Approximately Measuring Functional Dependencies: a Comparative Study

Marcel Parciak
Hasselt University, Belgium

Sebastiaan Weytjens
Hasselt University, Belgium

Niel Hens
Hasselt University, Belgium

Frank Neven
Hasselt University, Belgium

Liesbet Peeters
Hasselt University, Belgium

Stijn Vansummeren
Hasselt University, Belgium

Abstract

Functional dependency (FD) discovery is a fundamental step in data profiling. While current discovery algorithms focus on the discovery of FDs that hold exactly, in practice, however, data often contain FD violations. Discovery should therefore be broadened to approximate FDs (AFDs), that is, dependencies that almost hold in a relation. Even though various measures have been proposed to quantify the level to which an FD holds approximately, these measures are difficult to compare, let alone decide which one to use, as they measure vastly different quantities. This paper aims to formally and qualitatively compare the measures that have been proposed in the literature. We obtain a formal comparison through a novel presentation of these measures in terms of Shannon and logical entropy. Furthermore, we study their effectiveness for discovering AFDs on real world as well as synthetic data. We find that overall the little-known measure μ is the most effective, closely followed by the widely-known measure g_3 but only when correctly normalized. An additional advantage of μ over g_3 is that it is not susceptible to right-hand side data skew. On real-world data, the Shannon entropy-based measure *FI* is the least effective and known corrections of *FI* fail to reach the level of most logical entropy-based measures.

PVLDB Reference Format:
Marcel Parciak, Sebastiaan Weytjens, Niel Hens, Frank Neven, Liesbet Peeters, and Stijn Vansummeren. Approximately Measuring Functional Dependencies: a Comparative Study. PVLDB, 15(1): XXX-XXX, 2022.
doi:XX.XX/XXX.XX

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

1 Introduction

Traditionally, functional dependencies (FDs) are defined during the database design phase to restrict the set of admissible databases: databases where the FDs do not hold are considered incorrect and those are therefore forbidden. As such, FDs are an important tool for ensuring data consistency and aiding in data cleaning [13, 40]. In addition, FDs also play a prominent role in data integration [45] and query optimization [26, 28], among other tasks.

Proceedings of the VLDB Endowment, Vol. 15, No. 1 ISSN 2150-8097.
doi:XX.XX/XXX.XX

While there are many benefits to knowing the set of FDs that are expected to hold in a given relation, such knowledge is lacking or incomplete in many data science scenarios [33]. As such, *functional dependency discovery* is a fundamental task in data profiling: given a relation, derive the largest set of FDs that are satisfied in the relation. In so doing, we hence reverse engineer the design schema. FD discovery is an established field [1, 2, 22] and a multitude of algorithms have been proposed: see [32] for an experimental comparison and, e.g., [4, 7, 33, 43] for recent developments in this area.

Unfortunately, in practice, it may happen that the input relation itself does not satisfy the original set of FDs created during the database design phase: e.g., due to errors during data entry. In such a case, exact FD discovery algorithms will also fail to correctly reverse engineer the design schema. For this reason, the FD discovery problem should be broadened to discover *approximate functional dependencies* (AFDs) instead, that is, FDs that "almost hold" in the relation. Clearly, a key decision to make in the discovery of AFDs is when an FD "almost" holds. This decision is reflected in the adoption of an *AFD measure*, which formally quantifies the extent to which an FD holds approximately. Many such measures have been proposed in the literature [5, 19, 23, 25, 29, 30, 36, 38, 45].

While these measures vary widely in nature and are hence difficult to compare, there has been little study so far that contrasts and compares the measures. A notable exception is the work by Gianella and Robertson [19] who compare 3 measures both theoretically and empirically on 4 real world datasets where they report on the observed average difference for each pair of measures. In light of the multitude of measures that have been proposed, however, a clear guideline for deciding which AFD measure to adopt is currently lacking. As such, the central question that we aim to address in this paper is the following.

> *Which AFD measures are best suited to discover approximate functional dependencies in practice?*

To respond, we adopt the following methodology:

(1) We present a survey of the AFD measures that have been proposed in the literature and provide a new, formal, and uniform presentation. Specifically, we discern three classes of measures: (i) measures that quantify the fraction of violations; (ii) measures based on Shannon entropy [15]; and (iii) measures based on logical entropy [16]. We highlight the similarities and differences of measures in and between these classes.

(2) We compare the effectiveness of AFD measures on real-world data. A difficulty that we face in this respect is that existing benchmarks for exact FD discovery are designed to gauge algorithmic efficiency and typically do not contain the design schema which encodes the "ground truth" set of FDs to compare against for AFD

discovery. In response, we create a new benchmark for AFD discovery, denoted RWD, obtained by manually creating design schemas for existing benchmark relations.

(3) We study the measures' sensitivity to different kinds and different levels of errors. To that end, we create a new, polluted dataset from RWD (denoted RWD^e) by taking the design FDs that are satisfied in RWD and adding data errors in a principled fashion, hence turning them effectively into AFDs. In addition, we also study sensitivity to error on completely synthetic data.

(4) We study the measures' sensitivity to structural properties of the dataset to shed light on their implicit biases. Consider the following statistics which are defined w.r.t. an FD $X \to Y$ and a relation R: (i) *LHS-uniqueness*: the normalized number of unique values occurring in $\pi_X(R)$; and, (ii) *RHS-skew*: the skewness of the distribution of values occurring in $\pi_Y(R)$. Here, an RHS-skew of 0 means that values in Y are uniformly distributed while higher values quantify dominance of certain values. As these statistics refer to only X or to only Y (but never to both), they by themselves do not provide a good signal for discovering $\boldsymbol{X} \to \boldsymbol{Y}$. Nevertheless, some AFD measures like g_3 [25] would attribute a high score to *student_number* $\to$ *gender* when *gender* contains a dominating value, while other measures correct for this behavior. Similar to this, it is well known that as LHS-uniqueness rises, metrics based on Shannon entropy increase as well [29, 30], even when X and Y are generated independently at random. We therefore compare the sensitivity of all measures to both statistics on synthetic data.

Our conclusions are as follows. (i) The measure μ [38], which is based on logical entropy and was proposed in 1993 but has received little attention since, is the most effective measure on RWD and RWD^e. An additional advantage is that it is *not* biased w.r.t. LHS-uniqueness or RHS-skew.

(ii) The measure g_3, which is widely known and cited [4, 5, 18, 19, 21, 24, 25], is second most effective on RWD but *only* when it is correctly normalized. To the best of our knowledge only [19] considers the correctly normalized version to which we refer as g'_3 in this paper. Additionally, g'_3 is the second most effective measure on RWD^e as well but only for one error type. Furthermore, g_3 is biased w.r.t. LHS-uniqueness whereas g'_3 is *not*, and both measures are biased w.r.t. RHS-skew.

(iii) We find that the Shannon entropy-based measure *fraction of information* (FI [19]) is the least effective one when applied to real-world data. The correction SFI [36] makes FI worse while the correction RFI [29, 30] does improve over FI but is not capable of raising it to the level of most logical entropy-based measures (including μ), g_3 or g'_3. Over the polluted dataset RWD^e, we do see that when error levels rise, RFI' continues to improve but only surpasses μ' for an error level of 10% and only for specific error types.

(iv) The effectiveness of all measures quickly deteriorates for increasing error levels making them essentially useless for error levels above 5%. We observe that the AFDs in RWD have an error level smaller than 1%.

(v) We illustrate on RWD, perhaps contrary to popular belief, that by only inspecting a small number of candidate FDs that are ranked high (according to μ), one already succeeds in finding a large number of true design FDs that were obscured by errors.

Remark. The objective of this paper is a comparative study of the AFD-measures that have been considered in the literature. A related but orthogonal issue is the study of AFD discovery algorithms itself. Indeed, AFD discovery algorithms usually fix a way to quantify the approximateness of an FD– typically through the choice of an AFD-measure – but then combine a multitude of techniques to do the actual discovery. One outcome of this paper is that it is worthwhile to consider μ as the basis for AFD discovery as well. For purposes of illustration, we present examples of AFD discovery algorithms based on each measure in Section 3.

Outline. We provide the necessary background in Section 2. We survey and formally define the AFD measures in Section 3. We compare the measures on real world and synthetic data in Section 4. We discuss related work in Section 5. Our conclusions have already been discussed in the Introduction.

2 Preliminaries

We assume given a fixed set of attributes, where each attribute X has a domain $dom(X)$ of possible data values.We use uppercase letters X, Y, Z to denote attributes and boldface type like $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ to denote sets of attributes. Lowercase $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$ denote tuples over these sets. Formally, as usual, a tuple over $\boldsymbol{X}$ is a mapping $\boldsymbol{x}$ that assigns each attribute $X \in \boldsymbol{X}$ to a value $\boldsymbol{x}(X) \in dom(X)$. We write $\boldsymbol{x} \colon \boldsymbol{X}$ to indicate that $\boldsymbol{x}$ is a tuple over $\boldsymbol{X}$, and write $\boldsymbol{x}|_{\boldsymbol{Y}}$ for the restriction of $\boldsymbol{x}$ to $\boldsymbol{Y} \subseteq \boldsymbol{X}$. We use juxtapostion like $\boldsymbol{XY}$ to denote the union $\boldsymbol{X} \cup \boldsymbol{Y}$ of two sets of attributes, and also apply this notation to tuples: if $\boldsymbol{x} \colon \boldsymbol{X}$ and $\boldsymbol{y} \colon \boldsymbol{Y}$ with $\boldsymbol{X}$ and $\boldsymbol{Y}$ disjoint, then $\boldsymbol{xy}$ is the tuple that equals $\boldsymbol{x}$ on all attributes in $\boldsymbol{X}$ and $\boldsymbol{y}$ on all attributes in $\boldsymbol{Y}$, i.e. $\boldsymbol{xy}|_{\boldsymbol{X}} = \boldsymbol{x}$ and $\boldsymbol{xy}|_{\boldsymbol{Y}} = \boldsymbol{y}$.

We will work with bag-based relations. Formally, a relation over $\boldsymbol{X}$ (also called $\boldsymbol{X}$-relation) is a mapping R that assigns a natural number $R(\boldsymbol{x}) \in \mathbb{N}$ to each tuple $\boldsymbol{x} \colon \boldsymbol{X}$. We also call $R(\boldsymbol{x})$ the frequency of $\boldsymbol{x}$ in R. We require relations to be finite in the sense that $R(\boldsymbol{x})$ can be non-zero for at most a finite number of $\boldsymbol{x}$. In what follows, we write $\boldsymbol{x} \in R$ to denote that $R(\boldsymbol{x}) > 0$ and stress that R is an $\boldsymbol{X}$-relation by means of the notation $R(\boldsymbol{X})$. If $\boldsymbol{Y} \subseteq \boldsymbol{X}$ then we denote by $dom_R(\boldsymbol{Y})$ the set of $\boldsymbol{Y}$-tuples that occur in R, $dom_R(\boldsymbol{Y}) = \{\boldsymbol{x}|_{\boldsymbol{Y}} \mid \boldsymbol{x} \in R\}$. We denote by $|R|$ the total number of tuples in R, i.e. $|R| = \sum_{\boldsymbol{x} \colon \boldsymbol{X}} R(\boldsymbol{x})$. We denote bag-based relational projection and selection as usual by $\pi_{\boldsymbol{Y}}(R)$ and $\sigma_{\boldsymbol{X}=\boldsymbol{x}}(R)$, respectively.

Functional Dependencies. A *functional dependency* (FD for short) is an expression φ of the form $\boldsymbol{X} \to \boldsymbol{Y}$. A relation $R(\boldsymbol{W})$ with $\boldsymbol{X}, \boldsymbol{Y} \subseteq \boldsymbol{W}$ *satisfies* φ if for all tuples $\boldsymbol{w}, \boldsymbol{w'} \in R$ we have that $\boldsymbol{w}|_{\boldsymbol{Y}} = \boldsymbol{w'}|_{\boldsymbol{Y}}$ whenever $\boldsymbol{w}|_{\boldsymbol{X}} = \boldsymbol{w'}|_{\boldsymbol{X}}$. We write $R \models \varphi$ to indicate that R satisfies φ, and $R \not\models \varphi$ to indicate that it violates φ. In what follows, we always implicitly assume that $\boldsymbol{X}$ and $\boldsymbol{Y}$ are disjoint when considering FDs.

Dependency Discovery A *schema* is a finite set of FDs. In the *exact FD discovery problem* we are given a relation R that satisfies all FDs in some fixed design schema Δ, but have no knowledge of Δ itself. We are then asked to recover Δ by deriving the largest set $\Lambda \supseteq \Delta$ of functional dependencies that are satisfied by R.

In the *approximate FD discovery problem*, instead we are given a relation R that, e.g. due to data entry errors, itself does not satisfy Δ and again we are asked to recover Δ. Here, we assume that R is obtained by means of a noisy channel process as follows. First, a

clean relation R' is created that satisfies Δ. Next, R is obtained from R' by modifying certain values in tuples in R. We consider an *error* in R to correspond to a cell c for which R differs from the clean version R'. Note that by running exact FD discovery algorithms on R, we will still be able to recover the FDs in Δ that are satisfied in R. Our interest in this paper is in *approximate FD discovery*, i.e., deriving the FDs in Δ that are violated in R because of the errors introduced, and that therefore cannot be discovered by exact FD discovery algorithms.

In Section 3 we will survey various measures that have been proposed to quantify the level to which an FD holds approximately. As we will see, by and large, most of these measures are based on exploiting notions of entropy. However, some employ Shannon entropy while others are based on *logical* entropy. We introduce these notions next.

Probabilities. Both notions of entropy are defined w.r.t. a given joint probability distribution. In our setting, this probability distribution is defined by the relation R under consideration. Specifically, let $R(\boldsymbol{W})$ be a relation. The joint probability distribution $p_R(\boldsymbol{W})$ over $\boldsymbol{W}$ induced by R is defined by $p_R(\boldsymbol{W} = \boldsymbol{w}) = \frac{R(\boldsymbol{w})}{|R|}$. As such, $p_R(\boldsymbol{W} = \boldsymbol{w})$ is the probability of observing $\boldsymbol{w}$ when randomly drawing a tuple from R. We note that this probability distribution is only well-defined when R is non-empty. Because the empty relation vacuously satisfies all FDs, we will implicitly assume without loss of generality in the rest of this paper that relations are non-empty, so that we may always use the probability distribution.

