construction, where we rely on an aggregation of unidimensional histograms instead of a multidimensional histogram, it aggregates distances defined on each single attribute.

In order to investigate the impact of the KS distance for a sample S on the identification of the optimal rewriting, it is useful to introduce some metrics, quantifying the difference in using S instead of the initial dataset I for the detection of the optimal coverage-based rewriting. Such metrics compute the average difference, in terms of *minimality*, *proximity*, and *solution distance*, between the optimal rewritings $Q^{opt}_{S,CC}$ and $Q^{opt}_{I,CC}$, obtained by processing the sample S and the original dataset I, respectively, on a random set of queries QS, uniformly distributed in the query search space:

- *Average minimality difference* (property (iv) of the optimal rewriting). It quantifies how much different $Q^{opt}_{S,CC}$ and $Q^{opt}_{I,CC}$ are, in average, with respect to their result set cardinalities when they are executed over the input dataset I (thus, it quantifies, in average, the difference in the relaxation of the original query over the initial dataset): $m(I, S) \equiv avg_{Q\in QS}\frac{|card(Q^{opt}_{S,CC}(I))-card(Q^{opt}_{I,CC}(I))|}{card(Q(I))}$.
- *Average proximity difference* (property (v) of the optimal rewriting). It quantifies how much different $Q^{opt}_{S,CC}$ and $Q^{opt}_{I,CC}$ are, in average, with respect to their Euclidean distance $d()$ from the input query Q, in the unit space, further normalized between 0 and 1 (thus, it quantifies, in average, how far the two optimal rewritings are with respect to the original query): $p(I, S) \equiv avg_{Q\in QS}|d(Q^{opt}_{S,CC}, Q) - d(Q^{opt}_{I,CC}, Q)|$.
- *Average solution distance*: It quantifies how much different $Q^{opt}_{S,CC}$ and $Q^{opt}_{I,CC}$ are, in average, in terms of their

Euclidean distance in the unit space, further normalized between 0 and 1 (thus, it quantifies, in average, how far the two optimal rewritings are, without taking the input dataset and the original query into account): $sd(I,S) \equiv avg_{Q \in QS} d(Q^{opt}_{S,CC}, Q^{opt}_{I,CC})$.

Impact on constraint satisfaction. The error and the confidence related to the considered sample (see Section 2) have an impact also on coverage constraint satisfaction. A constraint $CC_i \equiv Q \downarrow^{S_i}_{s_i} \geq k_i$ is satisfied on I if $card(\sigma_{S_i=s_i}(Q(I))) \geq k_i$. Since the sample-based estimation of $card(\sigma_{S_i=s_i}(Q(I)))$ might lead to an error of $e \times card(I)$, in order to guarantee that the constraint is also satisfied by the input dataset I, we can modify the constraint, when evaluated over the sample, as: $Q \downarrow^{S_i}_{s_i} \geq k_i + (e \times card(I))$. This consideration makes the proposed sample-based approach reasonable for coverage constraints in which $\frac{k_i}{card(I)}$ has the same order of magnitude as e. Notice that, by changing the constraint as proposed above, we increase the probability of constraint satisfaction on the input dataset at the price of reducing proximity of the optimal rewriting with respect to the input query (since the optimal query will be further away from the initial one).

3.3 Solution-based accuracy

Let S be a sample of dataset I and $Q^{opt}_{S,CC} = Q\langle\overline{u}\rangle$ be the optimal solution obtained from S, given a query $Q\langle\overline{v}\rangle$ and a set of coverage constraints CC. It could be useful to introduce some additional measures to evaluate the quality of the obtained optimal rewriting $Q^{opt}_{S,CC}$ with respect to the discretized search space identified by the chosen dataset S, taking into account both the applied relaxation with respect to the original query and the approximation error, in line with what we have proposed for grid-based and sample-based accuracy. To this aim, we propose the following three measures:

