Empirical Glitch Explanations

Tamraparni Dasu
AT&T Labs - Research
tamr@research.att.com

Ji Meng Loh
New Jersey Institute of Technology
ji.m.loh@njit.edu

Divesh Srivastava
AT&T Labs - Research
divesh@research.att.com

ABSTRACT

Data glitches are unusual observations that do not conform to data quality expectations, be they logical, semantic or statistical. By applying data integrity constraints, potentially large sections of data could be flagged as being noncompliant. Ignoring or repairing significant sections of the data could fundamentally bias the results and conclusions drawn from analyses. In the context of Big Data where large numbers and volumes of feeds from disparate sources are integrated, it is likely that significant portions of seemingly noncompliant data are actually legitimate usable data.

In this paper, we introduce the notion of *Empirical Glitch Explanations* – concise, multi-dimensional descriptions of subsets of potentially dirty data – and propose a scalable method for empirically generating such explanatory characterizations. The explanations could serve two valuable functions: (1) Provide a way of identifying legitimate data and releasing it back into the pool of clean data. In doing so, we reduce cleaning-related *statistical distortion* of the data; (2) Used to refine existing data quality constraints and *generate and formalize domain knowledge.*

We conduct experiments using real and simulated data to demonstrate the scalability of our method and the robustness of explanations. In addition, we use two real world examples to demonstrate the utility of the explanations where we reclaim over 99% of the suspicious data, keeping data repair related statistical distortion close to 0.

1. INTRODUCTION

While much attention has been paid to identifying data quality constraint violations and developing cleaning strategies, there has not been much focus on whether all data that is noncompliant should be subject to repair, and if all data that violate a given constraint should be treated as a homogeneous set. By unnecessarily or incorrectly remediating noncompliant data, there is a danger of changing the data to such an extent that it is unrecognizable and suffers a high *statistical distortion* as defined in [6]. Conclusions and inferences drawn from over-treated and distorted data could likely be misleading.

Empl.	Status	Phone	Dept.	Room	Super.
ID_1	Active	1AAA3600000	D4000	—	ID_4
ID_2	—	1AAA3600000	—	—	—
ID_3	Retired	1AAA3600000	D2200	E260	ID_6
ID_5	Active	1AAA3608776	D2300	A115	ID_9
ID_7	New Hire	1AAA3608776	D2300	D284	ID_5
ID_8	New Hire	1AAA3608776	D2300	B106	ID_5
ID_10	Active	1AAA3605519	D8000	A132	ID_13
ID_11	Active	1AAA3605519	D8000	A132	ID_13
ID_12	Active	1AAA3605519	D8000	A132	ID_13

Table 1: *Sample data from a Human Resources database: Employee ID, Employee Status, Phone number, Department ID, Room Number, Supervisor ID. Three sets of duplicates corresponding to three different phone numbers violate the data quality constraint "Any given phone number must have only one record associated with it."*

Given that data quality constraints tend to be fairly broad and flag significantly large tracts of data as suspect, it is critical to study this data for additional, potentially explanatory relationships in the data that could reduce the cost and distortion associated with cleaning, as well as add to our domain knowledge of the data. Data quality is so highly context and domain dependent that any empirical method that facilitates the gathering of domain knowledge, particularly in Big Data scenarios, is valuable in itself.

In this paper, we provide evidence that significant portions of data that seem to violate constraints have valid explanations and can be released back into the clean pool of data without being altered. Identifying empirical explanations for seemingly suspicious data based on attribute patterns is a valuable contribution to the data quality process that preserves the original characteristics of the data, and to the best of our knowledge, has not been addressed before. Formalizing and generating domain knowledge, or suggesting repair strategies are outside the current scope of the paper and will be addressed in future research.

1.1 An Illustrative Example

For illustrative purposes, we focus on a small instance from the Human Resources (HR) database of a big corporation. We will explore the example in detail in Section 5.

In Table 1, we present nine records, each with six attributes – Employee ID, Employee Status, Phone Number, Department ID, Room Number, Supervisor ID. In principle,

KDD'14, August 24–27, 2014, New York, NY, USA.
Copyright 2014 ACM 978-1-4503-2956-9/14/08 ...$15.00.
http://dx.doi.org/10.1145/2623330.2623716.

a phone number is supposed to be unique and hence the attribute should follow the constraint: "Any given phone number must have only one record associated with it." However, we found several duplicates. We discuss three instances here, where each of the three phone numbers occurs three times. We have changed the actual values for proprietary reasons while preserving the attribute relationships.

In the first set of 3 duplicates corresponding to phone number 1AAA3600000, there are five missing values. In the second set corresponding to phone number 1AAA3608776, the employees are from the same department. Furthermore, employee with ID_5 is the supervisor of the other two employees, both new hires. Finally, in the third set corresponding to phone number 1AAA3605519, the three employees are again from the same department. In addition, they are all in the same room A132 and report to the same supervisor ID_13.

Note that even though each of the examples violates the same constraint, namely "a given phone number must have only one record associated with it", it is possible that the explanation could be different. The first set could represent genuinely bad data or even a default value, since the set contains other bad data (missing values). The second set could reflect the legitimate use of the supervisor's phone number for new hires. Finally the third set could reflect employees sitting in the same room A132 and hence sharing a physical phone. If these explanations are consistent with real world experience, we can return the second and third sets (6 records = 67% of bad data) to the "clean" data pool and modify the "no duplicates for a given phone number" rule to include the two exceptions – namely, if the employees are new hires and have the same phone number as their supervisor, or if they sit in the same physical room.

Our goal in this paper is to empirically discover such multi-attribute explanations for data quality violations. In doing so, we enable data consumers to sift through data glitches and identify positives that can be returned to the clean data pool, at the same time refining data quality constraints to better reflect the changing nature of the data. This is particularly important in the context of Big Data analytics where not just the data, but also the rules that govern them are in a state of flux, and where automation and speed are of essence.

1.2 Related Work

Data quality is an active area of research with extensive literature that covers a vast spectrum of topics. We briefly mention a small subset here, and refer the reader to literature that takes a broader overview of data quality such as the introductory [5] which focuses on an exploratory and analytical approach, or the more recent [8] which provides an overview of recent advances in the theory and application of data quality, including data inconsistencies, data deduplication, characterizing incomplete data, and data currency models; and applications in automatically discovering data quality rules, detecting errors in real-life data, and for correcting errors with performance guarantees.

There has also been considerable interest in refining data quality constraints. In [9], the authors focus on identifying subsets of data that do not conform to consistency constraints that are specified as functional dependencies. They propose a method for automatically generating "tableaux" that either violate or satisfy a given constraint.