The notions of marginal and conditional distributions derived from $p_R(\boldsymbol{W})$ are defined as follows. For the remainder of the section, let $\boldsymbol{X}, \boldsymbol{Y} \subseteq \boldsymbol{W}$ be disjoint subsets of $\boldsymbol{W}$. Then, $p_R(\boldsymbol{Y})$ denotes the marginal probability distribution on $\boldsymbol{Y}$-tuples in R, while $p_R(\boldsymbol{Y} \mid \boldsymbol{X} = \boldsymbol{x})$ is the conditional distribution on $\boldsymbol{Y}$ given $\boldsymbol{X} = \boldsymbol{x}$:

$$p_R(\boldsymbol{Y} = \boldsymbol{y}) = \sum_{\boldsymbol{w}: \boldsymbol{W} \text{ s.t. } \boldsymbol{w}|_{\boldsymbol{Y}} = \boldsymbol{y}} p_R(\boldsymbol{W} = \boldsymbol{w}),$$

$$p_R(\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{X} = \boldsymbol{x}) = \frac{p_R(\boldsymbol{XY} = \boldsymbol{xy})}{p_R(\boldsymbol{X} = \boldsymbol{x})}.$$

It is readily verified that $p_R(\boldsymbol{Y})$ equals the distribution induced by $\pi_{\boldsymbol{Y}}(R)$, while $p_R(\boldsymbol{Y} \mid \boldsymbol{X} = \boldsymbol{x})$ equals the distribution induced by $\pi_{\boldsymbol{Y}}\sigma_{\boldsymbol{X}=\boldsymbol{x}}(R)$.

To simplify notation in what follows, we adopt the convention that lowercase values and tuples are always over the corresponding uppercase (set of) attributes. Hence, $\boldsymbol{x}$, $\boldsymbol{y}$, and $\boldsymbol{w}$ represent generic tuples over $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{W}$, respectively. Under this convention it is redundant to specify $\boldsymbol{X}$, $\boldsymbol{Y}$, and $\boldsymbol{W}$ in distributions, and we hence simply write $p_R(\boldsymbol{w})$, $p_R(\boldsymbol{y})$ and $p_R(\boldsymbol{y} \mid \boldsymbol{x})$ instead of $p_R(\boldsymbol{W} = \boldsymbol{w})$, $p_R(\boldsymbol{Y} = \boldsymbol{y})$ and $p_R(\boldsymbol{Y} = \boldsymbol{y} \mid \boldsymbol{X} = \boldsymbol{x})$, respectively.

Shannon Entropy. We write $H_R(\boldsymbol{X})$ for the *Shannon entropy* of $\boldsymbol{X}$ in R, defined as usual [15] by[1]

$$H_R(\boldsymbol{X}) = -\sum_{\boldsymbol{x}: \boldsymbol{X}} p_R(\boldsymbol{x}) \log p_R(\boldsymbol{x}).$$

$H_R(\boldsymbol{X})$ reflects the average level of uncertainty inherent in the possible tuples over $\boldsymbol{X}$ in $\pi_{\boldsymbol{X}}(R)$. The *conditional entropy* $H_R(\boldsymbol{Y} \mid \boldsymbol{X})$ is the uncertainty in $\boldsymbol{Y}$ given $\boldsymbol{X}$, defined as

$$H_R(\boldsymbol{Y} \mid \boldsymbol{X}) = -\sum_{\boldsymbol{x}: \boldsymbol{X}, \boldsymbol{y}: \boldsymbol{Y}} p_R(\boldsymbol{xy}) \log \frac{p_R(\boldsymbol{xy})}{p_R(\boldsymbol{x})}.$$

Equivalently, denoting by $H_R(\boldsymbol{Y} \mid \boldsymbol{x})$ the Shannon entropy of $\boldsymbol{Y}$ in the conditional distribution $p_R(\boldsymbol{Y} \mid \boldsymbol{X} = \boldsymbol{x})$, we see that $H_R(\boldsymbol{Y} \mid \boldsymbol{X})$ is the expected value of $H_R(\boldsymbol{Y} \mid \boldsymbol{x})$, taken over all $\boldsymbol{x}$, i.e., $H_R(\boldsymbol{Y} \mid \boldsymbol{X}) = \mathbb{E}_{\boldsymbol{x}}[H_R(\boldsymbol{Y} \mid \boldsymbol{x})]$.

Logical Entropy. The *logical entropy* of $\boldsymbol{X}$ in R is the probability that two tuples $\boldsymbol{w}$ and $\boldsymbol{w'}$, drawn randomly with replacement from R according to p_R, differ in some attribute in $\boldsymbol{X}$. That is,

$$h_R(\boldsymbol{X}) := 1 - \sum_{\boldsymbol{x}: \boldsymbol{X}} p_R(\boldsymbol{x})^2.$$

Here, $p_R(\boldsymbol{x})^2$ is the probability that two random tuples are exactly equal to $\boldsymbol{x}$ on X.

We denote by $h_R(\boldsymbol{Y} \mid \boldsymbol{x})$ the logical entropy of $\boldsymbol{Y}$ in the conditional distribution $p_R(\boldsymbol{Y} \mid \boldsymbol{X} = \boldsymbol{x})$, i.e.,

$$h_R(\boldsymbol{Y} \mid \boldsymbol{x}) = 1 - \sum_{\boldsymbol{y}: \boldsymbol{Y}} p_R(\boldsymbol{y} \mid \boldsymbol{x})^2.$$

The *logical conditional entropy* of $\boldsymbol{Y}$ given $\boldsymbol{X}$ in R, denoted $h_R(\boldsymbol{Y} \mid \boldsymbol{X})$ is the probability that two tuples drawn at random with replacement from R according to p_R are equal in all attributes of $\boldsymbol{X}$ but differ in some attribute of $\boldsymbol{Y}$,

$$h_R(\boldsymbol{Y} \mid \boldsymbol{X}) := \sum_{\boldsymbol{x}, \boldsymbol{y}} p_R(\boldsymbol{xy})[p_R(\boldsymbol{x}) - p_R(\boldsymbol{xy})].$$

Here, the factor $p_R(\boldsymbol{xy})$ expresses the probability of observing $\boldsymbol{xy}$ in the first tuple and the factor $p_R(\boldsymbol{x}) - p_R(\boldsymbol{x}, \boldsymbol{y})$ is the probability that the second tuple has the same value for $\boldsymbol{x}$ but differs in $\boldsymbol{y}$.

Note that, in contrast to the case of Shannon entropy where $H_R(\boldsymbol{Y} \mid \boldsymbol{X}) = \mathbb{E}_{\boldsymbol{x}}[H_R(\boldsymbol{Y} \mid \boldsymbol{x})]$, in logical entropy $h_R(\boldsymbol{Y} \mid \boldsymbol{X}) \neq \mathbb{E}_{\boldsymbol{x}}[h_R(\boldsymbol{Y} \mid \boldsymbol{x})]$.

Discussion. The notion of logical entropy arises in mathematical philosophy [16], where it is observed to provide a theory of information based on logic. Importantly, formulas and equalities concerning logical entropy can be converted into corresponding formulas and equalities concerning Shannon entropy by the so-called dit-bit transform (see [16]). Logical and Shannon entropy are hence highly similar, but measure different things: logical entropy measures the probability of two random tuples to be distinguished, while Shannon entropy measures average uncertainty.

3 AFD Measures

In this section, we survey the literature on AFD measures.

AFD measures. Formally, an *AFD measure*, short for *approximate* FD measure, is a function that maps pairs (φ, R), with φ an FD and R a relation, to a number in the interval $[0, 1]$ that indicates the level to which φ holds in R. Higher values are intended to indicate that R makes fewer violations to φ, and it is required that $f(\varphi, R) = 1$ if R perfectly satisfies φ.

It is important to note that instead of defining AFD measures, some papers in the literature define *error measures* where a high value indicates a high number of errors against the FD, and low

[1] Here and in the sequel we use the common convention that $0 \log 0 = 0$ and $\frac{0}{0} = 0$.

values indicate fewer violations. In what follows, we routinely redefine such error measures e into an AFD measure f_e by setting $f_e(\varphi, R) := 1 - e(\varphi, R)$.

Every AFD measure f naturally gives rise to an associated AFD discovery algorithm. Indeed, from an abstract viewpoint, an AFD discovery algorithm simply consists of a fixed AFD measure f and a threshold $\epsilon \in [0, 1]$. Given a relation $R(\boldsymbol{W})$ the algorithm returns all FDs over $\boldsymbol{W}$ that are not satisfied by R and whose f-value lies in the range $[\epsilon, 1]$.

Interpretation and baselines. As with all threshold-based algorithms, a key difficulty for AFD discovery algorithms lies in determining the correct threshold ϵ to use. At its core, this question boils down to how we should interpret the significance of the values returned by f. It is tempting to see the values of f as a percentage with $f(\varphi, R) = 1$ indicating that R perfectly satisfies φ and $f(\varphi, R) = 0$ indicating that R completely fails to satisfy φ. This interpretation, however, is only valid if the measure has a notion of R "completely failing to satisfy" φ. In particular, this is only the case when there are relations for which $f(\varphi, R) = 0$: those relations are the completely failing ones. In what follows, we call a relation R with $f(\varphi, R) = 0$ a *baseline of f for φ*. If f has a baseline for every FD φ then we say that f *has baselines*, otherwise we call f *without baselines*. Only if f has baselines can we interpret $f(\varphi, R)$ as a percentage between completely not satisfying φ and completely satisfying it. Some measures are without baselines, as we will see, and for those measures such interpretation is not possible.

Conventions and organisation. Throughout this section, assume that R is a $\boldsymbol{W}$-relation, let $\boldsymbol{X}, \boldsymbol{Y}$ be disjoint subsets of $\boldsymbol{W}$ and let $\varphi = \boldsymbol{X} \to \boldsymbol{Y}$. We convene that for all measures f that we describe, we trivially set $f(\varphi, R) := 1$ if $R \models \varphi$. So, the definitions that follow only apply when $R \not\models \varphi$. In that case, observe that R must be non-empty, that $|dom_R(\boldsymbol{X})| \neq |R|$ and that $|dom_R(\boldsymbol{Y})| > 1$ since otherwise R trivially satisfies $\boldsymbol{X} \to \boldsymbol{Y}$. As a consequence, $H_R(\boldsymbol{Y}) > 0$ and $h_R(\boldsymbol{Y}) > 0$. This ensures that the denominator of fractions in the formulas that follow are never zero.

We next formally introduce the measures in Sections 3.1–3.4. Subsequently, in Section 3.5 we divide measures into three classes, and discuss similarities in the design of measures between these classes.

3.1 Co-occurrence ratio

Ilyas et al. [23] consider the derivation of AFDs (called *soft* FDs in their paper) as well as general correlations between attributes. To derive AFDs, they consider the ratio between the number of distinct $\boldsymbol{X}$-tuples and the number of distinct $\boldsymbol{XY}$-tuples occurring in R. We denote this measure by ρ, formally defined as:

$$\rho(\boldsymbol{X} \to \boldsymbol{Y}, R) := \frac{|\, dom_{\boldsymbol{X}}(R)|}{|\, dom_{\boldsymbol{XY}}(R)|}.$$

This ratio is 1 when R satisfies $\boldsymbol{X} \to \boldsymbol{Y}$ and decreases when more $\boldsymbol{y}$-tuples occur with the same $\boldsymbol{x}$-tuple. Note that ρ is a set-based measure, as it ignores the multiplicities of the tuples in R. It is also without baselines, as $|dom_{\boldsymbol{X}}(R)| > 0$ for any non-empty relation R and as, by convention, $\rho(\varphi, R) = 1$ when R is empty.

3.2 g-measures

Kivinen and Mannila [25] introduced three error measures on set-based relations. Generalized to bag-based relations, and converted to AFD measures, these are the following.

The measure g_1. The measure g_1 is based on logical entropy. Specifically, Kivinen and Manila defined g_1 to reflect the (normalized) number of violating pairs in R. Here, a pair $(\boldsymbol{w}, \boldsymbol{w}')$ of R-tuples is a *violating pair* if they are equal on $\boldsymbol{X}$ but differ on $\boldsymbol{Y}$. Formally, if we denote the bag of violating pairs in $R \times R$ by $G_1(\boldsymbol{X} \to \boldsymbol{Y}, R)$ then, converted to an AFD measure instead of an error measure, g_1 is defined as

$$\begin{aligned} g_1(\boldsymbol{X} \to \boldsymbol{Y}, R) &:= \frac{|R|^2 - |G_1(\boldsymbol{X} \to \boldsymbol{Y}, R)|}{|R|^2} \\ &= 1 - \frac{|G_1(\boldsymbol{X} \to \boldsymbol{Y}, R)|}{|R|^2} \\ &= 1 - h_R(\boldsymbol{Y} \mid \boldsymbol{X}). \end{aligned}$$

In other words, g_1 is maximized when the logical conditional entropy is minimized.

The measure g_1 is without baselines: because pairs of the form $(\boldsymbol{w}, \boldsymbol{w})$ are never violating, it is straightforward to see that the total number of violating pairs is bounded from above by $|R|^2 - \sum_{\boldsymbol{w}} R(\boldsymbol{w})^2$. We denote by g_1' the normalized version of g_1,

$$g_1'(\boldsymbol{X} \to \boldsymbol{Y}, R) := 1 - \frac{|G_1(\boldsymbol{X} \to \boldsymbol{Y}, R)|}{|R|^2 - \sum_{\boldsymbol{w}} R(\boldsymbol{w})^2}.$$

The baselines of g_1' are hence those relations for which the set $G_1(\boldsymbol{X} \to \boldsymbol{Y}, R)$ consists of all possible violating pairs.

Both g_1 and g_1' have been used as the basis of AFD discovery algorithms. In particular, g_1 is the basis of Fdx [49] while g_1' is the basis of Pyro [27]. Adaptations of g_1' are also used in the context of denial constraints [35] and roll-up dependencies [8].