- *Grid-based accuracy of* $Q^{opt}_{S,CC}$. According to what discussed in Subsection 3.1, it can be computed as the maximum distance between $\overline{u}$ and all its neighbours on the grid, preceding it in the search (thus, the maximum diagonal of the cells of the grid having $\overline{u}$ as a vertex and closer to $\overline{v}$ than $\overline{u}$).
- *Relaxation degree.* Similarly to [16], the relaxation degree, first proposed in [1], quantifies, through estimations over the initial dataset I, how much the optimal coverage-based rewriting $Q^{opt}_{S,CC}$ relaxes the original query Q, as the percentage of new added tuples with respect to those contained in the original query result: $\frac{|Q^{opt}_{S,CC}(I)|-|Q(I)|}{|Q(I)|}$.
- *Proximity.* It can be computed as the Euclidean distance between the optimal coverage-based rewriting $Q^{opt}_{S,CC}$ and Q in the unit space, further normalized between 0 and 1, thus indicating how far the optimal rewriting is with respect to the original query.

Example 3.3. The optimal coverage-based rewriting of the running example corresponds to query $Q\langle 43, 20\rangle$ (see Example 2.3). The *grid-based accuracy* of such solution is 0.16 (see Figure 2), since this is the maximum length between the solution and its neighbours on the grid. Since $card(Q\langle 43, 20\rangle(I)) = 38$ and $card(Q(I)) = 8$ (see Figure 2), the *relaxation degree* is 3.75 (to guarantee constraint satisfaction, the cardinality of the rewritten query is about 4 times that of the original query). Finally, the *proximity*, i.e., the Euclidean distance between $(30, 10)$ and $(43, 20)$ in the unit space, normalized between 0 and 1, is 0.19, thus corresponding to a relaxation of 19% with respect to the maximum query, returning the whole dataset. ◇

4 EXPERIMENTAL RESULTS

In this section, we present some preliminary experimental results with the aim of analyzing the accuracy of the proposed query rewriting approach.

4.1 Experimental Setup

All experiments were conducted on a PC with an Intel Core i5-8300H 2.30 GHz CPU and 16GB main memory, running Microsoft Windows 10. All programs were coded in Python 3.8.

The experiments refer to a real dataset, stored in PostgreSQL: `Diabetes US`[2] representing 10 years (1999-2008) of clinical care at 130 US hospitals and integrated delivery networks (100,000 instances). It includes over 50 features representing patient and hospital outcomes. For our experiments we use `gender` as sensitive attribute and we add a coverage constraint on $female$.

For this dataset, we generated many random samples with an increasing size, guaranteeing a variable percentage of error (1% or 3%) and a variable level of confidence (95% or 99%). More precisely, we considered the following sample sizes: 1067 (3% error and 95% confidence), 1843 (3% error and 99% confidence), 9604 (1% error and 95% confidence), 16588 (1% error and 99% confidence). For each sample size, we generated 5 random samples, for a total of 20 samples. The samples are all small enough to be stored in main memory.

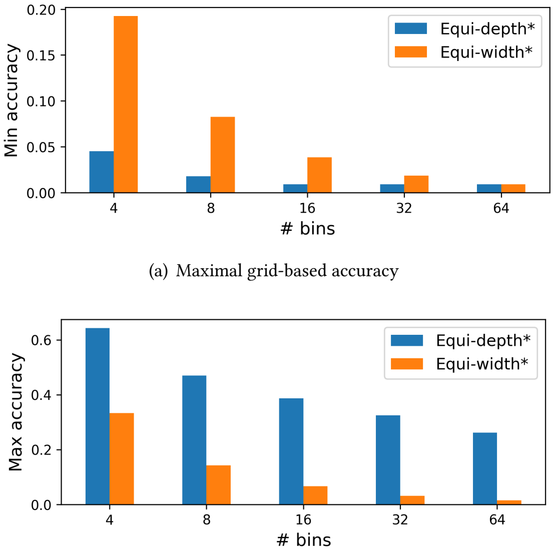
In order to evaluate the accuracy of the proposed approach, we randomly generated a set of 1000 queries, with different selectivities, and we compared the impact of the grid and of the considered sample in detecting the optimal coverage-based rewriting, by considering the measures presented in Section 3. All the selected queries contain three selection conditions (thus leading to a three-dimensional grid), with respect to the three numerical attributes with the highest number of distinct values, namely (`number_emergency,num_lab_procedures,num_medications`); selection values are picked at random from the attribute value range in the dataset. For the sake of simplicity, join queries are not considered for these preliminary experiments.