In [3], the authors propose a new data-driven tool that focuses on the discovery of context-dependent rules and conditional functional dependencies (CFDs) that almost hold. The tool returns the rules together with the noncompliant records. In subsequent work [4], the authors propose that in contemporary scenarios, the constraints evolve constantly as the underlying data processes change. They describe a framework where the data and the constraints are modified in conjunction to minimize the cost of repair.

Other work has focused on glitch patterns and correlations [1] and introduced the idea of multi-type and multidimensional glitches. The authors use glitch dependencies and patterns for identifying data-driven cleaning strategies. In [2], the authors propose a *masking index* to estimate the impact of glitches hidden by masking (e.g. missing data mask duplicates). The idea of *statistical distortion*, the distortion in data caused by well-intentioned data repair efforts was introduced as a critical criterion for measuring the utility of data cleaning strategies in [6].

Our work is fundamentally different and novel and goes beyond validating or modifying constraints. We aim to *empirically explain* the violations in order to reduce the amount of data to be be cleaned and modified. In fact, as we demonstrated in our illustrative example, the same constraint ("duplicate phone numbers") could generate different explanations. Using the same repair for all data that violate this constraint could introduce new data glitches where none existed. This is an important contribution because existing literature makes no further distinction once the set of data that violates a constraint has been identified. We do not merely verify, validate or modify existing constraints, we explain the constraint violations to redeem good data, and lay the groundwork for the refinement of existing constraints and automatic generation of new constraints.

1.3 Our Contributions

In this paper, we turn our attention to the fundamental task of explaining seemingly anomalous data by empirically discovering patterns, and characterizing subsets that can be returned to the clean data pool, thus reducing the statistical distortion induced by unnecessary repairs. We:
(1) Introduce the novel and important notion of *explainable glitches* which are seeming violations that can be collectively described by a succinct empirical description. Such descriptions have the potential to explain the glitches, either by consulting subject matter experts ("supervisor's phone number used initially for new hires") or other heuristics ("shared physical device in a shared room"). The explanations could serve two valuable functions:
- (a) Provide a way to identify legitimate data and release it back into the pool of clean data. In doing so, we reduce *statistical distortion* of the data due to misguided data repair;
- (b) Used to refine existing data quality constraints and *generate and formalize domain knowledge.*
(2) Propose a *robust and scalable* method for empirically generating the explanations by developing the new notion of *crossover subsampling* to create subsets that are similar to the noncompliant set. In doing so we reduce the redundancy of the resampling procedure caused by the disparity in sizes between dataset D and the suspicious subset A and ensure that our results are statistically significant. In addition, we define *two objective metrics, size and merit,* for evaluating and ranking the explanations. The metrics make

the method flexible and customizable depending on the application. Such flexibility is key to a highly domain dependent task like data cleaning.
(3) We evaluate the methodology within a comprehensive experimental framework using real and synthetic data sets, and explore the robustness and scalability of explanations. In one real data instance, we are able to reclaim 99% of the data flagged as suspicious, reducing the potential statistical distortion considerably.

In this paper we focus on the notion that data that violate constraints can be explained and reclaimed for normal use without any alteration, thus preserving the authenticity of the original data. Generating and formalizing domain knowledge is outside the current scope and will be addressed in future work.

1.4 Paper Organization

The rest of the paper is organized as follows. In Section 2, we introduce the problem and in Section 3, present our approach to solving it. We discuss our empirical framework in Section 4. We present two real world case studies in Section 5 and Section 6. Finally, we summarize our results and identify future research in Section 7.

2. PROBLEM DESCRIPTION

Suppose that we are given a data set D with N rows (records) and d columns (attributes), and a constraint C. Constraints are rules (logical, semantic, statistical) that are imposed on data, typically to ensure conformity to expectations about the data e.g. “social security numbers must have 9 digits”. Let the set A consist of all “suspicious” records in D that violate C. In our illustrative example, D would be the HR data, C would be the constraint “any given phone number must have only one record associated with it” and A would be the set of nine records in Table 1. In the absence of explanations, the problematic set Q that needs to be cleaned is given by

$$Q = A.$$

Our objective is to reduce the size of set Q by identifying portions of the set A that can be explained as “clean” using characteristics derived from other attributes and data values. By doing so, we cut the cost of cleaning *and* reduce distorting the statistical properties of the original data. A cleaning process typically makes an educated guess about the correct values. By adopting a frugal cleaning approach, we preserve more of the original data and stay faithful to the original statistical properties of the data.

Briefly, we achieve our objective by generating empirical explanations $\mathcal{E}$, each of which describes a set of records $P \subseteq A$. Explanations are typically of the form $\{s_j\}$, where s_j describes a condition on a value v_j in the suspicious set A.

For example, from the illustrative example in Section 1.1, for the suspicious sets corresponding to the phone numbers in parentheses, the following explanations were generated (we drop the attribute name when there is no ambiguity):
A (`1AAA3600000`) : $\mathcal{E}_1 = \{$“blank” is frequent and occurs in multiple attributes$\}$
A (`1AAA3608776`) : $\mathcal{E}_2 = \{$ `ID_5 in attributes 1 and 6, New Hire, D2300` $\}$
A (`1AAA3608519`) : $\mathcal{E}_3 = \{$ `ID_13, A132, D8000` $\}$
The empirical descriptions were then presented to an expert, who provided a real world description:
$\mathcal{E}_1 = \{$“blank” is frequent and occurs in multiple attributes $\} \rightarrow$ “Bad data, needs remediation.”
$\mathcal{E}_2 = \{$ `ID_5 in attributes 1 and 6, New Hire, D2300` $\}$ $\rightarrow$ “Clean data : New hires assigned supervisor’s phone #.”
$\mathcal{E}_3 = \{$ `ID_13, A132, D8000` $\} \rightarrow$ “Clean data : Members of same department working for the same supervisor, sharing a physical room, and a phone.”
Therefore, in our example, of the three suspicious sets, only the one associated with `1AAA3600000` was truly problematic.

3. OUR APPROACH

We take a nonparametric approach to the problem. By doing so, we ensure a general applicability that is agnostic to any underlying data distributions. The main steps are:
(1) Identify the set A by applying the constraints C to the data set D as shown in Figure 1. In the absence of further explanation, the entire set A is deemed *suspicious.* We avoid the word anomalous since our objective is to establish that not all of A is anomalous.
(2) For each value $v \in A$, generate a *propensity signature, s*. The signature is probabilistic and captures the propensity of occurrence of a value v across all records and attributes of A, as shown in Figure 1.
(3) Rank the signatures based on their *suspiciousness*, using statistical criteria. The significant signatures together constitute an explanation

$$\mathcal{E} = \{s_j\}.$$

The signatures can be used collectively in a conjunctive (conditioned upon multiple attributes), disjunctive (conditioned upon multiple values of same attribute), or in some other manner to define the explanation.
(4) Apply the explanation $\mathcal{E}$ to A, to isolate the corresponding set of records P of A.
(5) Quantify the *effectiveness* of an explanation using its *size* and *merit* in reducing the statistical distortion of impacted records.