The measure g_2. Kivinen and Manila defined g_2 to reflect the probability that a random tuple participates in a violating pair. Formally, define $G_2(\boldsymbol{X} \to \boldsymbol{Y}, R)$ to be the set of all tuples in R that participate in a violating pair,

$$G_2(\boldsymbol{X} \to \boldsymbol{Y}, R) := \{\boldsymbol{w} \in R \mid \exists \boldsymbol{w}' \in R, (\boldsymbol{w}, \boldsymbol{w}') \in G_1(\boldsymbol{X} \to \boldsymbol{Y}, R)\}.$$

Then, g_2, converted to an AFD measure instead of an error measure as originally proposed, computes the probability that a tuple, drawn randomly from R according to p_R, is not part of a violating pair,

$$g_2(\boldsymbol{X} \to \boldsymbol{Y}, R) := 1 - \sum_{\boldsymbol{w} \in G_2(\boldsymbol{X} \to \boldsymbol{Y}, R)} p_R(\boldsymbol{w}).$$

The FD-compliance-ratio that is used as one of the building blocks in UNI-DETECT [46] is based on g_2.

The measure g_3. The measure g_3 computes the relative size of a maximal subrelation of R for which $\boldsymbol{X} \to \boldsymbol{Y}$ holds. Specifically, define $R'(\boldsymbol{W})$ to be a subrelation of $R(\boldsymbol{W})$, denoted $R' \subseteq R$, if $R'(\boldsymbol{w}) \leq R(\boldsymbol{w})$ for all $\boldsymbol{w} \colon \boldsymbol{W}$. Let $G_3(\boldsymbol{X} \to \boldsymbol{Y}, R)$ denote the set of all subrelations of R that satisfy $\boldsymbol{X} \to \boldsymbol{Y}$,

$$G_3(\boldsymbol{X} \to \boldsymbol{Y}, R) := \{R' \mid R' \subseteq R, R' \models \boldsymbol{X} \to \boldsymbol{Y}\}.$$

Then g_3 is defined as the maximum relative size of a subrelation satisfying $X \to Y$:

$$g_3(X \to Y, R) := \max_{R' \in G_3(X \to Y, R)} \frac{|R'|}{|R|}.$$

Note that $1 - g_3(X \to Y, R)$ can can naturally be interpreted as the minimum fraction of tuples that need to be removed for $X \to Y$ to hold in R.

The measure g_3 is without baselines. Indeed, for any non-empty R we can always obtain a subrelation $R' \in G_3(\varphi, R)$ of size $|dom_X(R)|$ by arbitrarily fixing one y-value for each x-value. As such, g_3 is lower bounded by $\frac{|dom_X(R)|}{|R|} > 0$. Gianella and Robertson [19] proposed a normalized variant g_3' of g_3, defined as follows.

$$g_3'(X \to Y, R) := \max_{R' \in G_3(X \to Y, R)} \frac{|R'| - |dom_R(X)|}{|R| - |dom_R(X)|}.$$

This variant has baselines, namely all relations R for which no subrelation $R' \in G_3(\varphi, R)$ is larger than $|dom_R(X)|$.

The unnormalized measure g_3 is used in multiple AFD discovery algorithms [4, 21, 24, 25]. Furthermore, the 'per-tuple' probability of an FD as defined in [45] is precisely g_3. Exact and approximate solutions for the computation of g_3 are proposed in [18]. In addition, Berzal et al. [5] use it as the basis for relational decomposition based on AFDs instead of FDs. We note that g_3 has been generalized to other dependencies as well: e.g., conditional FDs [14, 39], inclusion dependencies [31], and conditional matching dependencies [47]. By contrast, apart from [19] we know of no other work that considers the normalized version g_3'.

3.3 Fraction of information

Fraction of Information. Cavallo and Pittarelli [11] introduced *fraction of information* (FI) as a way to generalize FDs from deterministic to probabilistic databases. Usage of FI as an AFD measure was later studied by Giannelli and Robertson [19]. FI is based on Shannon entropy and is formally defined as

$$FI(X \to Y, R) := \frac{H_R(Y) - H_R(Y \mid X)}{H_R(Y)}.$$

The numerator $H_R(Y) - H_R(Y \mid X)$ is known as *mutual information* [15], which we further denote by $I_R(X;Y)$.

We can understand FI as follows. $H_R(Y)$ measures the uncertainty of observing Y while $H_R(Y \mid X)$ measures the uncertainty of observing Y after observing X. FI hence represents the proportional reduction of uncertainty about Y that is achieved by knowing X. When R satisfies $X \to Y$ there is no uncertainty about Y after observing X and hence $H_R(Y \mid X) = 0$ and so FI is 1. Conversely, when X and Y are independent random variables in p_R then there is no reduction in uncertainty, and hence $H_R(Y \mid X) = H_R(Y)$ and so FI is 0. Thus, the baselines of FI for $X \to Y$ are those relations R for which X and Y are independent in p_R.

Bias. Mandros et al. [29, 30] and Pennerath et al. [36] proposed two refinements to FI specifically for AFD discovery, called *reliable FI* (RFI) and *smoothed FI* (SFI), respectively. They are motivated in proposing these refinements by the following observation. Consider a relation $S(W)$ and assume that we are given relation $R(W)$ of size n that is obtained by sampling n tuples from S according to distribution p_S. Further assume that we do not have access to S and wish to determine $FI(X \to Y, S)$ based on R. Then a result by Roulston [41] states that the expected value of $I_R(X;Y)$, taken over all R obtained in this manner, equals

$$I_S(X;Y) + \frac{|dom_S(XY)| - |dom_S(X)| - |dom_S(Y)| + 1}{2n}.$$

In other words, we may expect $I_R(X;Y)$ to overestimate $I_S(X;Y)$ and the magnitude of overestimation depends on the size of the active domains of XY, X, and Y in S, as well as on n. Because in addition $H_R(Y)$ underestimates $H_S(Y)$ [41], we may conclude that $FI(X \to Y, R)$ is expected to overestimate $FI(X \to Y, S)$ and the magnitude of overestimation depends on the active domain sizes and the size of S. This overestimation is problematic since even if X and Y are independent in p_S, and $FI(X \to Y, S)$ is hence 0, $FI(X \to Y, R)$ will be quite large.

Reliable FI. Reliable FI corrects for this bias by subtracting the mutual information value that is expected under random $(X;Y)$-permutations.

Definition 3.1. Relation R' is an $(X;Y)$-*permutation* of R, denoted $R \sim_{X;Y} R'$ if (i) $|R| = |R'|$; (ii) $\pi_X(R) = \pi_X(R')$; and (iii) $\pi_Y(R) = \pi_Y(R')$.

In particular, R' and R have the same marginal distributions on X and on Y, $p_{R'}(X) = p_R(X)$ and $p_{R'}(Y) = p_R(Y)$. In what follows, for a measure f, we denote by $\mathbb{E}_R[f(X \to Y, R))]$ the expected value of $f(X \to Y, R)$ where the expectation is taken over all $(X;Y)$-permutations of R.

Reliable fraction of information is then defined as

$$RFI(X \to Y, R) := FI(X \to Y, R) - \mathbb{E}_R[FI(X \to Y, R)].$$

Note that there are $|R|!$ $(X;Y)$-permutations of R, so we may compute $\mathbb{E}_R[FI(X \to Y, R)]$, and hence also $RFI(X \to Y, R)$ by simply computing $FI(X \to Y, R')$ for every permutation R' of R and taking the average. More efficient algorithms are proposed in [29, 30].

We note that, strictly speaking RFI is not an AFD measure since it can become negative when $FI(\varphi, R) \leq \mathbb{E}_R[FI(\varphi, R)]$. Because such negative RFI values indicate that there is weak evidence to conclude that φ is an AFD, we turn RFI into an actual AFD measure RFI' by setting

$$RFI'(X \to Y, R) := \max(RFI(X \to Y, R), 0).$$

The baselines of RFI' for $X \to Y$ are hence all relations whose FI value is smaller or equal than the expected value under random permutations.

Smoothed FI. Smoothed FI uses *laplace smoothing* to reduce bias. Laplace smoothing is a well-known statistical technique to reduce estimator variance. It is parameterized by a value $\alpha > 0$. Specifically, for a relation $S(XY)$ let $S^{(\alpha)}$ denote the α-smoothed version of S, defined by $S^{(\alpha)}(xy) := S(xy) + \alpha$ for every $x \in dom_S(X)$ and $y \in dom_S(Y)$. Note in particular, that it is possible that $S(xy) = 0$, in which case $S^{(\alpha)}(xy) = \alpha$. Then the smoothed FI of R is simply the normal FI of the α-smoothed version of $\pi_{XY}(R)$:

$$SFI_\alpha(X \to Y, R) := FI(X \to Y, \pi^{(\alpha)}_{XY}(R)).$$

AFD discovery algorithms based on RFI and SFI are presented in [29, 30] and [36], respectively.

3.4 Probabilistic dependency, τ and μ

Piatetsky-Shapiro and Matheus [38] proposed *probabilistic dependency* as another probabilistic generalization of a functional dependency. They also introduced a normalized version of probabilistic dependency, which is equivalent to the Goodman and Kruskal τ measure of association [20]. Finally, they also propose a rescaled version of τ. All three notions are defined as follows. It is worth noting, that, apart from [38], we are not aware of any work that considers these measures for AFD discovery in the database context, let alone designs AFD discovery algorithms for them.

Probabilistic dependency. The *probabilistic dependency of Y on X in R*, denoted by $pdep(X \to Y, R)$, represents the conditional probability that two tuples drawn randomly with replacement from R are equal on Y, given that they are equal on X. Formally,

$$pdep(X \to Y, R) := \sum_{x} p_R(x)\, pdep(Y \mid x, R),$$

where $pdep(Y \mid x, R)$ is the probability that two random Y-tuples drawn with replacement from the conditional distribution $p_R(Y \mid x)$ are equal:

$$pdep(Y \mid x, R) := \sum_{y} p_R(y \mid x)^2 = 1 - h_R(Y \mid x).$$

Probabilistic dependency is hence a measure based on logical entropy. It can be understood as follows. Suppose that we are given two tuples that equal x on X. Then $pdep(Y \mid x, R)$ is the probability that these tuples are also equal on Y, and $pdep(X \to Y, R)$ is the expected value of $pdep(Y \mid x, R)$ over all x.

We note that probabilistic dependency can also be seen as a generalization of the measure g_2. Whereas g_2 computes the probability that a random tuple cannot be extended to a violating pair, probabilistic dependency computes the average conditional probability that a given X-tuple x cannot be extended to a violating pair, where the average is taken over all values of X.

The measure τ. It is straightforward to see that $pdep(X \to Y, R) > 0$, always. As such, *pdep* is a measure without baselines. In fact, Piatetsky-Shapiro and Matheus [38] show that we always have

$$pdep(X \to Y, R) \geq pdep(Y, R)$$

where $pdep(Y, R)$, called *probabilistic self-dependency*, is defined as the probability that two random tuples in R have equal Y attributes,

$$pdep(Y, R) := \sum_{y} p_R(y)^2 = 1 - h_R(Y).$$

To account for the relationship between $pdep(X \to Y, R)$ and $pdep(Y, R)$, Piatetsky-Shapiro and Matheus propose to normalize $pdep(X \to Y, R)$ w.r.t. $pdep(Y, R)$. The resulting measure is equivalent to the Goodman and Kruskal τ (tau) measure of association [20], which is defined as

$$\tau(X \to Y, R) := \frac{pdep(X \to Y, R) - pdep(Y, R)}{1 - pdep(Y, R)}.$$

Piatetsky-Shapiro and Matheus explain τ in the following way [38]. Suppose we are given a tuple drawn randomly from R according to p_R, and we need to guess its Y value. One strategy is to make guesses randomly according to the marginal distribution of Y, i.e. guess value $Y = y$ with probability $p_R(y)$. Then the probability for a correct guess is $pdep(Y, R)$. If we also know that item has $X = x$, we can improve our guess using conditional probabilities of Y, given that $X = x$. Then our probability for success, averaged over all values of X, is $pdep(X \to Y, R)$, and $\tau(X \to Y, R)$ is the relative increase in our probability of successfully guessing Y, given X. The baselines of τ for $X \to Y$ are hence those relations where this relative increase is zero.

The measure μ. Piatetsky-Shapiro and Matheus [38] note that *pdep* and τ have the following undesirable property.

Theorem 3.2 (Piatetsky-Rotem-Shapiro [38]). *Given a random relation R of size $N \geq 2$ containing attributes X and Y, where X has $K = |\, dom_R(X)|$ distinct values in its active domain, the expected values of pdep and τ under random permutations of R are*

$$\mathbb{E}_R[pdep(X \to Y, R)] = pdep(Y, R) + \frac{K - 1}{N - 1}(1 - pdep(Y, R)),$$

$$\mathbb{E}_R[\tau(X \to Y, R)] = \frac{|\, dom_R(X)| - 1}{|R| - 1}.$$

Thus, for a fixed distribution of Y values, $\mathbb{E}_R[pdep(X \to Y, R)]$ depends only on the number of distinct X values and not on their relative frequency. Moreover, the formula for $\mathbb{E}_R[\tau(X \to Y, R)$ tells us that if we have two candidate AFDs with the same right hand side, $X \to Y$ and $Z \to Y$, then if $|\, dom_R(Z)| > |\, dom_R(X)|$, we may expect τ to score $Z \to Y$ better than $X \to Y$, regardless of any intrinsic better relationship between Z and Y over X and Y in R. In response, Piatetsky-Shapiro and Matheus compensate for this effect by introducing the measure μ which normalizes $pdep(X \to Y, R)$ with respect to $\mathbb{E}_R[pdep(X \to Y, R)]$ instead of $pdep(Y, R)$:[2]

$$\mu(X \to Y, R) := \frac{pdep(X \to Y, R) - \mathbb{E}_R[pdep(X \to Y, R)]}{1 - \mathbb{E}_R[pdep(X \to Y, R)]}$$

$$= 1 - \frac{1 - pdep(X \to Y, R)}{1 - pdep(Y, R)} \frac{|R| - 1}{|R| - |\, dom_R(X)|}$$

Strictly speaking, μ is not a measure since it returns negative values when $pdep(X \to Y, R) > \mathbb{E}_R[pdep(X \to Y, R)]$. Because such negative μ values indicate that there is weak evidence to conclude that φ is an AFD, we turn μ into an actual AFD measure μ by setting

$$\mu'(X \to Y, R) := \max(\mu(X \to Y, R), 0).$$

The baselines of μ' for $X \to Y$ are hence all relations where the $pdep(X \to Y)$ value is smaller or equa to the expected value under random permuations.