For each query, we then defined the coverage constraint taking into account the considered query and in such a way that it is neither satisfied on the dataset nor on the considered samples.

The experiments were performed by considering two distinct binning approaches during the pre-processing step: (i) *equi-width**, corresponding to an equi-width approach, dividing each axis in a fixed number of bins of equal size, set as the minimum between a selected number (denoted by *#bins* in the experimental results) and the number of distinct values for the considered attribute in the dataset; (ii) *equi-depth**, in which each bin, defined as for the equi-width* approach, contains a constant number of instances. A variable number of bins, namely 4, 8, 16, 32, 64, has been considered for specific experiments.

We then performed three groups of experiments aimed at analyzing the grid-based, the sample-based, and the solution-based accuracy of the optimal coverage-based rewriting.

[2] https://archive.ics.uci.edu/ml/datasets/Diabetes+130-US+hospitals+for+years+1999-2008



(a) Maximal grid-based accuracy

(b) Minimal grid-based accuracy

Figure 3: Grid-based accuracy, when varying the number of bins

Sample	KS Distance
1067_1	0.0185
1067_2	0.0185
1067_3	0.0185
1067_4	0.0199
1067_5	0.0198
Mean	0.0190
Variance	4.38e-7

Sample	KS Distance
1843_1	0.0156
1843_2	0.0183
1843_3	0.0130
1843_4	0.0130
1843_5	0.0119
Mean	0.0144
Variance	5.36e-6

Sample	KS Distance
9604_1	0.0058
9604_2	0.0064
9604_3	0.0026
9604_4	0.0071
9604_5	0.0134
Mean	0.0071
Variance	1.24e-5

Sample	KS Distance
16588_1	0.0034
16588_2	0.0044
16588_3	0.0032
16588_4	0.0055
16588_5	0.0035
Mean	0.0040
Variance	7.32e-7

Figure 4: KS distance from the samples to the reference dataset

Sample	KS distance	avg min. difference	avg prox. difference	avg sol. distance
9604_3	0.0026	0.0275	0.1072	0.1403
9604_5	0.0134	0.1350	0.0995	0.1291

Table 1: Sample-based accuracy measures

4.2 Experimental evaluation

4.2.1 Grid-based accuracy. The first group of experiments aims at analyzing the maximum and minimum grid-based accuracy, as presented in Subsection 3.1, by varying the number of bins, with respect to the selected binning approach, namely equi-depth* and equi-width*. To this aim, we selected a sample with size 16588 (sample 16588_3 in Figure 4) and we considered all the generated 1000 random queries. The obtained results are independent from the query selectivity, therefore in the following we discuss those obtained with selectivity equal to 2.5%.

Figure 3 shows that both the maximum and the minimum grid-based accuracy decrease by increasing the number of bins, independently from the chosen binning approach. This is because, by increasing the number of bins, the space is discretized into a higher number of cells, thus obtaining cells of smaller size.

From Figure 3 we also observe the behaviour described by Proposition 1: the maximal grid-based accuracy obtained with the *equi-depth** approach is always higher than the grid-based accuracy obtained with the *equi-width** approach; minimal accuracy behaves in the opposite way.