3.1 Suspicious Set

Given a data quality constraint C, we apply the constraint to the entire data set D and identify A, the suspicious subset of data violations. Identifying A is relatively easy for obvious glitches like null missing values or exact duplicates. However, it is non-trivial in more complex cases such as disguised missing values [12] and where the glitches are masked or hidden [2]. In addition, if the glitch detection is dependent on thresholds, for example in outliers, then determining A is even more task dependent. However, methods for formulating C and determining A are outside the scope of this paper. We assume that the data quality constraint C and the resultant set of violations A are both clearly specified.

Definition 3.1. *The set of records $A \subseteq D$ that violate the data quality constraint C constitute the suspicious set.*

Usually $|A| << |D|$, but as we will see in the case studies, there could be exceptions. Let the “good” or non-suspicious data be given by

$$A' = D - A,$$

the complement of A with respect to the entire data set D. Our objective is to identify values $v \in A$ that exhibit different statistical behavior in A and A'.

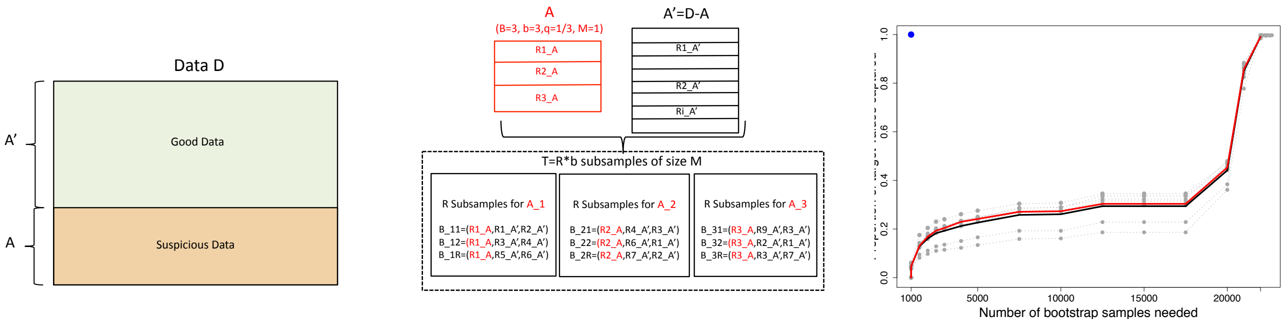

Figure 1: (a) Given a data set D, and a subset A of potentially suspicious data, we can compute *propensity signatures* for every value v in A by estimating its propensity of occurrence across the attributes $\{C_k\}$. (b) Crossover Subsampling: The suspicious set A of size $B = 3$ is divided into $b = 3$ parts of crossover proportion $q = 1/3$. Each part A_i is equivalent to a record R_i_A by construction. For each part R_i_A of size $M = B/b = 1$, the remaining $B - M = 2$ records R_j_D are drawn randomly from $D - A_i$ and a subsample of size $B = 3$ is created. This is repeated $R = 3$ times for each A_i, yielding $R * b = 3 * 3 = 9$ crossover subsamples in all. (c) Crossover sampling (blue dot), due to its design, captures all the values in a "suspicious set" in a specified number of samples (1000) while simple random sampling will require more than 20,000 samples to capture half the suspicious set, as shown by 10 iterations (grey dots), and mean (red) and median (black) over the 10 iterations.

3.2 Propensity Signatures

In order to capture the behavior of a value v in a set, we propose propensity signatures. Let v be a value in A, the suspicious set. Further, let the total number of attributes in A be d, and let p_k be the probability of v occurring in attribute C_k of A, where $k = 1, \ldots, d$. Then,

Definition 3.2. *The propensity signature of a value v in set A is a d-dimensional vector given by*

$$s_A(v) = (p_1, \ldots, p_k, \ldots, p_d), k = 1, \ldots, d,$$

and captures the propensity of occurrence of v in A.

Analogously, the signature of v in A' is given by:

$$s_{A'}(v) = (P_1, \ldots, P_k, \ldots, P_d), k = 1, \ldots, d,$$

where P_k is the probability of occurrence of v across A'.

Note that propensity signatures differ from the traditional joint density functions which focus on the joint occurrence of different values in a record, while propensity signatures focus on the occurrence of a value across all records and attributes in a sample.

Since we do not know the distributions of v *a priori*, we will use the empirical estimates of propensity signatures $\widehat{s(v)}$ to identify the set of *suspicious* values $\mathbf{V} = \{v\}$ that have statistically different signatures in the suspicious data set A compared to the "good" data A'.

For example, given the suspicious set A of duplicate phone numbers corresponding to `1AAA3608776` discussed in Section 1.1, the *estimated propensity signatures* are:

$$s_A\widehat{(ID_5)} = (1/3, 0, 0, 0, 0, 2/3)$$

and

$$s_A(\widehat{NewHire}) = (0, 2/3, 0, 0, 0, 0).$$

We describe the empirical estimates in detail in Section 4.2. We wish to demonstrate that not all suspicious values are necessarily "dirty", and can be reclaimed without cleaning or altering in any way.

3.3 Statistical Significance

How do we determine whether the propensity signature of a value is statistically significant? One approach would be to compute the distances of propensity signatures of all values in A from the corresponding value signatures in the good set A', and rank the values based on the signature distances. The values with the greatest signature distances (say top 10%) could be considered statistically different. However, the problem with this approach is that the signatures of different values are not comparable, nor the distances between them. The signature of a common numeric value like 0 that spans multiple attributes and serves both as a real value as well as a default, is bound to be different from that of a specific character string like "Florida", for example. The relative ranking of signatures and distances of different values might be distorted by inherent differences in the way the values are used.

3.3.1 Crossover Subsampling

An alternate approach is to use resampling, where we draw samples repeatedly, compute propensity signatures of a given value in each sample, and construct a *sampling distribution* of the propensity signatures. From the sampling distribution, we can infer the expected signature of the value, as well as the expected variability in its signature. Resampling is an established technique for capturing the variability of statistical and empirical estimates, see [7, 11]. Since all the signatures in the sampling distribution pertain to the *same* value, the question of comparability does not arise.

In addition to the comparability of propensity signatures, we need to ensure that the signatures are computed from like sized data sets. The size of the set influences the variability of statistical estimates, and when we compare the propensity signature of a value from the suspicious set A, it is important to draw sample sets of similar size. This is achieved through *subsampling*, i.e. choosing a sample of a smaller size from a bigger sample, and in our case, choosing one the same size as A from the good set $A' = D - A$.