3.5 Discussion

Among the measures listed in the previous sections we discern the following three classes.

(1) The class of measures that have a notion of "violation" and quantify the number of violations, consisting of ρ, g_2, g_3, and g_3'. We denote this class by Simple.
(2) The class of measures based on Shannon entropy, consisting of FI, RFI', and SFI. We denote this class by Shannon.
(3) The class of measures based on logical entropy, consisting of $g_1, g_1', pdep, \tau$, and μ' and denoted by Logical.

[2] This fraction is ill-defined if the denominator $1 - \mathbb{E}_R[pdep(\varphi, R)] = 0$. This only happens, however, when $R \models \varphi$, which we have assumed not to be the case throughout this section, since we have already convened to set $\mu(\varphi, R) = 1$ whenever $R \models \varphi$. See the Appendix for a proof.

	Logical measure	Simple/Shannon
①	$g_1 = 1 - h_R(Y \mid X)$	$1 - H_R(Y \mid X)$
②	$pdep = \sum_{\boldsymbol{x}} p_R(\boldsymbol{x})\,(1 - h_R(Y \mid \boldsymbol{x}))$ $= 1 - \sum_{\boldsymbol{x}} p_R(\boldsymbol{x}) h_R(Y \mid \boldsymbol{x})$	$g_3 = \sum_{\boldsymbol{x}} p_R(\boldsymbol{x}) \max_{\boldsymbol{y}} p_R(\boldsymbol{y} \mid \boldsymbol{x})$ $g_2 = 1 - \sum_{\boldsymbol{w} \in G_2(X \to Y,R)} p_R(\boldsymbol{w})$
③	$\tau = 1 - \frac{\mathbb{E}_{\boldsymbol{x}}[h_R(Y\mid\boldsymbol{x})]}{h_R(Y)}$	$FI = 1 - \frac{H_R(Y\mid X)}{H_R(Y)}$ $= 1 - \frac{\mathbb{E}_{\boldsymbol{x}}[H_R(Y\mid\boldsymbol{x})]}{H_R(Y)}$
④	$\mu = \frac{pdep(\varphi,R) - \mathbb{E}_R[pdep(\varphi,R)]}{1 - \mathbb{E}_R[pdep(\varphi,R)]}$	$RFI = FI(\varphi, R) - \mathbb{E}_R[FI(\varphi, R)]$

Table 1: Overview of similarities between Logical measures and measures in Simple/ Shannon.

We observe that there are striking similarities in the design of Logical measures and measures in the Simple and Shannon class. We will discuss these similarities by means of Table 1, which clusters measures into groups that we find similar. There, we rewrite measures into equivalent form when this is necessary to stress the similarities.

Theorem 3.3. *The alternate formulas given in Table 1 are equivalent to their definition given in Sections 3.1–3.4.*

The proof may be found in the Appendix.

① We have already observed that g_1 is a measure based on logical entropy, $g_1(X \to Y, R) = 1 - h_R(Y \mid X)$. We find it interesting to observe that Giannella and Robertson [19] considered an axiomatisation of FD error measures, and showed that Shannon entropy $H_R(Y \mid X)$ is, up to a multiplicative constant, the unique unnormalized error measure that satisfies their axioms. As such, we may view $1 - H_R(Y \mid X)$ as the Shannon equivalent of g_1. Unfortunately, however, $1 - H_R(Y \mid X)$ is not an AFD measure: the value of $H_R(Y \mid X)$ is unbounded and $1 - H_R(Y \mid X)$ hence has range $[-\infty, 1]$ instead of $[0, 1]$. Giannella and Robertson [19] therefore turn $1 - H_R(Y \mid X)$ into an AFD measure by moving to FI, which normalizes $H_R(Y \mid X)$ w.r.t. $H_R(Y)$. This is no longer the conceptual Shannon counterpart of g_1, however, as further discussed below.

② We have already observed in Section 3.4 that we may view $pdep$ as a generalisation of g_2. We may also view it as an alternate to g_3. Indeed, $pdep$ equals the expected value of $1 - h_R(Y \mid \boldsymbol{x})$—expressing the probability of $\boldsymbol{x}$ not participating in a violating pair—while g_3 equals the expected value of $\max_{\boldsymbol{y}} p_R(\boldsymbol{y} \mid \boldsymbol{x})$—expressing the largest subgroup of non-violating tuples in $\pi_Y \sigma_{X=\boldsymbol{x}}(R)$, where in both cases expectation is taken over all $\boldsymbol{x}$.

③ As can be seen by the rewritten formulas in line 3 of Table 1, FI is simply the Shannon entropy-based version of τ.

④ The similarity between τ and FI extends to a conceptual similarity between μ and RFI: μ corrects for the bias of τ under random permutations while RFI corrects for the bias of FI under random permutations. Despite this conceptual similarity, note that the corrections are done differently: μ corrects by taking the *normalized* difference between $pdep$ and $\mathbb{E}_R[pdep]$ while RFI corrects by taking the *absolute* difference between FI and $\mathbb{E}_R[FI]$.

4 Evaluation

We compare the effectiveness of the described AFD measures on three benchmarks: (i) RWD: a set of real-world tables for which we have manually created the design schema that serves as the ground truth for comparison; (ii) RWD^e: obtained from RWD by adding errors in a controlled fashion to design FDs that are fully satisfied in RWD; and, (iii) SYN: a synthetically generated benchmark. These benchmarks are discussed in detail together with their evaluation goals in Section 4.2. The comparison itself is detailed in Sections 4.3–4.5. We first discuss the evaluation methodology in Section 4.1.

4.1 Methodology

We are interested in comparing the suitability of AFDs measures for the purpose of AFD discovery and employ the following methodology. For each benchmark $\mathcal{B} \in \{\text{RWD}, \text{RWD}^e, \text{SYN}\}$, and for each relation $R \in \mathcal{B}$ we have available the design schema of R, which we denote by $\Delta(R)$. This set of FDs is partitioned into two sets:

$$PFD(R) := \{\varphi \in \Delta(R) \mid R \models \varphi\}, \quad AFD(R) := \{\varphi \in \Delta(R) \mid R \not\models \varphi\}.$$

We will refer to the elements in these sets as the *perfect* (design) FDs and *approximate* (design) FDs, respectively. In particular, $AFD(R)$ forms the ground truth of FDs to discover during AFD discovery on R, and we denote by $AFD(\mathcal{B}) = \{(R, \varphi) \mid R \in \mathcal{B}, \varphi \in AFD(R)\}$ the entire set of design AFDs to discover in $\mathcal{B}$.

An *AFD discovery algorithm* is any algorithm A that, given a relation $R(\boldsymbol{W})$ returns a set $A(R)$ of FDs over the set of attributes $\boldsymbol{W}$ such that $R \not\models \varphi$, for any $\varphi \in A(R)$. Let $A(\mathcal{B}) = \{(R, \varphi) \mid R \in \mathcal{B}, \varphi \in A(R)\}$ be the entire set of FDs returned by A on $\mathcal{B}$. The precision and recall of A on benchmark $\mathcal{B}$ is defined as usual:

$$prec(A, \mathcal{B}) := \frac{|\,AFD(\mathcal{B}) \cap A(\mathcal{B})|}{|A(\mathcal{B})|} \qquad rcl(A, \mathcal{B}) := \frac{|\,AFD(\mathcal{B}) \cap A(\mathcal{B})|}{|\,AFD(\mathcal{B})|}.$$

We can then compare AFD measures as follows. Remember from Section 3 that every AFD measure f and every threshold $\epsilon \in [0, 1]$ naturally induces a discovery algorithm A_f^ϵ which, on input relation $R(\boldsymbol{W})$, returns all FDs φ over $\boldsymbol{W}$ with $R \not\models \varphi$ and $f(\varphi, R) \in [\epsilon, 1]$. In this respect, every measure hence defines a class $DISC_f$ of discovery algorithms, namely $DISC_f = \{A_f^\epsilon \mid 0 \le \epsilon \le 1\}$. We compare the effectiveness of measures by computing the area under the precision-recall curve (AUC-PR) of $DISC_f$ for each measure f, where the PR-curve is the set $\{(rcl(A, \mathcal{B}), prec(A, \mathcal{B})) \mid A \in DISC_f\}$. It is known that PR curves are well-suited to visualize the tradeoff between precision and recall at various values of ϵ when the prediction classes are very imbalanced, which is the case here. So, the measure with the highest AUC-PR score is the measure providing the best such tradeoff.

While the PR curves provide an aggregated overview per benchmark, we also want a more fine-grained view of measures on the level of each relation $R(\boldsymbol{W})$ individually. To this end, we use the notion of *rank at max recall*, denoted $r@mr(f, R)$ which is defined as follows. Consider all possible FDs over $\boldsymbol{W}$ with a score $f < 1$ and sort them decreasingly according to f-score. Then $r@mr(f, R)$ is the smallest natural number k such that (i) the k highest-ranked FDs include all of $AFD(R)$ and (ii) the f-score of the FD at rank k is strictly larger than the f-score of the FD at rank $k + 1$. The latter implies that when an FD is among the highest-ranked FDs, all FDs with the same score should be included as well. This is to avoid that ties can be broken arbitrarily among FDs with the same score which could result in an unfair comparison between measures.

Since smoothed FI is parameterized by a parameter α it is not one measure but a collection of measures. We performed experiments with the same values of α as in the original SFI paper [36], namely $\alpha \in \{0.5, 1, 2\}$. Because the performance of $\alpha = 0.5$ consistently dominates the performance of $\alpha \in \{1, 2\}$, we only report the performance of SFI for $\alpha = 0.5$ in what follows.

We implemented all measures in a Python library. This library, together with the benchmark datasets is publicly available [34].

4.2 Benchmarks

Real world data (RWD). The RWD benchmark is used to compare measure performance on real-world data, which hence exhibits data distributions as well as data errors that occur in practice. We created the RWD benchmark as follows. We started by considering all relations mentioned in [6],[3] which collects the real-world relations most commonly used in the dependency discovery literature. This base set was extended with the relation Adult[4] used, e.g., in [12, 27, 48]. Since design schemas for these relations are unavailable, we manually created them as follows. First, in order to ensure semantically sound design schemas, we restricted our attention to the subset of relations that have a generally interpretable domain. Further, to keep the endeavor manageable, we restricted ourselves to relations that have no more than 50 columns and to linear[5] FDs. Otherwise the number of candidate FDs to inspect for inclusion in the design schema becomes prohibitive. This results in 10 relations, listed in Table 2. Next, we enumerate all candidate linear FDs (i.e., pairs (X, Y)), but only validate whether this candidate FD is semantically meaningful, and is hence part of the design schema or not, if its g_3-score is ≥ 0.5. We feel that this is a reasonable way to keep the endeavor manageable, as a g_3-score < 0.5 means that we should remove more than 50% of the tuples to obtain a subrelation that satisfies the candidate FD; making it improbable that the candidate FD should be in the design schema. Furthermore, we observe that, when we find a candidate FD semantically meaningfull, its g_3-score is always ≥ 0.99.

In summary, there are 143 design FDs across all relations in RWD, of which 126 are perfect design FDs and 17 are approximate design FDs. To appreciate the difficulty of the AFD discovery task, it is worth pointing out that the search space during AFD discovery on relation R is the total set of linear FDs that are not satisfied by R, of which there are 1519 across all relations in RWD. Out of these, only a small number (17 to be precise) are approximate design FDs, which emphasizes the intrinsic difficulty of AFD discovery and illustrates the need for good measures to distinguish AFDs from the rest of the search space.

Real world data with errors (RWD^e) To study the measures' sensitivity to different kinds and different levels of errors on real world data, we created the benchmark RWD^e. We obtain RWD^e by passing the relations $R \in$ RWD through a controlled error channel such that, denoting by R' the obtained relation, some FDs in $PFD(R)$ do not hold anymore in R' and hence become part of $AFD(R')$. Existing AFDs are always maintained, i.e., $AFD(R) \subseteq AFD(R')$.

[3] https://owncloud.hpi.de/s/j6Z0yvXC0qhtGCk/download
[4] https://archive.ics.uci.edu/ml/datasets/Adult
[5] An $FD\ \boldsymbol{X} \rightarrow \boldsymbol{Y}$ is linear when $|\boldsymbol{X}| = |\boldsymbol{Y}| = 1$.

Relation R		#rows	#attrs	#insp	#$PFD(R)$	#$AFD(R)$
R_1	adult	32561	15	111	2	0
R_2	claims	97231	13	36	2	2
R_3	dblp10k	10000	34	368	75	2
R_4	hospital	114919	15	74	22	7
R_5	tax	1000000	15	84	3	0
R_6	gath. agent	72737	18	46	5	2
R_7	gath. area	137710	11	35	3	2
R_8	gathering	90991	35	36	0	1
R_9	ident. taxon	562958	3	2	0	1
R_{10}	ident.	91799	38	69	14	0

Table 2: Overview of relations in RWD benchmark. The #insp column indicates the number of (X, Y) pairs with $g_3 \geq 0.5$ that were inspected to determine the design schema.

We actually have multiple error channels, which are parameterized by an error level $\eta \in [0, 1]$ and an error type. When passing R through the channel we consider all $\boldsymbol{X} \rightarrow \boldsymbol{Y} \in PFD(R)$ and modify $k = \lfloor \eta |R| \rfloor$ $\boldsymbol{Y}$-values. To avoid interference, we select at most one FD $\boldsymbol{X} \rightarrow \boldsymbol{Y}$ for every unique $\boldsymbol{Y}$ per relation, ensuring that $\boldsymbol{Y}$ does not appear in $AFD(R)$, and that no FD $\boldsymbol{Y} \rightarrow \boldsymbol{Z}$ has previously been selected. The procedure to modify the $\boldsymbol{Y}$ values is determined by the chosen type of data error for which we consider three categories inspired by Arocena et al. [3]: copy error, typo and bogus value. For a chosen tuple $\boldsymbol{w} \in R$, only $\boldsymbol{w}|_Y$ is changed, where the change depends on the data error type:

(i) copy: Randomly pick any $\tilde{\boldsymbol{w}} \in R$ with $\tilde{\boldsymbol{w}}|_Y \neq \boldsymbol{w}|_Y$ and make $\tilde{\boldsymbol{w}}|_Y$ the new value for $\boldsymbol{w}|_Y$.
(ii) typo: To every $\boldsymbol{y} \in dom_R(\boldsymbol{Y})$, we associate three new values representing three common typos. From these, one is chosen each time at random as the new value for $\boldsymbol{w}|_Y$.
(iii) bogus: $\boldsymbol{w}|_{\boldsymbol{y}}$ is assigned a unique newly generated value.