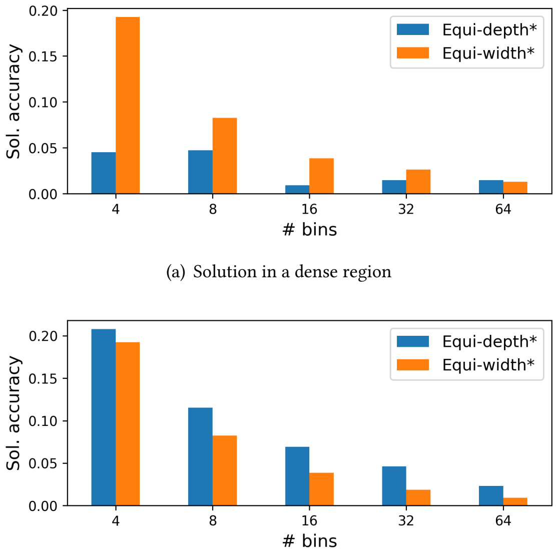
4.2.2 Sample-based accuracy. The second group of experiments deals with the analysis of the sample-based accuracy, according to the measures introduced in Subsection 3.2. To this aim, we considered all the samples described in Subsection 4.1 and, for each of them, we computed the KS distance with respect to the input dataset.

Results, presented in Figure 4, show that the obtained values are very similar and quite small: for most sample sizes, the greatest differences refer to the third decimal digit. As expected, by increasing the sample size, the KS distance tends to decrease.

In order to evaluate the impact of the KS distance on sample-based accuracy measures, we considered the sample size leading to the highest variance (9604) and we selected the samples with the highest difference between the corresponding KS distances (namely, 9604_3 and 9604_5). We then computed the sample-based accuracy measures on the set of 1000 random queries.

Table 1 shows the results obtained by considering the equi-depth* approach and 32 bins (similar values has been obtained for the equi-width* and other numbers of bins). As you can see, there is no clear ordering between the two samples when considering such measures. In particular, sample 9604_3 is better than sample 9604_5 with respect to minimality; however, for proximity and solution distance the opposite result holds. Thus, it seems that the KS measure is not good enough for discriminating between samples with a different behaviour with respect to the detection of the optimal coverage-based rewriting. Additional work is therefore needed for investigating or defining alternative functions able to discriminate between samples under the considered scenario.

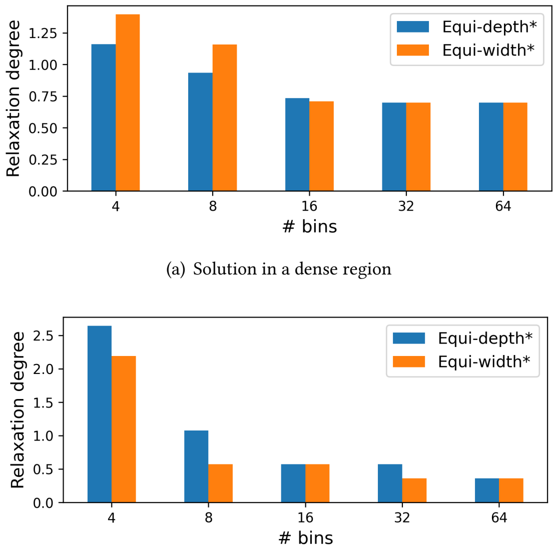
4.2.3 Solution-based accuracy. The last group of experiments aims at analyzing the solution-based accuracy, according to the measures introduced in Subsection 3.3, by varying the number of bins, with respect to the selected binning approach, namely equi-depth* and equi-width*. To this aim, we selected sample 16588_3 (i.e., the biggest sample with the smallest KS distance with respect to the initial dataset) and we considered all the generated 1000 random queries. The obtained results show a different behaviour with respect to the region in which the optimal solution is located, either dense or sparse. In particular, Figure 5 shows that, as expected, by increasing the number of bins, the grid-based accuracy of the solution will decrease as well. However, depending on where the optimal solution is located, the equi-depth* and the equi-width* approaches behave in a different way. More precisely, when the solution region is dense (Figure 5(a)), the accuracy is lower with the equi-depth* approach since in this case higher density will lead to smaller bins. On the other hand, when the solution region is sparse (Figure 5(b)), the accuracy is usually