However, A is significantly smaller than D, and therefore A', because we expect the suspicious set of data quality violations to be fairly small. This makes random subsampling unsuitable for our purposes where it is likely that many of the subsamples drawn from A' will not capture the values in the records of A, and therefore make no contribution to the sampling distribution estimation. We would like to construct specialized subsamples that share some characteristics of A, in addition to being like-sized.

We accomplish this by proposing a novel subsampling technique called *crossover subsampling.* Note that crossover subsampling described below is different from stratified subsampling, where the subsample is drawn randomly and proportionally from each of the classes of interest, for example, A and $D - A$. However, with crossover subsampling, we are guaranteed that *every* record in A is represented in a specified proportion of the subsamples. Figure 1(c) empirically shows that over 10 iterations, given a target "suspicious set", crossover sampling (blue dot), due to its design, captures all the target values in a specified number of samples (e.g. 1000), while simple random sampling will require more than 20,000 samples to capture half the target values in any of the 10 iterations (grey dots), or the mean (red) and median (black) computed over the 10 iterations.

Definition 3.3. *A q-crossover subsample of size B drawn from two sets D and $A \subset D$ where the size of set A is $|A| = B$, is defined to be a set that contains q proportion of samples from A, and the rest from $D - A$, and every record in A occurs in exactly q proportion of the subsamples.*

A q-crossover subsample is constructed as follows. In the absence of any prior knowledge, we partition A (size B) into $b = 1/q$ chunks of size $M = B/b$, denoted by $A = A_1 + A_2 + \ldots + A_b$, and cross each piece A_i with a random piece of size $B - M$ drawn from $D - A_i$ to create a like-sized sample of size B. We replicate this process R times, holding A_i fixed but drawing randomly without replacement from $D - A_i$. This yields R samples of size B corresponding to A_i. We then compute the sampling distribution of propensity signatures of each value v in A_i from these R replications corresponding to chunk A_i, denoted by $\widehat{\mathcal{F}_{A_i}(v)}$.

We test the estimated signature $\widehat{s_A(v)}$ against $\widehat{\mathcal{F}_{A_i}(v)}$, and establish whether that particular value has a statistically different pattern of occurrence in A_i using the method described in Section 3.3.2. Each chunk A_i then gets to vote on the suspiciousness of the value v.

Definition 3.4. *A value v in set A is voted to be suspicious with respect to the empirical sampling distribution $\widehat{\mathcal{F}_{A_i(v)}}$ corresponding to chunk A_i of A if it is statistically significant with respect to that distribution. The vote is denoted by the indicator function $I_{A_i}(v)$ which takes the value 1 if significant, 0 otherwise.*

We repeat this step with each of the b pieces of A. Each chunk yields a vote $I_{A_i}(v)$ for each value v. The more votes a value has, the more confident we are about its significance and the more informative it is in an explanation.

Definition 3.5. *The informativeness of a value v is measured by the proportion of votes*

$$K = \sum_i I_{A_i}(v)/b.$$

In summary, the crossover sampling process results in a total of $T = R * b$ samples of size B, and a collection of *empirical sampling distributions* $\{\widehat{\mathcal{F}_{A_i}(v)}\}_{i=1}^{b}$ corresponding to the b chunks $\{A_1, \ldots, A_b\}$, each of which gets one vote for each value v.

We demonstrate the process in Figure 1(b), where:
B (subsample size, same as size of suspicious set A) = 3,
b (number of chunks) = 3,
M (chunk size) = $B/b = 1$,
q (crossover proportion) = $1/b = 1/3$,
R (the number of replications) = 3, and
T (the total number of subsamples) = $R * b = 9$.

In the process described above, all the R replications correspond to a given partition of $A = A_1 + A_2 + \ldots + A_b$ into b chunks. We could make the process more general by spreading the number of replications R over a small number of randomized partitions of A. For example, we could run $R/5$ replications for a given partition $A = A_1 + A_2 + \ldots + A_b$, another $R/5$ replications for $A = B_1 + B_2 + \ldots + B_b$ and so on until a final $R/5$ replications for the fifth partition $A = E_1 + E_2 + \ldots + E_b$. This would make it possible to combine the votes from each chunk with greater generality.

3.3.2 Testing for Statistical Significance

We flag a value v as significant if its propensity signature lies outside the chosen error bounds of its corresponding sampling distribution $\widehat{\mathcal{F}_A(v)}$. These bounds are computed component-wise for each attribute. We compare each element of the signature with the corresponding bootstrap distribution and if any element lies above or below the chosen bounds, (mean ± 2 standard deviations; 97.5 and 2.5 percentiles), we declare the signature to be significant. For example, consider the propensity signature

$$\widehat{s_A(ID_5)} = (1/3, 0, 0, 0, 0, 2/3)$$

from our illustrative example. The following error bounds are based on the mean ± 2 standard deviations (hence the negative and fractional values of the bounds) of the bootstrap sampling distribution corresponding to ID_5 :

```
Lower (2.5%) Bound:  (-0.18, 0, 0, 0, 0, 0.44)
Upper (97.5%) Bound:  (0.2, 0, 0, 0, 0, 0, 1.80).
```

Now, given that $\widehat{s_A(ID_5)}$'s first component corresponding to component (attribute 1), 1/3=0.33 is above the upper bound 0.2 for the corresponding component, we declare the value ID_5 to be significant even though for component (attribute) 6, 2/3=0.67 lies within the interval [0.44,1.8].

We use statistically significant propensity signatures that have been identified in this way to construct *explanations.*

3.4 Glitch Explanations

Let the collection of values v in A with statistically significant signatures be $\mathbf{V} = \{v_1, v_2, \ldots, v_L\}$.

Definition 3.6. *A glitch explanation $\mathcal{E} \subseteq \mathbf{V}$ is a collection of values in A that have statistically significant propensity signatures.*

For example, for the suspicious set A of duplicate phone numbers corresponding to phone number `1AAA3608776` discussed in Section 1.1, the estimated signatures of ID_5 and $NewHire$ are significant and lead to the explanation: $\mathcal{E} = ID_5, NewHire$.

Note that explanations need to be human interpretable (vetted by domain experts), and therefore the more succinct they are, the easier to understand and explain. To capture this aspect, we introduce the notion of the size of an explanation.

Definition 3.7. *The size of an explanation is the smallest number of informative, non-redundant values in the explanation.*

We can use a threshold on the informativeness (Definition 3.5) i.e. $K > \alpha$ for including a value v in an explanation, providing a measure of customizability to the data consumer.

In addition, the set of values with statistically significant propensity signatures could exhibit redundancy. For instance, if two values have a one-to-one relationship and always occur together in every record of the suspicious set A (e.g., unique organizational code such as "DEPT007" and a unique name like "Department of Shaken, Not Stirred"). Finally, values such as blanks are usually not informative and do not contribute towards a general explanation. Note that many redundancies can be automatically generated using algorithms for discovering functional dependencies and conditional dependencies e.g. in [9].