We point out that copy does not introduce any new values and keeps $dom_R(\boldsymbol{Y})$ stable, while typo (resp., bogus) introduces a number of new values independent of (resp., dependent on) the error level. $\boldsymbol{X}$ is not modified, and therefore $p_{R'}(\boldsymbol{X}) = p_R(\boldsymbol{X})$. To ensure that increasing error levels do not accidentally reduce errors, we ensure that, for each $\boldsymbol{x}$: $\boldsymbol{X}$ we pick at most $\lfloor N_{\boldsymbol{x}}/2 \rfloor$ tuples $\boldsymbol{w}$ with $\boldsymbol{w}|_X = \boldsymbol{x}$ to modify, where $N_{\boldsymbol{X}}$ is the number of times that $\boldsymbol{x}$ occurs in $\pi_{\boldsymbol{X}}(R)$. PFDs for which this cannot be guaranteed are omitted. The number of new AFDs that can be constructed therefore depends on the error level.

We consider four error levels: 1%, 2%, 5% and 10%. For each type of data error t and each error level η, we obtain a new benchmark RWD$^e[t, \eta]$. Consequently, we generate 12 RWD^e tables per RWD table R for which $|PFD(R)| > 0$ (so, tables R_8 and R_9 are excluded). Overall the number of AFDs increases from 17 in RWD to 39 in RWD^e[copy, 1%]. That number is the same for the other error types but can drop a little for higher noise levels as explained above. A complete overview of the number of additional AFDs per relation and per error level is given in [34]. Per combination of parameters, the ground truth then consists of the thus constructed AFDs together with the AFDs from R.

Synthetic data (SYN) Finally, we have created three synthetic benchmarks, denoted SYN^e, SYN^u, and SYN^s. We use SYN^e to provide an additional study of the measures' sensitivity to errors, now

on synthetic instead of on real-world data. We use SYN$^{\mathrm{u}}$ to study sensitivity to *left-hand-side (LHS) uniqueness*, defined as the ratio $|dom_R(\boldsymbol{X})|/|R|$, and SYN$^{\mathrm{s}}$ to study sensitivity to *right-hand-side (RHS) skew*, defined as the skewness of the distribution $p_R(\boldsymbol{Y})$. It has been observed multiple times in the literature that measures are often biased w.r.t. LHS uniqueness and RHS skew while these statistics alone do not provide a good signal for discovering $\boldsymbol{X} \to \boldsymbol{Y}$ because these statistics look only at $\boldsymbol{X}$ or only at $\boldsymbol{Y}$ but not their correlation. This is in particular true when one wants to discover non-linear FDs: as $|\boldsymbol{X}|$ increases, LHS uniqueness naturally tends to 1. Therefore, the experiments on SYN$^{\mathrm{u}}$ also give an idea of measure performance on non-linear FDs.

Each synthetic benchmark $\mathcal{B}$ consists of relations $R(\boldsymbol{XY})$ with $\boldsymbol{X} = \{X\}$ and $\boldsymbol{Y} = \{Y\}$ and is partitioned into two subsets: (1) $\mathcal{B}_{\mathrm{nfd}}$ containing relations R where $\boldsymbol{X} \to \boldsymbol{Y} \notin \Delta(R)$; and (2) $\mathcal{B}_{\mathrm{fd}}$ containing relations where $\boldsymbol{X} \to \boldsymbol{Y} \in \Delta(R)$. Each set employs a distinct random process to generate relations: for relations in $\mathcal{B}_{\mathrm{nfd}}$, $\boldsymbol{X}$ and $\boldsymbol{Y}$ values are generated independently at random, while relations in $\mathcal{B}_{\mathrm{fd}}$ are generated by first constructing a relation in which $\boldsymbol{X} \to \boldsymbol{Y}$ perfectly holds, and then passing it through an error channel as for RWD$^{\mathrm{e}}$. We are then interested in the average value of $f(\boldsymbol{X} \to \boldsymbol{Y}, R)$ within $\mathcal{B}_{\mathrm{nfd}}$ and $\mathcal{B}_{\mathrm{fd}}$, respectively. If these averages are close to each other this means that f by itself does not distinguish between random relations and relations containing an FD with errors.

We now provide more detail on the random process for generating relations. The generation process of a relation R depends on a number of parameters that are drawn uniformly at random from the following ranges: $|R| \in [100; 10000]$; $|dom_R(\boldsymbol{X})| \in [\frac{1}{5}|R|, \frac{3}{4}|R|]$, $|dom_R(\boldsymbol{Y})| \in [5, \frac{1}{2}|dom_X(R)|]$. Values for $\boldsymbol{X}$ and $\boldsymbol{Y}$ are drawn according to the Beta distribution[6], $B(\alpha, \beta)$, which is a family of continuous probability distributions defined on the interval $[0, 1]$ in terms of two positive parameters α and β that control the shape of the distribution. We consider the ranges $\alpha \in (0, 1]$ and $\beta \in [1, 10]$. For $\alpha = \beta = 1$ the distribution is uniform and for any other values it is reverse J-shaped with a right tail. The *skewness* is defined as $\frac{2(\beta-\alpha)\sqrt{\alpha+\beta+1}}{(\alpha+\beta+2)\sqrt{\alpha\beta}}$ and is known to measure the asymmetry of the probability distribution about its mean. In particular, the skew is zero for the uniform distribution and increasing values indicate longer tails with lower mass, that is, a higher mass near the left end of the interval $[0, 1]$. We sample values for α and β such that the skewness is at most one (except for SYN$^{\mathrm{s}}$ below where we consider skew values up to 10). Finally, the error rate η lies in $[0.5\%, 2\%]$.

So, for every relation R the parameters $|R|$, $|dom_R(X)|$, $|dom_R(Y)|$, $\alpha_{\boldsymbol{X}}, \beta_{\boldsymbol{X}}, \alpha_{\boldsymbol{Y}}, \beta_{\boldsymbol{Y}}, \eta$ are chosen uniformly at random under the above described conditions. To generate a table R in $\mathcal{B}_{\mathrm{nfd}}$, the following procedure is repeated $|R|$ times: sample an $x \in dom_R(\boldsymbol{X})$ (resp., $y \in dom_R(\boldsymbol{Y})$) according to $B(\alpha_{\boldsymbol{X}}, \beta_{\boldsymbol{X}})$ (resp., $B(\alpha_{\boldsymbol{Y}}, \beta_{\boldsymbol{Y}})$) and add (x, y) to R. To generate a table R in $\mathcal{B}_{\mathrm{fd}}$, we first construct a dictionary D by, for each value $x \in dom_R(\boldsymbol{X})$, assigning a value $D(x) \in dom_R(\boldsymbol{Y})$ drawn at random according to $B(\alpha_{\boldsymbol{Y}}, \beta_{\boldsymbol{Y}})$. Then, we populate R by repeatedly adding tuples $(x, D(x))$ until the required number of tuples is reached and where $x \in dom_R(X)$ is drawn at random according to $B(\alpha_{\boldsymbol{X}}, \beta_{\boldsymbol{X}})$. Finally, we add errors using the procedure described in the previous section based on η and the data error type copy.[7] We note that the generation process is related to the one from Zhang et al. [50] but with the addition of value distributions for both $\boldsymbol{X}$ and $\boldsymbol{Y}$ based on the Beta distribution.

The three synthetic benchmarks are now created by controlling one of the parameters in the parameter set. Every benchmark $\mathcal{B}$ consists of 2500 tables in $\mathcal{B}_{\mathrm{nfd}}$ and 2500 tables in $\mathcal{B}_{\mathrm{fd}}$.

SYN$^{\mathrm{e}}$. We iteratively increase the error rate η from 0% to 10% in 50 steps and generate 50 relations in SYN$^{\mathrm{e}}_{\mathrm{fd}}$ per step, varying all other parameters as described above. SYN$^{\mathrm{e}}$ is then extended with 2500 tables generated in SYN$^{\mathrm{e}}_{\mathrm{nfd}}$.

SYN$^{\mathrm{u}}$. We construct SYN$^{\mathrm{u}}$ by iteratively increasing LHS-uniqueness from $\frac{1}{5}|R|$ to $10|R|$ in 50 steps and by generating for every step 50 tables in SYN$^{\mathrm{u}}_{\mathrm{fd}}$ and SYN$^{\mathrm{u}}_{\mathrm{nfd}}$.

SYN$^{\mathrm{s}}$. We construct SYN$^{\mathrm{s}}$ by iteratively increasing RHS-skew from 0 to 10 in 50 steps and by generating for every step 50 tables in SYN$^{\mathrm{s}}_{\mathrm{fd}}$ and SYN$^{\mathrm{s}}_{\mathrm{nfd}}$.

4.3 RWD experiments

AUC. Figure 1 plots the PR curves over RWD, grouped per measure class as discussed in Section 3.5 while Table 3 lists the AUC scores. We observe that μ' (AUC = 0.95) is the most effective measure, closely followed by g'_3 (AUC = 0.90). This means that when the correct number of AFDs is not known beforehand and a specific threshold needs to be set uniformly for all relations, μ' provides the best tradeoff between precision and recall. The other measures in Logical as well as g_3 have a score around 0.65, followed by *RFI* with a score of 0.59. The other Shannon measures, g_1 and g'_1 and ρ all have a rather low AUC score ≤ 0.5, they are, hence, outperformed by all other measures. In particular, Shannon measures *FI* and *SFI* have overall the overall lowest scores.

Within Simple, we see that there is a strict order: $\rho < g_2 < g_3 < g'_3$. Within Logical, we observe that τ is almost the same as *pdep*, while μ' is a vast improvement over *pdep*. Within Shannon, we observe that the *RFI*$'$ improves upon *FI*, while *SFI* deteriorates the performance of *FI*.

Comparing variants of measures without and with baselines, we observe that g_1 and g'_1 perform equally poor (both AUC = 0.40) while g'_3 (AUC = 0.90) is considerably better than g_3 (AUC = 0.67). As a possible explanation for this behavior, we note that g'_1 determines baselines for g_1 based solely on renormalizing g_1 according to the number of distinct pairs that can be formed, which depends only on the numbers of tuples in input relation R and hence does not provide any intrinsic signal about the FD under consideration. In contrast, g'_3 determines baselines for g_3 based on renormalisation according to $|\,dom_R(\boldsymbol{X})|$, which does provide a signal of how difficult it is to falsify $\boldsymbol{X} \to \boldsymbol{Y}$ in R. We illustrate this by means of the following hypothetical example. Consider two relations S_1 and S_2 both of size 100, and FD $\varphi = \boldsymbol{X} \to \boldsymbol{Y}$ such that the largest subrelations S'_1 and S'_2 of S_1 resp. S_2 that satisfy φ are both of size 80. Then g_3 will not distinguish between φ holding in S_1 versus S_2, as in both cases $g_3 = 0.8$. Hence, for a fixed threshold ϵ, either φ is in both $A^{\epsilon}_{g_3}(S_1)$

[6] https://en.wikipedia.org/wiki/Beta_distribution

[7] For the SYN benchmark we also generated and analysed the measures on tables of type (2) with the data error types typo and bogus but as the results where similar to that of copy we restrict to the latter.

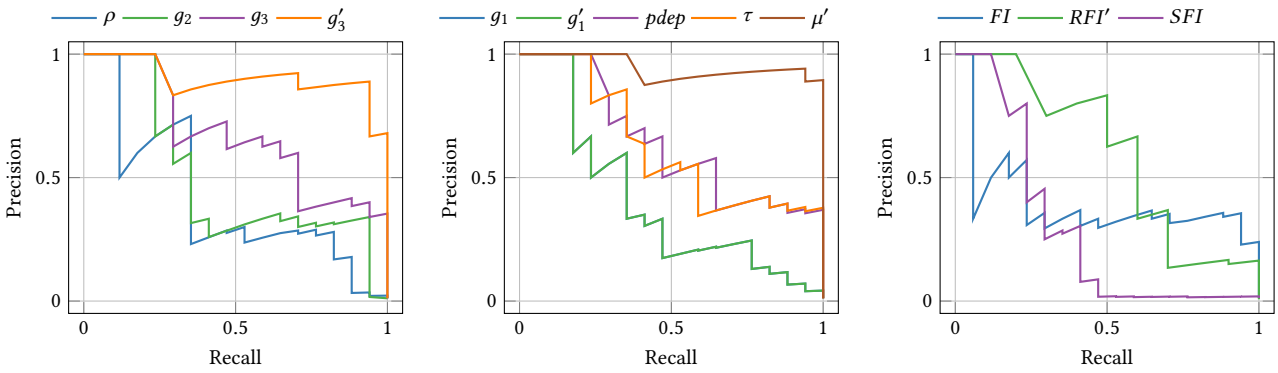

Figure 1: Precision-Recall curves over RWD grouped per measure class: Simple, Logical, and Shannon.

and $A^{\epsilon}_{g_3}(S_2)$, or in neither. Now assume that $dom_{S_1}(X)$ is of size 20 while $dom_{S_2}(X)$ is of size 80. Then g'_3 does allow to distinguish between φ in S_1 versus S_2 as it measures the size of the maximal subrelation relative to $|S_i| - |\,dom_{S_i}(X)|$, yielding $g'_3(\varphi, S_1) = 0.75$ and $g'_3(\varphi, S_2) = 0$. Clearly, in this case, S_1 had more opportunity to falsify φ, and hence provides stronger evidence that φ truly holds in S_1, while φ is trivially satisfied in S_2 and lacks such evidence.

Rank at max recall is shown in Table 4. The first row indicates the total number of design AFDs to discover. We see that when ignoring R_3, all measures except RFI and SFI, have a *r@mr* value that is very close to the number of design AFDs, meaning that at maximum recall we have very high precision. For R_3, only g'_3 and μ' have an optimal rank whereas for the other measures *r@mr* is much larger (sometimes even one or two orders of magnitude). These numbers indicate that for the measures g_3 and μ' it suffices for most relations to only examine a few top-ranked FDs.