(a) Solution in a dense region

(b) Solution in a sparse region

Figure 5: Grid-based accuracy of the solution



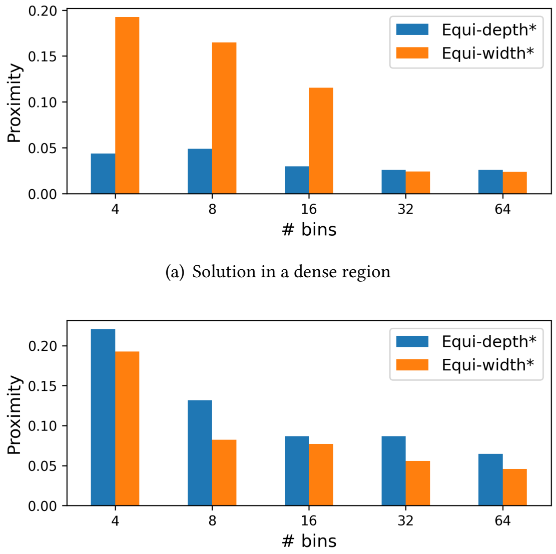
(a) Solution in a dense region

(b) Solution in a sparse region

Figure 6: Relaxation degree

lower with the equi-width* approach since in this case lower density will lead to longer bins under the equi-depth* approach.

A similar behavior can be observed for the relaxation degree (Figure 6) and the proximity (Figure 7): by increasing the number of bins, they decrease for both the binning approaches. When the solution is contained in a dense region, equi-depth* behaves better than equi-width*, especially for low numbers of bins. We further notice that, for very low numbers of bins, proximity and grid-based accuracy coincide (the farthest neighbour of the solution is the query point itself).



(a) Solution in a dense region

(b) Solution in a sparse region

Figure 7: Proximity

Finally, notice that dense regions have a greater impact on query cardinalities and, as a consequence, on the relaxation degree, whose values are usually higher in this case (starting from 8 bins), independently on the selected binning approach. On the other hand, dense regions tend to generate optimal solutions with lower proximity (more evident with the equi-depth* approach, that is more sensible to data distribution), since constraint satisfaction can be obtained on query points closer to the initial one. In general, for uniformly distributed datasets, in which no sparse regions can be detected, equi-depth* can be considered the best option. By combining the obtained results with those presented in [1, 3], related to performance, a number of bins equal to 16 can be considered a good compromise between effectiveness and efficiency.

5 RELATED WORK

The interest for coverage constraints has been introduced in [5, 6], drawing inspiration from the literature on diversity [11]. The problem of evaluating the coverage of a given dataset has been considered in the context of the MithraLabel system [12, 24], in which the lack of coverage is modeled as a widget in the nutritional label [26] of a dataset. Once the lack of coverage has been identified, the smallest number of data points needed to hit all the "large uncovered spaces" is identified with the aim of helping data owners in achieving coverage through a data repairing approach. When protected categories are defined in terms of many attributes, the identification of attribute patterns associated with coverage problems might lead to performance issues, due to the combinatorial explosion of such patterns. Efficient techniques, inspired from set enumeration and association rule mining, addressing this problem have been proposed in [6]. To fix coverage unsatisfaction, additional data can be acquired. Since data acquisition has a cost in term of data processing, techniques have been presented in [6] for determining the patterns that can be covered given a certain maximum cost. An efficient approach for coverage analysis, given a set of attributes across multiple tables, is presented in [27]. As pointed out by the previous discussion, most existing approaches chase coverage through data repair and focus on efficiency issues. By contrast, we consider accuracy for coverage-based query rewriting during data transformation, thus complementing existing approaches.