3.5 Evaluating an Explanation

We measure the efficacy of an explanation by the *statistical distortion* of the data prevented by reclaiming the data corresponding to the explanation. Statistical distortion can be measured in many ways, from simple measures like difference in aggregates such as means and medians, to more complex measures such as the histogram distance between two data sets D and D'. Different ways of measuring statistical distortion, including the Earth Mover Distance, are described in [6].

For the purpose of this paper, we use the general notion of the proportion of records that are touched by data repair. This is because any other metric, such as histogram distance, would need a knowledge of the actual repairs and changes made to the data. Since our focus is on explaining glitches and not statistical distortion *per se*, this general notion is enough for the purpose of illustration.

Any records that are reclaimed by glitch explanations and left untouched, result in a *reduction* in the statistical distortion caused by cleaning. Let S be the reclaimed set with size $|S|$. Then, the *reduction* τ in the statistical distortion is given by:

$$\tau = \frac{|S|}{|A|}.$$

Definition 3.8. *The merit τ of an explanation $\mathcal{E}$ is the reduction in statistical distortion caused by reclaiming the records explained by $\mathcal{E}$.*

For instance, we can reclaim 6 of the 9 suspicious records in the example of Section 1.1, resulting in a merit of

$$\tau = 6/9 = 0.667.$$

3.6 Automation and Scale

In general, we expect the suspicious set A to be small, and therefore the number of values for which to compute signatures to be relatively small as well. If not, the target values for which to compute propensity signatures can be selected on a prioritized basis e.g. top 5% of frequently occurring values.

In addition, the choice of parameters such as crossover proportion, q, and informativeness, K, while customizable, are rooted in statistical theory. We have found that $q \in [0.1, 0.25]$ and $K > 0.8$ are good default values.

Finally, while explanations are often validated by human experts, it is possible to refine and cross-validate them automatically by empirical repetition and replication.

4. EMPIRICAL FRAMEWORK

From the preceding discussion, it is is clear that our approach for generating signatures and explanations is data-driven and necessarily requires a rigorous experimental basis to ensure the validity of the empirical results. We describe the experimental framework.

4.1 Identifying the Suspicious Set

For the purpose of this paper, we assume that identifying the suspicious set A of data quality violations is simply a matter of testing well-defined constraints. However, it is likely that identifying A might involve uncertainty, in which case we might need an empirical approach, as in the case of disguised missing values [10].

4.2 Constructing Propensity Signatures

Each of the sets A and $D - A$ contain a collection of distinct values. We construct the empirical estimates of the propensity signatures described in Section 3.2 in a single pass over the data, for each distinct value in A and $D - A$. Let A have N_A rows (records). Suppose that v occurs n_k times in column (attribute) C_k in the suspicious set A. Let

$$\widehat{p_k} = n_k/N_A,$$

be an empirical estimate of the probability p_k. For example, in Figure 1, an estimate of the propensity signature of v is given by:

$$\widehat{s_A(v)} = (0, 2/3, 2/3, 1/3, 3/3, 0).$$

Similarly, the estimated propensity signature of v in $A' = D - A$ is given by

$$\widehat{s_{A'}(v)} = (\widehat{P_1}, \ldots, \widehat{P_k}, \ldots, \widehat{P_d}), k = 1, \ldots, d,$$

where $\widehat{P_k} = m_k/N_{A'}$, m_k is the number of occurrences of v in attribute C_k of A' , $N_{A'}$ is the number of rows (records) in A'. Note that $\widehat{p_k} = n_k/N_A$ is the maximum likelihood estimate (MLE) [13] of the true probability p_k that v will occur in the k^{th} column (attribute) of A, and similarly $\widehat{P_k} = m_k/N_{A'}$ is the MLE with respect to P_k, the probability that v will occur in the k^{th} column (attribute) of A'.

4.3 Resampling and Subsampling

Once we have identified A and $A' = D - A$, we need to isolate values $v \in A$ that help us to statistically differentiate A from A', in order to explain the suspiciousness. We accomplish this using crossover subsampling discussed in Section 3.3.1.

In our experiments in this paper, we used a thousand bootstrap replications ($R = 1000$) to generate 1000 signatures for each value $v \in A$. Note that the actual signature contributed by the suspicious set A is included in the 1000. We flag value v to be significant if any element in its signature lies outside the error bounds computed from the sampling distribution based on the 1000 bootstrap signatures. For the rest of this paper we use quantile based error bounds at the 0.005 level of significance. The results in the case studies in Sections 5 and 6 are based on a crossover proportion of $q = 0.1$, except in one set of experiments where we vary the crossover proportion. The computation was performed in 10 parallel R language batch jobs on a cluster of Intel multi-core Xeon processors (2.53GHz) running Scientific Linux 5.5 operating system.

5. ORGANIZATIONAL DATA

Our first real world data consisted of the Human Resources database of a large company. It contained 50,084 records, each record with 17 attributes. Our data quality constraint was "Given any telephone number, there should be only one record". Note that validating this constraint involves multiple records. When we applied the constraint, we found that 14,872 (29%) records were in violation, generated by 530 distinct phone numbers, each of which gave rise to a suspicious set of duplicates. The sets came in 54 distinct sizes. Only a handful of suspicious sets had significant sizes, most had fewer than 100 records, considerably fewer than the overall set size of 50K records. The distribution of the size of the suspicious sets is shown in Figure 2(a), where the X-axis is the size of the suspicious set and the Y-axis the number of suspicious sets with that size. The axes are staggered with variable scales for better readability. Only two sets have more than 1000 records, and most have fewer than 50 records, with suspicious sets with just 2 records accounting for more than 100 such sets. We list the 5 most duplicated phone numbers below. While we have anonymized the actual values for proprietary reasons, the explanations are real.

```
+1 (BBB) 999-9999 (8011 duplicates);
+ CCC9999999      (2209 duplicates);
+1 (DDD) 392-2600 (619 duplicates);
                  (538 duplicates);
+1 (EEE) 000-0000 (475 duplicates).
```

It is interesting that at first glance, the worst offenders (with the exception of `+1 (DDD) 392-2600`) seem to be bogus phone numbers used as defaults.