4.4 RWD^{e} experiments

AUC. Table 3 lists AUC scores over RWD^{e} per error type and for different error levels. We observe that μ' has the highest AUC score in 10 out of 12 cases. The only two exceptions are RWD^{e}[copy, 10%] and RWD^{e}[typo, 10%] where RFI' is better. The second-best measure depends on the error type and the error level. Indeed, for RWD^{e}[copy, η], τ is second best for $\eta = 1\%, 2\%$, while *RFI'* is second best for larger values. For RWD^{e}[bogus, η], g'_3 is second best for all values of η. Finally, for RWD^{e}[typo, η], g'_3 is second best for $\eta = 1\%$, while *RFI'* is second best for values of $\eta > 1\%$.

We remark that for some measures the AUC score on RWD^{e} is larger at the 1% error level than for RWD. This is not completely unexpected as the ground truth for both is different. We do see that, as expected, the AUC score for each of the measures deteriorates at increasing error levels to an absolute low at error level 10%. The exception is *RFI'* whose performance increases at higher error levels for error types copy but not for the other error types. When error types and error levels are unknown but expected to be small, μ' therefore remains the best choice of AFD-measure. Furthermore, it is evident from Table 3 that AFD-measures are not very effective when error levels are greater than 5%.

Within Simple g'_3 remains the best measure and the observed ordering for RWD ($\rho < g_2 < g_3 < g'_3$) generally still holds. Within Logical we observe that τ is usually an improvement over *pdep*, and μ' is an improvement over τ. We see that *RFI'* improves over *FI* or is very similar for error types copy and typo, except for bogus where FI is better at lower error levels. The effectiveness of SFI remains very poor overall.

It remains to discuss the measures variants with and without baseline. Here the situation is similar to RWD: g'_3 continues to be better than g_3, albeit the improvement is not always significant, while g'_1 and g'_2 remain equally poor.

Rank at max recall. We lack the space to include the *r@mr* value for each (relation, t, η) triple. Instead, we show in Table 5 a qualitative comparison between measures by listing, for each measure f and error type t, its *winning number*, which is defined as follows. Consider a particular (relation, t, η) combination in RWD^{e}. A measure f *wins* this triple if its *r@mr* is minimal among all measures on this triple. The winning number of f for error type t is then the number of times f wins, taken over all triples of type t. Here, we see again that μ' is the best performing measure, across all error types, winning 28/32 times on copy; 21/32 times on bogus, and 19/32 times on typo. The second-best measure is τ (17/32) on copy; g'_3 (14/22) on bogus, and g_3 (15/32) on typo.

4.5 SYN experiments

As the average measure values are identical for g_1 and g'_1 we group them together in the figures.

SYN^{e}: error level. The top row of Figure 2 shows the average measure value for $\text{SYN}^{e}_{\text{fd}}$ and $\text{SYN}^{e}_{\text{nfd}}$ as a function of error rate η. Values over $\text{SYN}^{e}_{\text{fd}}$ and $\text{SYN}^{e}_{\text{nfd}}$ are presented using solid and dashed lines, respectively. For g_1 and g'_1, the solid and dashed lines coincide while for *SFI* they are very close. In particular, this means that these measures cannot distinguish between cases where X and Y are sampled independently at random and cases where data is generated according to $X \to Y$ and subsequently exposed to an error channel. For all other measures there is a clear separation even though for *pdep* and Shannon this separation is less pronounced. Furthermore, as expected, the average value over $\text{SYN}^{e}_{\text{fd}}$ decreases at increasing error levels save for g_1, g'_1 and *SFI* where it remains constant. Note that while Shannon measures other than *SFI* also decrease, this is less noticeable compared to Simple and Logical.

SYN^{u}: LHS-uniqueness. The middle row of Figure 2 shows the average measure value for $\text{SYN}^{u}_{\text{fd}}$ and $\text{SYN}^{u}_{\text{nfd}}$ as a function of LHS-uniqueness. Over $\text{SYN}^{u}_{\text{nfd}}$ we distinguish two groups: the group whose values increase with the LHS-uniqueness level (ρ, g_2, g_3,

measure	RWD	copy 1	copy 2	copy 5	copy 10	bogus 1	bogus 2	bogus 5	bogus 10	typo 1	typo 2	typo 5	typo 10
ρ	0.411	0.393	0.264	0.181	0.093	0.27	0.187	0.111	0.064	0.31	0.226	0.145	0.083
g_2	0.497	0.358	0.275	0.21	0.13	0.267	0.234	0.168	0.113	0.271	0.24	0.168	0.113
g_3	0.669	0.63	0.466	0.334	0.226	0.538	0.375	0.259	0.197	0.54	0.377	0.259	0.197
g'_3	**0.901**	0.601	0.483	0.357	0.276	**0.598**	**0.441**	**0.283**	**0.242**	**0.581**	0.458	0.285	0.242
g_1	0.401	0.367	0.316	0.251	0.177	0.328	0.298	0.218	0.151	0.323	0.299	0.221	0.152
g'_1	0.401	0.367	0.316	0.251	0.177	0.328	0.298	0.218	0.151	0.323	0.299	0.221	0.152
pdep	0.642	0.548	0.414	0.294	0.198	0.463	0.355	0.237	0.171	0.461	0.36	0.239	0.172
τ	0.623	**0.662**	**0.504**	0.342	0.216	0.503	0.385	0.253	0.19	0.506	0.392	0.255	0.189
μ'	**0.946**	**0.78**	**0.653**	**0.555**	**0.382**	**0.661**	**0.524**	**0.402**	**0.31**	**0.662**	**0.542**	**0.404**	**0.311**
FI	0.397	0.49	0.381	0.278	0.184	0.408	0.337	0.224	0.164	0.414	0.346	0.228	0.165
RFI'	0.592	0.475	0.471	**0.463**	**0.509**	0.358	0.24	0.208	0.204	0.476	**0.474**	**0.378**	**0.379**
SFI	0.287	0.221	0.218	0.2	0.208	0.078	0.074	0.068	0.074	0.157	0.153	0.144	0.156

Table 3: AUC-PR for each of the measures over RWD and RWD^e $[t, \eta]$ for $t \in \{\texttt{copy}, \texttt{bogus}, \texttt{typo}\}$ and $\eta \in \{1\%, 2\%, 5\%, 10\%\}$. Per column the two highest values are typeset in bold; the highest value is underlined as well.

Relation R	R_2	R_3	R_4	R_6	R_7	R_8	R_9
$\#AFD(R)$	2	2	7	2	2	1	1
ρ	42	27	7	2	3	1	1
g_2	98	27	7	2	3	1	1
g_3	2	8	7	3	3	1	1
g'_3	2	2	7	3	3	1	1
g_1	2	75	7	8	3	1	1
g'_1	2	75	7	8	3	1	1
pdep	2	19	7	3	3	1	1
τ	2	19	7	2	3	1	1
μ'	2	2	7	2	3	1	1
FI	2	23	7	2	3	1	1
RFI'	2	8	188	55	43	2	6
SFI	2	303	112	22	4	6	1

Table 4: *r@mr* per measure and relation in RWD. (Tables R_1, R_5, and R_{10} have no design AFDs and are omitted.)

	ρ	g_2	g_3	g'_3	g_1	g'_1	*pdep*	τ	μ'	*FI*	*RFI'*	*SFI*
copy	1	1	7	8	2	2	7	17	28	9	4	4
bogus	0	0	13	14	2	2	10	11	21	8	0	1
typo	0	1	15	13	2	2	10	10	19	9	4	4

Table 5: Winning numbers per error type on RWD^e.

pdep, and τ,) and those whose values remain constant (g'_3, *RFI'*, *SFI* and μ'). This means in particular that the first group is biased w.r.t. LHS-uniqueness: their score for $X \to Y$ increases solely on the basis of X and independent of Y and even if relations are generated by a process that sampled X and Y independently at random. For these measures it will therefore prove problematic to discover non-linear AFDs especially when the number of attributes in the LHS increases and LHS-uniqueness is expected to increase naturally to 1. The second group corrects for this behavior.

Over $\text{SYN}^{\text{u}}_{\text{fd}}$ we observe that g'_3 and μ' decrease slightly at the largest values for LHS-uniqueness which implies that they become less confident to have found an FD $X \to Y$ in a relation R when $\pi_X(R)$ contains fewer duplicates. We notice that *RFI* decreases for increasing LHS-uniqueness. The measures g_1 and g'_1 are unaffected by LHS-uniqueness.

SYN$^{\text{s}}$: RHS-skew. The bottom row of Figure 2 shows the average measure value for $\text{SYN}^{\text{s}}_{\text{fd}}$ and $\text{SYN}^{\text{s}}_{\text{nfd}}$ for different levels of RHS-skew. Over $\text{SYN}^{\text{s}}_{\text{nfd}}$, the average value of Simple and τ increases when RHS-skew increases. These measures are biased w.r.t. RHS-skew: their score for $X \to Y$ increases solely on the basis of Y and independent of X even if relations are generated by a process that sampled X and Y independently at random. We observe that Shannon, τ, and μ' correct for this behavior. Furthermore, *FI* and τ slightly decrease at higher levels of RHS-skew.

5 Related Work

In addition to the work already cited previously, the following work is related.

Correlation When an FD holds in a relation, there is clearly a statistical correlation among the FD's attributes. Conversely, correlated attributes may (but need not) indicate the presence of an FD. The techniques that are typically used to test statistical correlation, such as the χ^2 test or mean-square contingency [23], however, only measure the strength of correlation (e.g., X and Y are correlated) but do not indicate the direction in which functional dependence ($X \to Y$ or $Y \to X$) is likely to hold. As such, these techniques do not form appropriate AFD measures [38].

Exact FD discovery In the context of exact FD discovery, some works consider the problem of ranking exact FDs according to relevance, where the challenge lies in quantifying relevance [48]. In contrast, we are not concerned with exact FD discovery, but with measures for quantifying the extent to which FDs hold approximately. Discovery of AFDs should also not be confused with the approximate discovery of exact FDs as e.g., done in [7]. There, only a subset of all FDs that satisfy input R are computed in return for performance improvements.

Relaxing FDs. Caruccio et al. [10] provide a survey of the many ways in which the notion of FD can be relaxed. Broadly speaking, there are two distinct categories of relaxations: relax the constraint that an FD $X \to Y$ needs to be fully satisfied, as we do here; or replace the way in which tuples are compared on their X-values by a similarity function rather than equality, which is hence unrelated to AFD discovery, and which leads to matching dependencies [17, 42] and relaxed FDs [9]. In their survey, Caruccio et al. [10] also survey some of the AFD measures that are considered here, but do not provide a formal and qualitative comparative study.

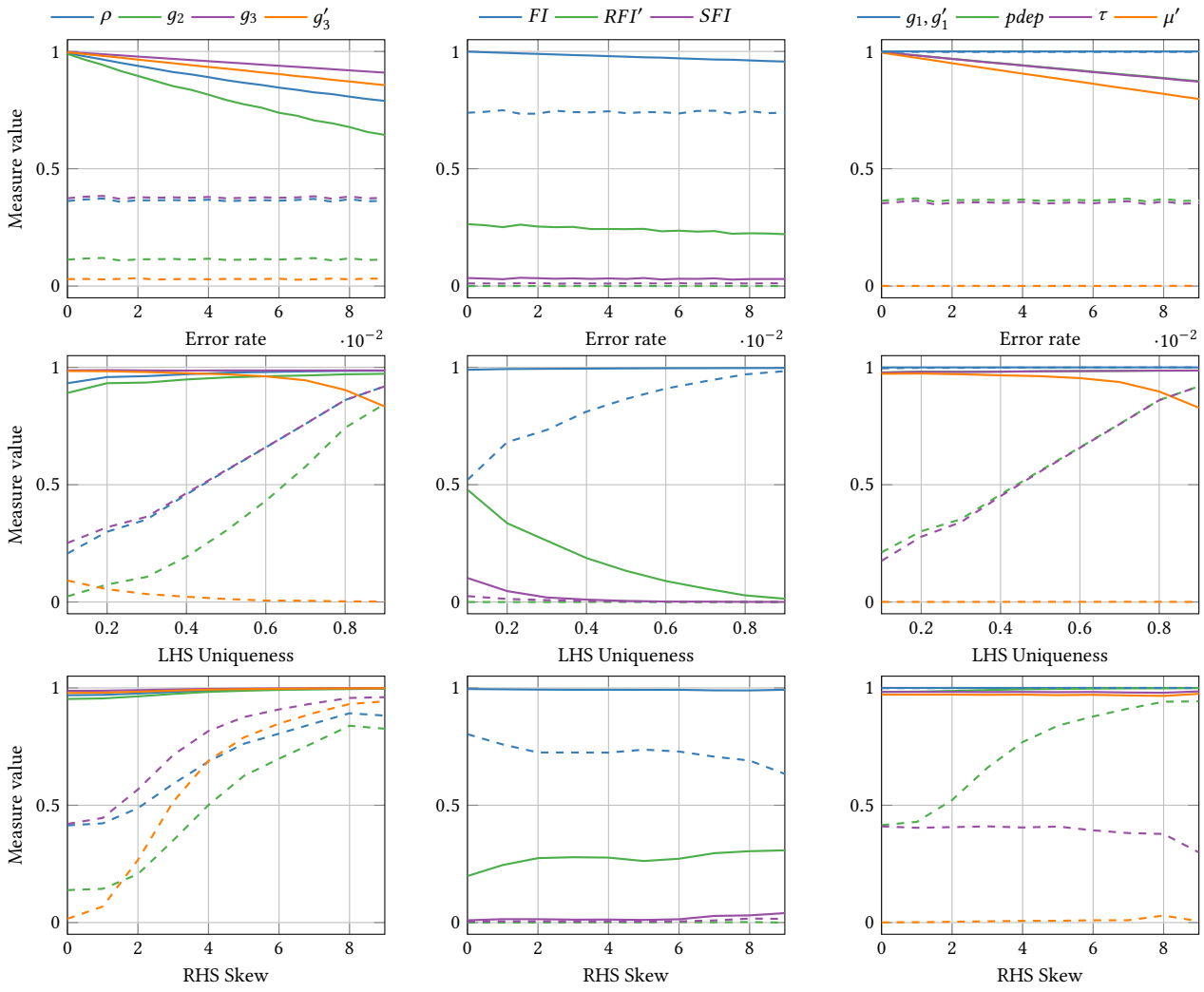

Figure 2: Average measures values of the three experiments on SYN data. The top row shows SYN^{e}_{fd} (solid lines) and SYN^{e}_{nfd} (dashed lines) for different error levels. Middle row shows SYN^{u}_{fd} (solid lines) and SYN^{u}_{nfd} (dashed lines) for different LHS-uniqueness levels. Bottom row shows SYN^{s}_{fd} (solid lines) and SYN^{s}_{nfd} (dashed lines) for different RHS-skew levels.