The technique considered in this paper relies on rewriting. Other fairness-aware rewriting approaches have been proposed for OLAP queries [19, 20]. Bias is defined in terms of causal fairness (checking for causal relationships from the sensitive attributes to the outcome) and detected, explained, and resolved through rewriting. On the other hand, we focus on data transformations in presence of coverage constraints.

Impact evaluation is quite relevant in the design and the execution of non-discriminating pipelines, usually very complex in real-world scenarios. Various systems have been designed for supporting the user during this activity. Among them, we recall: Fair-DAGs [25], an open-source library aiming at representing data processing pipelines in terms of a directed acyclic graph (DAG) and identifying distortions with respect to protected groups as the data flows through the pipeline; FairPrep [21], an environment for investigating the impact of fairness-enhancing interventions inside data processing pipelines; AI Fairness 360 [8], an open-source Python toolkit for algorithmic fairness, aimed at facilitating the transition of fairness-aware research algorithms to usage in an industrial setting and at providing a common framework to share and evaluate algorithms.

6 CONCLUDING REMARKS

Rewriting approaches have been recognized as an interesting mean for enforcing specific non-discriminating properties guaranteeing transparency. In this paper, we started from the approach proposed in [1] with the aim of investigating the impact of rewriting on coverage-constraint satisfaction. The approach is approximate and relies on a sample for both the construction of the solution search space and the detection of the optimal rewriting. Three different groups of measures have been proposed for quantifying the accuracy induced by the approximation and the impact of the sample in the detection of the optimal solution.

Preliminary experimental results show that: (i) different binning approaches lead to different grid-based accuracy degrees; (ii) common measures for computing the distance between distributions are not effective for analyzing their behaviour in the detection of the optimal coverage-based solution; (iii) the number of bins used in the generation of the search space has an impact in the accuracy of the detected solution, and not only on the performance [1], while the optimal binning approach depends on the position of the optimal rewriting in the search space; (iv) a number of bins greater than 16 represents a good compromise between accuracy and efficiency.

The proposed approach is at the basis of covRew [2], a Python toolkit for rewriting slicing operations in pre-processing pipelines, ensuring coverage constraint satisfaction. covRew takes in input a two-dimensional tabular dataset I, with the related sensitive attribute specification, for the identification of protected groups, a processing pipeline represented as a Pandas script [15], and a set of coverage constraints. It includes three main components: (i) a pipeline analyzer, which identifies candidate operations for rewriting, (ii) a pipeline rewriter, which transforms operations that are selected by the user according to the input coverage constraints, and (iii) an impact evaluator, assessing the impact of the rewriting through the usage of the grid-based and solution-based measures presented in Section 3. Such measures could lead the user to reconsider some of the choices made in the selection of the operations to be rewritten, thus producing a new annotated script.

Future work is needed in order to understand how to define new distance functions between distributions, focusing on their behaviour during coverage-based rewriting. An additional issue concerns the definition and the analysis of further solution-based measures, evaluating the quality of the detected solution with respect to the solution that would have been obtained without considering a sample, as well as their integration in the covRew prototype system.

REFERENCES

[1] C. Accinelli, S. Minisi, B. Catania. Coverage-based rewriting for data preparation. In *Proc. of the EDBT/ICDT Workshops*, CEUR-WS.org, 2020.

[2] C. Accinelli, B. Catania, G. Guerrini, S. Minisi. covRew: a Python toolkit for pre-processing pipeline rewriting ensuring coverage constraint satisfaction. In *EDBT, Proc. of the 24th Int. Conf. on Extending Database Technology*, 2021.

[3] C. Accinelli, B. Catania, G. Guerrini, S. Minisi. A coverage-based approach to data transformations. *In preparation.*

[4] S. Agarwal et al. Knowing when you're wrong: building fast and reliable approximate query processing systems. In *Proc. of the ACM SIGMOD, Int. Conf. on Management of Data*, pages 481–492, 2014.

[5] A. Asudeh, H. V. Jagadish, J. Stoyanovich. Towards responsible data-driven decision making in score-based systems. *IEEE Data Eng. Bull.*, 42(3):76–87, 2019.