Let us first consider the explanation corresponding to the non-trivial duplicate phone number `1 (DDD) 392-2600` which is duplicated 619 times. It is given by:
$\mathcal{E}$ (`1 (DDD) 392-2600`)={ MeanBoss, USA, HER MAJESTY'S CUSTOMER SERVICE, MIRAMAR, FL, C, Contractor, Bogus Co., =33027, LAKESIDE DR STE 620 }
The signature provides an interesting explanation. The phone number corresponds to `Contractors` that work for the company `Bogus` under supervisor `MeanBoss`; and are located at `LAKESIDE DR STE 620, MIRAMAR, FL, 33027, USA`; shared the phone number `1 (DDD) 392-2600`; and were dedicated to working on `HER MAJESTY'S CUSTOMER SERVICE`. It is a centralized office number, and is therefore acceptable as a duplicate.

Similarly, consider the explanation for the phone number `1 (EEE) 000-0000` duplicated 475 times:
$\mathcal{E}$ (`1 (EEE) 000-0000`) = { USA , C, SuperBoss, WESTLAKE VILLAGE , Q's SOLUTIONS, TX , Shady Marketing, MI, Fishy Co. , CA , =91361, =76054, TOWNSGATE RD, NORWOOD DR }
The phone numbers corresponds to employees that work for supervisor `SuperBoss`, and were contracted from the companies Fishy Co. or Shady Marketing, to work on Q's SOLUTIONS and given the default phone number of `1 (EEE) 000-0000`.

The explanation for the worst offender `+1 (BBB) 999-9999`, after removing redundancies like blanks and other values like states and cities, consisted of 18 zip codes, corresponding to locations of contractors working across the USA. The zip codes appeared only in this suspicious set and not in any of the other suspicious sets, making them distinctive.

These examples show that while the phone numbers were "dirty" and violated a constraint, the other attributes provide enough of an explanation for us to trust that data and reclaim it for regular use. With just the five suspicious sets described above, we were able to reclaim 11,852 of the 14,872 duplicate records.

5.1 Reclaiming data with Explanations

In general, explanations consisting of between 3 to 20 values are ideal. Very small explanations might not be specific enough (e.g. "USA"), and those with too many values might be hard to interpret. Sometimes, by relaxing the threshold K from 1 to 0.8, we found more useful explanations.

On other occasions, despite our best efforts, the explanations were not useful. Consider the one associated with a small suspicious set of 29 duplicates.
$\mathcal{E}$ (`1 0`) = { C }
Clearly, even the phone number `1 0` is mangled. And the significant value in the explanation, "C", does not provide any useful information other than the fact that it is a code associated with contractors. The locations were spread across multiple cities in multiple countries, across multiple services and supervisors. There was no discernible pattern. Therefore we could not salvage these records.

In total, using our method we could explain all but around 70 records corresponding to suspicious sets of size 29, 16, 6, and 2 (multiple sets) which did not garner enough votes to pass confidence guarantees. Therefore, in this real world case study, our explanations reclaimed around 14,800 records and caused a total reduction in statistical distortion of

$$\sum \tau = (14800)/14872 = 0.9951.$$

Therefore the collective merit of our explanations for the Organizational data is 0.9951. Next, we ran experiments to test various sampling parameters.

5.2 Experiment 1: Robustness of Explanations

To test whether the explanations for close variants of a phone number are similar, we took a suspicious set and created two syntactically different variants. One set corresponding to `FFF7474014` with |A|=298, and second set corresponding to `FFF 747 4014` with |A|=30. We were gratified that our method generated the same explanation for both, namely:
$\mathcal{E}$ = { "M", IIND, II, GURGAON, GRGNIIAF, DLF ATRIA PHASE II GURGAON HARYANA, Desi Company, Contractor, BOND CUSTOMER EXPERIENCE, C, =122 002 }
From this we could infer that both sets of duplicate phone numbers corresponded to contractors from an Indian company assigned to supervisor `"M"`, and that the two phone numbers were *near-duplicates*, but our method was robust

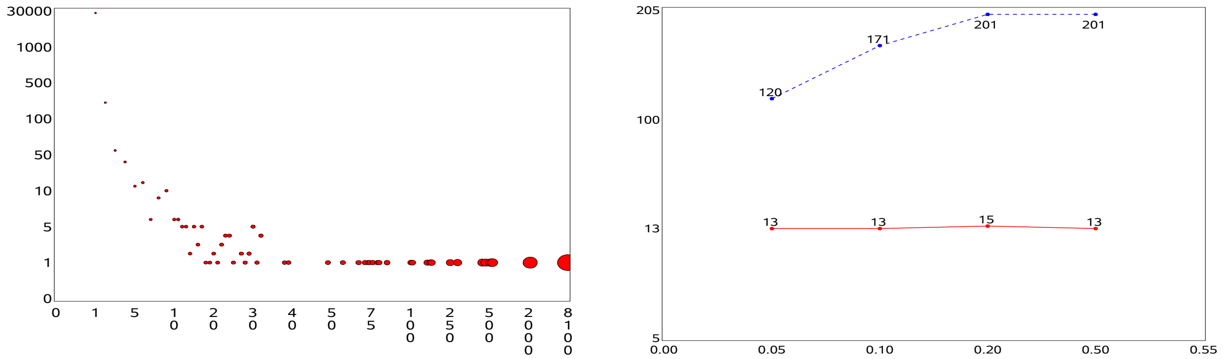

Figure 2: (a) For the case study in Section 5, the distribution of the size of suspicious sets; (b) The change in the size of explanations with crossover proportion q. For the suspicious set of `+1 (BBB) 999-9999` (shown as blue dashed curve), the size of the explanations increases with the crossover proportion. For the suspicious set of `FFF7474014` (shown as solid red curve), the size of the explanation is very steady, implying a well-defined succinct set.

enough to generate the same set of values in the explanation, despite the difference in sizes of the suspicious sets.

5.3 Experiment 2: Crossover Proportion

For our second experiment, we chose two suspicious sets corresponding to two different phone numbers in the HR data. The first suspicious set consisted of duplicates of `+1 (BBB) 999-9999` and had 8011 records with 33,063 unique values. The second suspicious set consisted of duplicates of `FFF7474014` and had 298 records with 1,364 unique values. We varied $q = (0.05, 0.1, 0.2, 0.5)$ and measured three quantities for each of the two suspicious sets: (1) the number of significant values that got at least one vote from the $b = 1/q$ chunks, (2) the size of the explanation defined as the number of values within each proportion q that got a unanimous vote, i.e. $K = 1$ from Definition 3.5, and (3) the number of "clean" values in the explanation that got a perfect vote of $K = 1$ for all four values of the crossover proportion q.

Figure 2 (b) shows the size of the explanations with each crossover proportion. The X-axis shows the crossover proportion q, the Y-axis shows the size of the explanation. In this particular discussion, the size is based solely on votes, and not on redundancy. We chose to keep the redundant values in order to maintain comparability since they will be consistently included in all signatures. In Figure 2(b), the suspicious set of `+1 (BBB) 999-9999` (shown as blue dashed curve) had 283 significant values of which 100 were unanimous in all the proportions. The size of the explanations increases with the crossover proportion, almost doubling. This indicates that there is no succinct explanation, it just expands as the bootstraps include more and more of the suspicious set. (Most of these were redundant values like geographical states and "USA". As noted above, we reduced these to 18 non-redundant values of zip codes.) The suspicious set of `FFF7474014` (shown as solid red curve) had 36 significant values of which 13 were unanimous in all four proportions. The size of the explanation is very steady, implying a well-defined succinct set.