Comparison of measures for AFDs Giannella and Robertson [19] compare FI with g'_3 and τ on theoretical examples as well as on 4 real world datasets. In their experiments, they report on average differences pairs of measures and do not compare with a ground truth set of FDs. They therefore do not empirically compare the effectiveness of the measures considered here. UNI-DETECT [46] is a framework to automatically detect four common types of errors in relations: numeric-outliers, misspellings, uniqueness and FD violations. For FD violations, UNI-DETECT uses the FD-compliance-ratio which is similar to g_2, and it is shown that UNI-DETECT outperforms the naive algorithm ranking AFDs based on the measures ρ, g_1 and g_2. However, no comparative study of the measures for AFD discovery is provided.

Additional AFD measures. Pfahringer and Kramer propose an AFD measure based on how well the considered FD can be used to compress the input table [37]. Their approach applies only to tables in which all attributes have a binary domain, however, which is why we do not consider it here. Simovici et al. [44] study so-called impurity measures on sets and partitions and observe that each impurity measure induces an FD error measure. For an impurity measure i and threshold α, they define the *purity dependency* $X \rightarrow_{i,\alpha} Y$ to hold in R if the corresponding FD error measure value is at most α. They study relational decomposition for purity dependencies, as well as algorithms to discover the purity dependencies that hold in a relation R, given a fixed threshold α and a fixed set of impurity measures. No comparative study of impurity measures for AFD discovery is provided. Chiang and Miller [12] introduce interest metrics for conditional FDs. These are not directly relevant for AFD discovery.

References

[1] Ziawasch Abedjan, Lukasz Golab, and Felix Naumann. 2015. Profiling relational data: a survey. *VLDB J.* 24, 4 (2015), 557–581. https://doi.org/10.1007/s00778-015-0389-y

[2] Ziawasch Abedjan, Lukasz Golab, Felix Naumann, and Thorsten Papenbrock. 2018. *Data Profiling.* Morgan & Claypool Publishers. https://doi.org/10.2200/S00878ED1V01Y201810DTM052

[3] Patricia C. Arocena, Boris Glavic, Giansalvatore Mecca, Renée J. Miller, Paolo Papotti, and Donatello Santoro. 2015. Messing Up with BART: Error Generation for Evaluating Data-Cleaning Algorithms. *Proc. VLDB Endow.* 9, 2 (2015), 36–47. https://doi.org/10.14778/2850578.2850579

[4] Laure Berti-Équille, Hazar Harmouch, Felix Naumann, Noël Novelli, and Saravanan Thirumuruganathan. 2018. Discovery of Genuine Functional Dependencies from Relational Data with Missing Values. *Proc. VLDB Endow.* 11, 8 (2018), 880–892. https://doi.org/10.14778/3204028.3204032

[5] F. Berzal, J.C. Cubero, F. Cuenca, and J.M. Medina. 2002. Relational decomposition through partial functional dependencies. *Data and Knowledge Engineering* 43, 2 (2002), 207–234. https://doi.org/10.1016/S0169-023X(02)00056-3

[6] Johann Birnick, Thomas Bläsius, Tobias Friedrich, Felix Naumann, Thorsten Papenbrock, and Martin Schirneck. 2020. Hitting Set Enumeration with Partial Information for Unique Column Combination Discovery. *Proc. VLDB Endow.* 13, 11 (2020), 2270–2283.

[7] Tobias Bleifuß, Susanne Bülow, Johannes Frohnhofen, Julian Risch, Georg Wiese, Sebastian Kruse, Thorsten Papenbrock, and Felix Naumann. 2016. Approximate Discovery of Functional Dependencies for Large Datasets. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016*, Snehasis Mukhopadhyay, ChengXiang Zhai, Elisa Bertino, Fabio Crestani, Javed Mostafa, Jie Tang, Luo Si, Xiaofang Zhou, Yi Chang, Yunyao Li, and Parikshit Sondhi (Eds.). ACM, 1803–1812. https://doi.org/10.1145/2983323.2983781

[8] Toon Calders, Raymond T. Ng, and Jef Wijsen. 2002. Searching for dependencies at multiple abstraction levels. *ACM Trans. Database Syst.* 27, 3 (2002), 229–260. https://doi.org/10.1145/581751.581752

[9] Loredana Caruccio, Vincenzo Deufemia, Felix Naumann, and Giuseppe Polese. 2021. Discovering Relaxed Functional Dependencies Based on Multi-Attribute Dominance. *IEEE Trans. Knowl. Data Eng.* 33, 9 (2021), 3212–3228. https://doi.org/10.1109/TKDE.2020.2967722

[10] Loredana Caruccio, Vincenzo Deufemia, and Giuseppe Polese. 2016. Relaxed Functional Dependencies - A Survey of Approaches. *IEEE Trans. Knowl. Data Eng.* 28, 1 (2016), 147–165. https://doi.org/10.1109/TKDE.2015.2472010

[11] Roger Cavallo and Michael Pittarelli. 1987. The Theory of Probabilistic Databases. In *VLDB'87, Proceedings of 13th International Conference on Very Large Data Bases, September 1-4, 1987, Brighton, England*, Peter M. Stocker, William Kent, and Peter Hammersley (Eds.). Morgan Kaufmann, 71–81.

[12] Fei Chiang and Renée J. Miller. 2008. Discovering data quality rules. *Proc. VLDB Endow.* 1, 1 (2008), 1166–1177. https://doi.org/10.14778/1453856.1453980

[13] Xu Chu, Ihab F. Ilyas, and Paolo Papotti. 2013. Holistic data cleaning: Putting violations into context. In *29th IEEE International Conference on Data Engineering, ICDE 2013, Brisbane, Australia, April 8-12, 2013*, Christian S. Jensen, Christopher M. Jermaine, and Xiaofang Zhou (Eds.). IEEE Computer Society, 458–469. https://doi.org/10.1109/ICDE.2013.6544847

[14] Graham Cormode, Lukasz Golab, Flip Korn, Andrew McGregor, Divesh Srivastava, and Xi Zhang. 2009. Estimating the confidence of conditional functional dependencies. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2009, Providence, Rhode Island, USA, June 29 - July 2, 2009*, Ugur Çetintemel, Stanley B. Zdonik, Donald Kossmann, and Nesime Tatbul (Eds.). ACM, 469–482. https://doi.org/10.1145/1559845.1559895

[15] Sheri Edwards. 2008. Thomas M. Cover and Joy A. Thomas, Elements of Information Theory (2nd ed.), John Wiley & Sons, Inc. (2006). *Inf. Process. Manag.* 44, 1 (2008), 400–401. https://doi.org/10.1016/j.ipm.2007.02.009

[16] David Ellerman. 2021. *New Foundations for Information Theory - Logical Entropy and Shannon Entropy.* Springer.

[17] Wenfei Fan. 2008. Dependencies revisited for improving data quality. In *Proceedings of the Twenty-Seventh ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2008, June 9-11, 2008, Vancouver, BC, Canada*, Maurizio Lenzerini and Domenico Lembo (Eds.). ACM, 159–170. https://doi.org/10.1145/1376916.1376940

[18] Pierre Faure-Giovagnoli, Jean-Marc Petit, and Vasile-Marian Scuturici. 2022. Assessing the Existence of a Function in a Dataset with the g3 Indicator. In *38th IEEE International Conference on Data Engineering, ICDE 2022, Kuala Lumpur, Malaysia, May 9-12, 2022*. IEEE, 607–620. https://doi.org/10.1109/ICDE53745.2022.00050

[19] Chris Giannella and Edward L. Robertson. 2004. On approximation measures for functional dependencies. *Inf. Syst.* 29, 6 (2004), 483–507. https://doi.org/10.1016/j.is.2003.10.006

[20] Leo A. Goodman and William H. Kruskal. 1954. Measures of Association for Cross Classifications. *J. Amer. Statist. Assoc.* 49, 268 (1954), 732–764. http://www.jstor.org/stable/2281536

[21] Ykä Huhtala, Juha Kärkkäinen, Pasi Porkka, and Hannu Toivonen. 1999. TANE: An Efficient Algorithm for Discovering Functional and Approximate Dependencies. *Comput. J.* 42, 2 (1999), 100–111. https://doi.org/10.1093/comjnl/42.2.100

[22] Ihab F. Ilyas and Xu Chu. 2019. *Data Cleaning.* ACM. https://doi.org/10.1145/3310205

[23] Ihab F. Ilyas, Volker Markl, Peter J. Haas, Paul Brown, and Ashraf Aboulnaga. 2004. CORDS: Automatic Discovery of Correlations and Soft Functional Dependencies. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, Paris, France, June 13-18, 2004*, Gerhard Weikum, Arnd Christian König, and Stefan Deßloch (Eds.). ACM, 647–658. https://doi.org/10.1145/1007568.1007641

[24] Ronald S. King and James J. Legendre. 2003. Discovery of functional and approximate functional dependencies in relational databases. *Adv. Decis. Sci.* 7, 1 (2003), 49–59. https://doi.org/10.1155/S117391260300004X

[25] Jyrki Kivinen and Heikki Mannila. 1995. Approximate Inference of Functional Dependencies from Relations. *Theor. Comput. Sci.* 149, 1 (1995), 129–149. https://doi.org/10.1016/0304-3975(95)00028-U

[26] Jan Kossmann, Thorsten Papenbrock, and Felix Naumann. 2022. Data dependencies for query optimization: a survey. *VLDB J.* 31, 1 (2022), 1–22. https://doi.org/10.1007/s00778-021-00676-3

[27] Sebastian Kruse and Felix Naumann. 2018. Efficient Discovery of Approximate Dependencies. *Proc. VLDB Endow.* 11, 7 (2018), 759–772. https://doi.org/10.14778/3192965.3192968

[28] Hai Liu, Dongqing Xiao, Pankaj Didwania, and Mohamed Y. Eltabakh. 2016. Exploiting Soft and Hard Correlations in Big Data Query Optimization. *Proc. VLDB Endow.* 9, 12 (2016), 1005–1016. https://doi.org/10.14778/2994509.2994519

[29] Panagiotis Mandros, Mario Boley, and Jilles Vreeken. 2017. Discovering Reliable Approximate Functional Dependencies. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*. ACM, 355–363. https://doi.org/10.1145/3097983.3098062

[30] Panagiotis Mandros, Mario Boley, and Jilles Vreeken. 2020. Discovering dependencies with reliable mutual information. *Knowl. Inf. Syst.* 62, 11 (2020), 4223–4253. https://doi.org/10.1007/s10115-020-01494-9

[31] Fabien De Marchi, Stéphane Lopes, and Jean-Marc Petit. 2009. Unary and n-ary inclusion dependency discovery in relational databases. *J. Intell. Inf. Syst.* 32, 1 (2009), 53–73.

[32] Thorsten Papenbrock, Jens Ehrlich, Jannik Marten, Tommy Neubert, Jan-Peer Rudolph, Martin Schönberg, Jakob Zwiener, and Felix Naumann. 2015. Functional Dependency Discovery: An Experimental Evaluation of Seven Algorithms. *Proc. VLDB Endow.* 8, 10 (2015), 1082–1093. https://doi.org/10.14778/2794367.2794377

[33] Thorsten Papenbrock and Felix Naumann. 2016. A Hybrid Approach to Functional Dependency Discovery. In *Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, Fatma Özcan, Georgia Koutrika, and Sam Madden (Eds.). ACM, 821–833. https://doi.org/10.1145/2882903.2915203

[34] Marcel Parciak, Sebastiaan Weytjens, Niel Hens, Frank Neven, Liesbet Peeters, and Stijn Vansummeren. 2022. Artifacts related to "Approximately Measuring Functional Dependencies: a Comparitive Study". Available at https://github.com/MarcelPa/AFD_comparative_study.

[35] Eduardo H. M. Pena, Eduardo Cunha de Almeida, and Felix Naumann. 2019. Discovery of Approximate (and Exact) Denial Constraints. *Proc. VLDB Endow.* 13, 3 (2019), 266–278.

[36] Frédéric Pennerath, Panagiotis Mandros, and Jilles Vreeken. 2020. Discovering Approximate Functional Dependencies using Smoothed Mutual Information. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, Rajesh Gupta, Yan Liu, Jiliang Tang, and B. Aditya Prakash (Eds.). ACM, 1254–1264. https://doi.org/10.1145/3394486.3403178

[37] Bernhard Pfahringer and Stefan Kramer. 1995. Compression-Based Evaluation of Partial Determinations. In *Proceedings of the First International Conference on Knowledge Discovery and Data Mining (KDD-95), Montreal, Canada, August 20-21, 1995*, Usama M. Fayyad and Ramasamy Uthurusamy (Eds.). AAAI Press, 234–239. http://www.aaai.org/Library/KDD/1995/kdd95-027.php

[38] Gregory Piatetsky-Shapiro and Christopher J. Matheus. 1993. Measuring Data Dependencies in Large Databases. In *Proceedings of the 2nd International Conference on Knowledge Discovery in Databases*. AAAI Press, 162––173.