[6] A. Asudeh, Z. Jin, H. V. Jagadish. Assessing and remedying coverage for a given dataset. In *ICDE, Proc. of the 35th IEEE Int. Conf. on Data Engineering*, pages 554–565, 2019.

[7] M. Basseville. Divergence measures for statistical data processing. 2010.

[8] R. K. Bellamy et al. Ai fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. *IBM Journal of Research and Development*, 63(4/5):4–1, 2019.

[9] G. Cormode, M. N. Garofalakis, P. J. Haas, C. Jermaine. Synopses for massive data: Samples, histograms, wavelets, sketches. *Found. Trends Databases*, 4(1-3):1–294, 2012.

[10] I. Dagan, L. Lee, F. Pereira. Similarity-based methods for word sense disambiguation. *arXiv preprint cmp-lg/9708010*, 1997.

[11] M. Drosou, H. V. Jagadish, E. Pitoura, J. Stoyanovich. Diversity in big data: A review. *Big Data*, 5(2):73–84, 2017.

[12] Z. Jin et al. Mithracoverage: A system for investigating population bias for intersectional fairness. In *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, pages 2721–2724, 2020.

[13] S. Kullback. *Information theory and statistics.* Courier Corporation, 1997.

[14] S. Kullback, R. A. Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

[15] W. McKinney. Python for data analysis: Data wrangling with Pandas, NumPy, and IPython. O'Reilly Media, Inc., 2012.

[16] C. Mishra, N. Koudas. Interactive query refinement. In *EDBT, Proc. of the 12th Int. Conf. on Extending Database Technology, Proceedings*, pages 862–873, 2009.

[17] I. Olkin, F. Pukelsheim. The distance between two random vectors with given dispersion matrices. *Linear Algebra and its Applications*, 48:257–263, 1982.

[18] M. Riondato et al. The VC-dimension of SQL queries and selectivity estimation through sampling. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD, Proceedings, Part II*, pages 661–676, 2011.

[19] B. Salimi et al. Hypdb: A demonstration of detecting, explaining and resolving bias in OLAP queries. *Proc. VLDB Endow.*, 11(12):2062–2065, 2018.

[20] B. Salimi, J. Gehrke, D. Suciu. Bias in OLAP queries: Detection, explanation, and removal. In *Proc. of the ACM SIGMOD Int. Conf. on Management of Data*, pages 1021–1035, 2018.

[21] S. Schelter, Y. He, J. Khilnani, J. Stoyanovich. FairPrep: Promoting data to a first-class citizen in studies on fairness-enhancing interventions. In *EDBT, Proc. of the 23nd Int. Conf. on Extending Database Technology*, pages 395–398, 2020.

[22] M. A. Stephens. Edf statistics for goodness of fit and some comparisons. *Journal of the American statistical Association*, 69(347):730–737, 1974.

[23] J. Stoyanovich, B. Howe, H. V. Jagadish. Responsible Data Management. *Proc. VLDB Endow.*, 13(12): 3474–3488, 2020.

[24] C. Sun et al. Mithralabel: Flexible dataset nutritional labels for responsible data science. In *CIKM, Proc. of the 28th Int. Conf. on Information and Knowledge Management*, pages 2893–2896, 2019.

[25] K. Yang, B. Huang, J. Stoyanovich, S. Schelter. Fairness-aware instrumentation of preprocessing pipelines for machine learning. In *Workshop on Human-In-the-Loop Data Analytics (HILDA 2020)*, 2020.

[26] K. Yang et al. A nutritional label for rankings. In *Proc. of the 2018 ACM SIGMOD Int. Conf. on Management of Data*, pages 1773–1776, 2018.

[27] Y. Lin, Y. Guan, A. Asudeh, H. V. Jagadish. Identifying insufficient data coverage in databases with multiple relations. *Proc. of the VLDB Endow*, pages 2229–2242, 2020.