6. MOBILE TELEPHONY DATA

Our second example consists of mobile telephony data, collected over a period of two weeks. We anonymized the data by preserving the NPA (area code) of each phone number and hashing the 7 digit phone number in a consistent fashion. Next, we aggregated the data by zip codes and into 15 minute bins. Each zip code was associated with a Metro area. The variables of interest include number of calls made, number of texts sent, number of calls dropped during set up, and number of calls dropped while the call was in progress. The aggregated data set had 27,291,446 records. A sample:

```
ZIP|UNIX-TIME|CALLS|BAD-1|BAD-2|TEXTS|METRO AREA
10001|1360231200|208|0|0|463|Manhattan
10001|1360232100|227|0|2|410|Manhattan
```

We suspect that some of the data might be erroneous for reasons that are probably data quality related, rather than anything to do with the actual network performance. A rationale is outside the scope of this paper, but the data quality constraint was specified by experts as:

$$C_T \leq 0.25,$$

where C_T is the ratio of the sum of `BAD-1` and `BAD-2` calls to the total number of calls handled. Any records that violated this constraint i.e. $C_T > 0.25$ were considered suspicious and put in the set A. In reality, there were only 78,464 records across more than 27 Million records. Note that this instance is (1) a valid case where $|A| << |D|$, and (2) an example of a single-record constraint where each record can be assessed for constraint violation independently of other records.

Even though the suspicious records were scattered, we wanted to study if we could (1) recover patterns if they existed in such large data of millions of records and (2) study the impact of scale on our method. The mobility data used in the following two studies study was synthetic data created from the real mobility data described above, by injecting suspicious data from the set A in a controlled manner.

Suspicious Zip Codes: We took 174,326 records corresponding to the zip codes from metro areas labelled New York, Other NYC Boroughs, Chicago, Chicago Loop and San Francisco and created a test data set D_T. This particular data set had no bad records at all. We simulated bad data by injecting bad records from the suspicious set A into the test data set D_T. We selected the worst 6 zip codes from

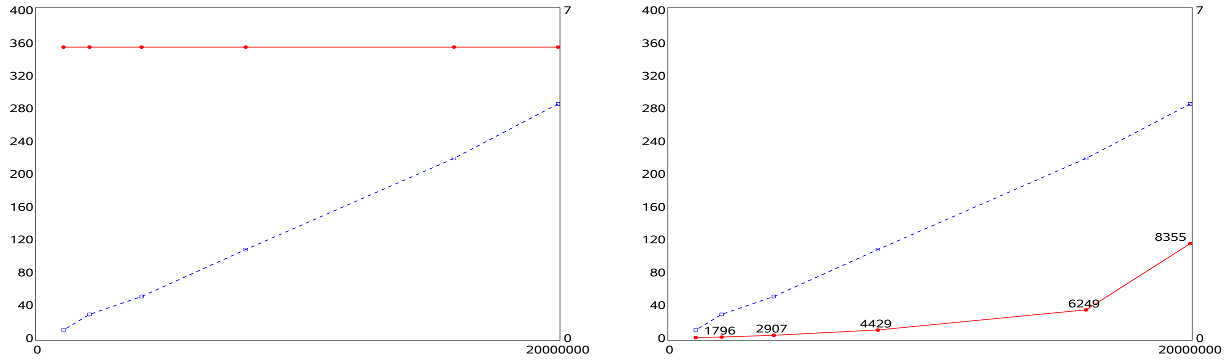

Figure 3: Plots (a) and (b) show the run times of (1) creating the crossover samples (blue dashed line) and (2) computing the signatures and the bootstrap sampling distribution (solid red curve) as the sizes of the good and bad data set vary. In (a) the bad data set is fixed at 78000 while the good data set varies as shown on the X-axis. In (b) the sizes of both the data sets vary. The two curves are measured on different scales (Y-axes), the red curve against the Y-axis on the left side, the blue curve on the Y-axis on the right side of the plots. The numbers on the curve represent the number of unique values in the bad data set.

A listed under the column "Worst Zip" of Table 2. These 6 zip codes contributed 1,517 bad records, and a total of 7,146 records. We then chose 6 zip codes in D_T that had a similar distribution of records as the 6 worst zips. We removed all these records, and replaced them with all the records corresponding to the 6 worst zips. Finally, we mapped the actual zip codes of the worst 6 zips to the zip codes of the records that we removed from the data set D_T. The mapping is shown in Table 2. While mapping the zips is not necessary, we do it for reasons of consistency to keep the zip codes within the metro areas for interpretation purposes. Note that the simulated bad zip codes correspond to the three metro areas Chicago Loop, San Francisco and Manhattan.

The resulting data set D_{ZIP} now has 174,391 total records with 1,517 bad records concentrated in 6 zips listed under the column "Mapped To Zip" of Table 2. That is, if the bad records occur in these zip codes, then they are not suspicious since we put them there. Our method correctly generated the zip codes 94143, 94119, 60603, 60602, 10041, 10020 as significant values, each with 10 votes from each of the randomizations. In addition, the following explanation for the Metro areas were generated. The number of votes are shown in parentheses: Chicago Loop(10), San Francisco(10) for the upper tail (much higher). Chicago Loop and San Francisco are as expected due to the zip code mapping, and are redundant to the zip codes. However, the metro area Manhattan did not show up. This is because the proportion of Manhattan was the same in the bad data as it was in the good data. But this has no impact on our explanations since Metro areas are redundant to the zip codes.

Worst Zip	MappedTo Zip	MappedTo Metro	Total Records	Suspicous Records
88434	60602	Chicago	1329	395
80744	94119	San Fran	1303	252
55925	10041	Manhattan	900	229
93225	60603	Chicago	1344	220
03278	10020	Manhattan	1249	212
67481	94143	San Fran	1021	209

Table 2: Zip mapping

Therefore our minimal explanation is that bad records in the following zip codes will not be considered "dirty" since we expect them to be there.
$\mathcal{E}$ = { zip codes: 94143, 94119, 60603, 60602, 10041, 10020 }.
As a consequence, we are able to reclaim all the suspicious records, resulting in a statistical distortion of 0. The merit τ of our explanation is 1.

Had we not used the explanations, the statistical distortion associated with the dirty data would have been
$StatisticalDistortion = 1517/174391 = 0.0087$.