[39] Joeri Rammelaere and Floris Geerts. 2018. Explaining Repaired Data with CFDs. *Proc. VLDB Endow.* 11, 11 (2018), 1387–1399. https://doi.org/10.14778/3236187.3236193

[40] Theodoros Rekatsinas, Xu Chu, Ihab F. Ilyas, and Christopher Ré. 2017. HoloClean: Holistic Data Repairs with Probabilistic Inference. *Proc. VLDB Endow.* 10, 11 (2017), 1190–1201. https://doi.org/10.14778/3137628.3137631

[41] Mark S Roulston. 1999. Estimating the errors on measured entropy and mutual information. *Physica D: Nonlinear Phenomena* 125, 3 (1999), 285–294. https://doi.org/10.1016/S0167-2789(98)00269-3

[42] Philipp Schirmer, Thorsten Papenbrock, Ioannis K. Koumarelas, and Felix Naumann. 2020. Efficient Discovery of Matching Dependencies. *ACM Trans. Database*

Syst. 45, 3 (2020), 13:1–13:33. https://doi.org/10.1145/3392778
[43] Philipp Schirmer, Thorsten Papenbrock, Sebastian Kruse, Felix Naumann, Dennis Hempfing, Torben Mayer, and Daniel Neuschäfer-Rube. 2019. DynFD: Functional Dependency Discovery in Dynamic Datasets. In *Advances in Database Technology - 22nd International Conference on Extending Database Technology, EDBT 2019, Lisbon, Portugal, March 26-29, 2019*, Melanie Herschel, Helena Galhardas, Berthold Reinwald, Irini Fundulaki, Carsten Binnig, and Zoi Kaoudi (Eds.). OpenProceedings.org, 253–264. https://doi.org/10.5441/002/edbt.2019.23
[44] Dan A. Simovici, Dana Cristofor, and Laurentiu Cristofor. 2002. Impurity measures in databases. *Acta Informatica* 38, 5 (2002), 307–324. https://doi.org/10.1007/s002360100078
[45] Daisy Zhe Wang, Xin Luna Dong, Anish Das Sarma, Michael J. Franklin, and Alon Y. Halevy. 2009. Functional Dependency Generation and Applications in Pay-As-You-Go Data Integration Systems. In *12th International Workshop on the Web and Databases, WebDB 2009, Providence, Rhode Island, USA, June 28, 2009.* http://webdb09.cse.buffalo.edu/papers/Paper18/webdb09.pdf
[46] Pei Wang and Yeye He. 2019. Uni-Detect: A Unified Approach to Automated Error Detection in Tables. In *SIGMOD*. ACM, 811–828.
[47] Yihan Wang, Shaoxu Song, Lei Chen, Jeffrey Xu Yu, and Hong Cheng. 2017. Discovering Conditional Matching Rules. *ACM Trans. Knowl. Discov. Data* 11, 4 (2017), 46:1–46:38. https://doi.org/10.1145/3070647
[48] Ziheng Wei and Sebastian Link. 2019. Discovery and Ranking of Functional Dependencies. In *35th IEEE International Conference on Data Engineering, ICDE 2019, Macao, China, April 8-11, 2019*. IEEE, 1526–1537. https://doi.org/10.1109/ICDE.2019.00137
[49] Yunjia Zhang, Zhihan Guo, and Theodoros Rekatsinas. 2020. A Statistical Perspective on Discovering Functional Dependencies in Noisy Data. In *Proceedings of the 2020 International Conference on Management of Data, SIGMOD Conference 2020, online conference [Portland, OR, USA], June 14-19, 2020*, David Maier, Rachel Pottinger, AnHai Doan, Wang-Chiew Tan, Abdussalam Alawini, and Hung Q. Ngo (Eds.). ACM, 861–876. https://doi.org/10.1145/3318464.3389749
[50] 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*. ACM. https://doi.org/10.1145/3318464.3389749

A Proof of well-definedness of μ

In this section we prove the following result mentioned in Footnote 6.

Lemma A.1. *If* $\mathbb{E}_R[pdep(\varphi, R)] = 1$ *then* $R \models \varphi$.

Proof. Assume that $\mathbb{E}_R[pdep(\varphi, R)] = 1$. Let $R_1, \ldots, R_n$ be an enumeration of all permuations of R. Then

$$\mathbb{E}_R[pdep(\varphi, R)] = \frac{\sum_{i=1}^{N} pdep(\varphi, R_i)}{N}$$

Hence, $\mathbb{E}_R[pdep(\varphi, R)] = 1$ iff $\sum_{i=1}^{N} pdep(\varphi, R_i) = N$. Because the range of $pdep$ is the interval $[0, 1]$ this sum can equal N if, and only if, $pdep(\varphi, R_i) = 1$ for every R_i, including R itself. Suppose, for the purpose of contradiction, that $R \not\models \varphi$. Then, the value of $pdep$ is given by the formula in Section 3.4, i.e.

$$\begin{aligned} pdep(X \to Y, R) &= \sum_{\boldsymbol{x}} p_R(\boldsymbol{x})[1 - pdep(Y \mid \boldsymbol{x}, R) \\ &= 1 - \mathbb{E}_{\boldsymbol{x}}[h_R(Y \mid \boldsymbol{x})] \end{aligned}$$

where the second equality is due to Lemma B.2. Since $pdep(X \to Y, R) = 1$, this means in particular that $\mathbb{E}_{\boldsymbol{x}}[h_R(Y \mid \boldsymbol{x})] = 0$, which by reasoning similar as above can only happen if $h_R(Y \mid \boldsymbol{x}) = 0$ for every $\boldsymbol{x} \in \pi_{\boldsymbol{X}}(R)$. This means, that for every $\boldsymbol{x} \in \pi_{\boldsymbol{X}}(R)$, the probability to draw two distinct Y-tuples in $\pi_Y(\sigma_{\boldsymbol{X}=\boldsymbol{x}}(R))$ is zero. But that can only happen if there is only one Y-value $\pi_Y(\sigma_{\boldsymbol{X}=\boldsymbol{x}}(R))$, in which case $R \models \varphi$ and we obtain our desired contradiction. □

B Proofs of equivalence

In this section we prove that the alternate measure formulations shown in Table 1 are correct, hence proving Theorem 3.3. The theorem is proved as a sequence of lemmas. Throughout this section, assume that $R \not\models X \to Y$.

Lemma B.1. $g_3(X \to Y, R) = \sum_{\boldsymbol{x}} p_R(\boldsymbol{x}) \max_{\boldsymbol{y}} p_R(\boldsymbol{y} \mid \boldsymbol{x})$.

Proof. We reason as follows.

$$\begin{aligned} g_3(X \to Y, R) &= \max_{R' \in G_3(X \to Y, R)} \frac{|R'|}{|R|} \\ &= \max_{R' \in G_3(X \to Y, R)} \sum_{\boldsymbol{w} \in R'} p_R(\boldsymbol{w}) \\ &= \sum_{\boldsymbol{x}} \max_{\boldsymbol{y}} p_R(\boldsymbol{xy}) \\ &= \sum_{\boldsymbol{x}} p_R(\boldsymbol{x}) \max_{\boldsymbol{y}} p_R(\boldsymbol{y} \mid \boldsymbol{x}). \end{aligned}$$

Here, the first equality is the definition of g_3. The second equality follows by definition of p_R. The third equality follows from the following observation: a relation $R' \subseteq R$ can only be maximal if $R'(\boldsymbol{w}) = R(\boldsymbol{w})$ whenever $R'(\boldsymbol{w}) > 0$ for all $\boldsymbol{w} \in R$. That is, either we keep all occurrences of $\boldsymbol{w}$ or we remove all of them. So, maximizing $\sum_{\boldsymbol{w} \in R'} p_R(\boldsymbol{w})$ corresponds to, for every $\boldsymbol{x}$, keeping that $\boldsymbol{y}$ that maximizes $p_R(\boldsymbol{xy})$. Thereby, effectively removing all other tuples $\boldsymbol{xy}'$ with $\boldsymbol{y} \neq \boldsymbol{y}'$. The last equality then follows from the definition of conditional probability. □

Lemma B.2. $pdep(Y \mid \boldsymbol{x}, R) = 1 - h_R(Y \mid \boldsymbol{x})$ *and therefore*

$$\begin{aligned} pdep(X \to Y, R) &= \sum_{\boldsymbol{x}} p_R(\boldsymbol{x})(1 - h_R(Y \mid x)) \\ &= 1 - \sum_{\boldsymbol{x}} p_R(\boldsymbol{x}) h_R(Y \mid \boldsymbol{x}) \\ &= 1 - \mathbb{E}_{\boldsymbol{x}}[h_R(Y \mid \boldsymbol{x})] \end{aligned}$$

Proof. We first observe

$$\begin{aligned} pdep(Y \mid \boldsymbol{x}, R) &= \sum_{\boldsymbol{y}} p_R(\boldsymbol{y} \mid \boldsymbol{x})^2 \\ &= 1 - (1 - \sum_{\boldsymbol{y}} p_R(\boldsymbol{y} \mid \boldsymbol{x})^2) \\ &= 1 - h_R(Y \mid \boldsymbol{x}). \end{aligned}$$

Hence,

$$\begin{aligned} pdep(X \to Y, R) &= \sum_{\boldsymbol{x}} p_R(\boldsymbol{x})\, pdep(Y \mid \boldsymbol{x}, R) \\ &= \sum_{\boldsymbol{x}} p_R(\boldsymbol{x})(1 - h_R(Y \mid x)) \\ &= \sum_{\boldsymbol{x}} p_R(\boldsymbol{x}) - \sum_{\boldsymbol{x}} p_R(\boldsymbol{x}) h_R(Y \mid \boldsymbol{x}) \\ &= 1 - \sum_{\boldsymbol{x}} p_R(\boldsymbol{x}) h_R(Y \mid \boldsymbol{x}). \end{aligned}$$

□

Lemma B.3. $\tau(X \to Y, R) = 1 - \frac{\mathbb{E}_{\boldsymbol{x}}[h_R(Y|\boldsymbol{x})]}{h_R(Y)}$

Proof. We reason as follows.

$$\begin{aligned}\tau(X \to Y, R) &= \frac{pdep(\boldsymbol{X} \to \boldsymbol{Y}, R) - pdep(\boldsymbol{Y}, R)}{1 - pdep(\boldsymbol{Y}, R)} \\ &= \frac{(1 - \mathbb{E}_{\boldsymbol{x}}[h_R(\boldsymbol{Y} \mid \boldsymbol{x})]) - (1 - h_R(\boldsymbol{Y}))}{1 - (1 - h_R(\boldsymbol{Y}))} \\ &= \frac{h_R(\boldsymbol{Y}) - \mathbb{E}_{\boldsymbol{x}}[h_R(\boldsymbol{Y} \mid \boldsymbol{x})]}{h_R(\boldsymbol{Y})} \\ &= 1 - \frac{\mathbb{E}_{\boldsymbol{x}}[h_R(\boldsymbol{Y} \mid \boldsymbol{x})]}{h_R(\boldsymbol{Y})}\end{aligned}$$

□

Here, the second equality is by Lemma B.2 and the fact that $pdep(\boldsymbol{Y}, R) = 1 - h_R(\boldsymbol{Y})$ by definition.

Relating this to logical entropy we observe

Lemma B.4. $\mu(\boldsymbol{X} \to \boldsymbol{Y}, R) = 1 - \frac{\mathbb{E}_{\boldsymbol{x}}[h_R(\boldsymbol{Y}|\boldsymbol{x})]}{h_R(\boldsymbol{Y})} \frac{|R|-1}{|R|-|\,dom(\boldsymbol{X},R)|}$

Proof. We reason as follows.

$$\begin{aligned}\mu(\boldsymbol{X} \to \boldsymbol{Y}, R) &:= \frac{pdep(\boldsymbol{X} \to \boldsymbol{Y}, R) - \mathbb{E}_R[pdep(\boldsymbol{X} \to \boldsymbol{Y}, R)]}{1 - \mathbb{E}_R[pdep(\boldsymbol{X} \to \boldsymbol{Y}, R)]} \\ &= 1 - \frac{1 - pdep(\boldsymbol{X} \to \boldsymbol{Y}, R)}{1 - pdep(\boldsymbol{Y}, R)} \frac{|R| - 1}{|R| - |\,dom(\boldsymbol{X}, R)|} \\ &= 1 - \frac{1 - (1 - \mathbb{E}_{\boldsymbol{x}}[h_R(\boldsymbol{Y} \mid \boldsymbol{x})])}{1 - (1 - h_R(\boldsymbol{Y}))} \frac{|R| - 1}{|R| - |\,dom(\boldsymbol{X}, R)|} \\ &= 1 - \frac{\mathbb{E}_{\boldsymbol{x}}[h_R(\boldsymbol{Y} \mid \boldsymbol{x})]}{h_R(\boldsymbol{Y})} \frac{|R| - 1}{|R| - |\,dom(\boldsymbol{X}, R)|}.\end{aligned}$$

□

Lemma B.5.

$$FI(\boldsymbol{X} \to \boldsymbol{Y}, R) = 1 - \frac{H_R(\boldsymbol{Y} \mid \boldsymbol{X})}{H_R(\boldsymbol{Y})}.$$

Proof. To show the claimed equality, we reason as follows. Recall that we implicitly assume throughout the paper that R is non-empty. By definition

$$FI(\boldsymbol{X} \to \boldsymbol{Y}, R) := \begin{cases} 1 & \text{if } |\,dom_R(\boldsymbol{Y})| = 1, \\ \frac{H_R(\boldsymbol{Y}) - H_R(\boldsymbol{Y}|\boldsymbol{X})}{H_R(\boldsymbol{Y})} & \text{otherwise.} \end{cases}$$

We now make a case analysis.

- If $|\,dom_R(\boldsymbol{Y})| = 1$ then $H_R(\boldsymbol{Y}) = 0$. Moreover, if $H_R(\boldsymbol{Y}) = 0$, also $H_R(\boldsymbol{Y} \mid \boldsymbol{X}) = 0$. As such,

  $$1 - \frac{H_R(\boldsymbol{Y} \mid \boldsymbol{X})}{H_R(\boldsymbol{X})} = 1 - \frac{0}{0} = 1 - 0 = 1 = FI(\boldsymbol{X} \to \boldsymbol{Y}, R),$$

  as desired.
- If $|\,dom_R(\boldsymbol{Y})| > 1$ then

  $$\begin{aligned}FI(\boldsymbol{X} \to \boldsymbol{Y}, R) &= \frac{H_R(\boldsymbol{Y}) - H_R(\boldsymbol{Y} \mid \boldsymbol{X})}{H_R(\boldsymbol{Y})} \\ &= 1 - \frac{H_R(\boldsymbol{Y} \mid \boldsymbol{X})}{H(\boldsymbol{Y})}\end{aligned}$$

  □