Suspicious Time Periods: We simulated badly behaved time periods in a manner similar to the zip codes. We took the worst 6 15-minute time slots, namely 2.45 AM, 3.00 AM, 3.15 AM, 3.30 AM, 3.45 AM, 4.00 AM, from the master file of 27 Million records. These contributed 7908 records. We replaced the data corresponding to these time slots in the test dataset D_T with the bad data. Table 3 shows the number of injected records corresponding to the time slot, and the corresponding suspicious records contained in them.

Time Slot	Suspicious Records	Total Records
02:45 AM	1266	1338
03:00 AM	1247	1326
03:15 AM	1271	1322
03:30 AM	1212	1276
03:45 AM	1301	1356
04:00 AM	1229	1290

Table 3: Simulated time slots

We derived the time and day of the week from the Unix time stamp and generated the following explanations.
$\mathcal{E}$ = { Tue, Wed, Thurs, Sat, Sun; 2.45 AM, 3.00 AM, 3.15 AM, 3.30 AM, 3.45 AM, 4.00 AM }
The explanation of the bad time periods holds for 5 days of the week, but doesn't seem to include Mondays and Fridays. The bad records corresponding to these two days that fall

in the 6 identified time slots, 1,090 and 1,148 respectively, cannot be explained. Therefore the merit of our explanation is:

$$\tau = (7908 - (1090 + 1148))/7908 = 0.717.$$

6.1 Experiment 3: Size of Data Sets

Finally, we wanted to test the scalability of our method, in terms of computation as well as robustness of explanations, by varying the sizes of both the suspicious set A and the good set $A' = D - A$. We resampled from the mobility data described at the beginning of this section to create the following synthetic data. The experiments are summarized in Figure 3 which features two Y-axes, on the left and right side of the plotting frames, one for each of the two curves that are measured on different scales. The X-axis denotes the number of records in the good data set.

First, we kept the bad data set at the fixed size of 78,464 records (10,111 unique values) and varied the size of the good data set from 1Million records to 2, 4, 8, 16 and 20 Million records. We split the task into two steps (1) create the bootstrap crossover subsamples (dashed blue curve in Figure 3 measured against the Y-axis on the right side of the plotting frame) and (2) compute the signatures and the corresponding sampling distribution from the bootstrap samples (solid red curve in Figure 3 measured against the Y-axis on the left side of the plotting frame). We found, as expected, that creating the crossover subsamples from the good data set to form the 1000 bootstrap samples took longer with the increase in the size of the good data. In Figure 3(a), the dashed blue curve increases with the size of the good data, measured against the second Y-axis on the right side of the frame. The computation of the bootstrap signatures and the sampling distribution depends only on the size of the samples (fixed at 78000) and the number of bootstraps (fixed at 1000). This is reflected in the solid straight line at the top of the plot measured against the first Y-axis on the left side of the frame. We also found that the set of suspicious values, and hence the resulting explanations, were the same in all cases.

Next, we changed the sizes of both the good and bad data. The good data was varied just as above, but in addition we varied the size of the bad data over 1250, 2500, 5000, 10000, 20000 and 40000 records. Number of unique values in the bad set are denoted on the red curve. As expected, the computing times increased with both the data set sizes, for both the sample creation (dashed blue) and the signature and sampling distribution computation (solid red). However, fewer suspicious values were deemed significant in the instances with smaller bad data sets. This phenomenon was mainly due to the smaller number of unique values in the bad data records. For example, the smallest bad data set of 1250 records is about 1.5% of the original bad data set of 78,464 records. In addition, there is more uncertainty (variance) in the distribution of the bootstrap signatures for the smaller bad data sets. Many of the values, even when significant, fail the strict criterion of $K = 1$, i.e. all 10 blocks generated by the $q = 0.1$ crossover proportion must vote for the signatures.

7. CONCLUSION

In this paper, we introduced the notion of *empirical glitch explanations*, which are data-driven, multi-attribute descriptions of subsets of potentially dirty data. The explanations are used by domain experts to decide whether the data is genuinely dirty, or is acceptable based on the explanations. The explanations reduce the amount of data subjected to unnecessary repair, and reduce the statistical distortion induced by cleaning. We evaluate explanations based on their *size*, which is related to usefulness and interpretability, and *merit*, the amount of statistical distortion prevented by the explanations.

We described an empirical framework for generating glitch explanations by proposing a novel subsampling technique called *crossover subsampling*. We demonstrated the utility of our approach based on real world data sets where we could reclaim up to 99% of the data, and ran experiments to demonstrate the scalability and robustness of our method.

A major thrust of our future work, which could have critical applications in Big Data, involves generating and formalizing the domain knowledge we learn from the glitch explanations, in order to: (1) Reason with it and answer questions e.g. "what is the most common explanation for glitches in one organization vs another?" and (2) Analyze explanations over time as new data gets added to understand the temporal nature and frequency of glitch patterns.

8. REFERENCES

[1] L. Berti-Equille, T. Dasu, and D. Srivastava. Discovery of complex glitch patterns: A novel approach to quantitative data cleaning. In *ICDE*, 2011.

[2] L. Berti-Equille, J. M. Loh, and T. Dasu. A masking index for quantifying hidden glitches. In *ICDM*, 2013.

[3] F. Chiang and R. J. Miller. Discovering data quality rules. *PVLDB*, 1(1):1166–1177, 2008.

[4] F. Chiang and R. J. Miller. A unified model for data and constraint repair. *ICDE*, 2011.

[5] T. Dasu and T. Johnson. *Exploratory Data Mining and Data Cleaning.* John Wiley, New York, 2003.

[6] T. Dasu and J. M. Loh. Statistical distortion: Consequences of data cleaning. *PVLDB*, 5(11):1674–1683, 2012.

[7] B. Efron and R. J. Tibshirani. *An Introduction to the Bootstrap.* Chapman and Hall, 1993.

[8] W. Fan. Data quality: Theory and practice. In H. Gao, L. Lim, W. Wang, C. Li, and L. Chen, editors, *Web-Age Information Management*, volume 7418 of *Lecture Notes in Computer Science*, pages 1–16. Springer Berlin Heidelberg, 2012.

[9] L. Golab, H. J. Karloff, F. Korn, D. Srivastava, and B. Yu. On generating near-optimal tableaux for conditional functional dependencies. *PVLDB*, 1(1):376–390, 2008.

[10] M. Hua and J. Pei. Cleaning disguised missing data: A heuristic approach. *KDD*, 2007.

[11] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. *IJCAI*, 1995.

[12] R. K. Pearson. The problem of disguised missing data. *SIGKDD Explor. Newsl.*, 8(1):83–92, June 2006.

[13] C. R. Rao. *Linear Statistical Inference and Its Applications.* Wiley, New York, 1973.