Towards Sustainable Insights

or why polygamy is bad for you

Carsten Binnig Lorenzo De Stefani Tim Kraska Eli Upfal
Emanuel Zgraggen Zheguang Zhao

Department of Computer Science, Brown University
{firstname_lastname}@brown.edu

ABSTRACT

Have you ever been in a sauna? If yes, according to our recent survey conducted on Amazon Mechanical Turk, people who go to saunas are more likely to know that Mike Stonebraker is not a character in "The Simpsons". While this result clearly makes no sense, recently proposed tools to automatically suggest visualizations, correlations, or perform visual data exploration, significantly increase the chance that a user makes a false discovery like this one. In this paper, we first show how current tools mislead users to consider random fluctuations as significant discoveries. We then describe our vision and early results for QUDE, a new system for automatically controlling the various risk factors during the data exploration process.

1. INTRODUCTION

"A new study shows that drinking a glass of wine is just as good as spending an hour at the gym" [Fox News, 02/15]. "A new study shows how sugar might fuel the growth of cancer" [Today, 01/16]. "A new study shows late night snacking could damage the part of your brain that creates and stores memories" [Fox News, 05/16].

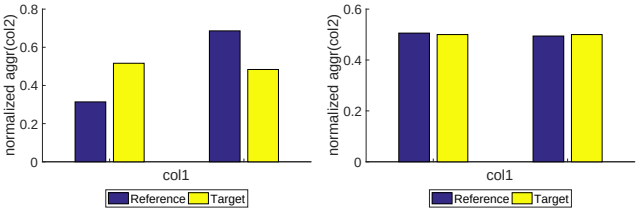
Over the last years we have seen an explosion of data-driven discoveries like the ones mentioned above. While several of these are indeed legit, there has also been an increase of more and more questionable findings [19]. Albeit the reasons behind this trend are manifold, we recently observed that the research community started to develop tools, like Vizdom/IDEA [7], SeeDB [34] or Data Polygamy [5], that are likely to considerably increase the number of false discoveries from data analysis. For instance, visual data exploration tools, such as Vizdom/IDEA [7] or Tableau, significantly simplify data exploration for domain experts and, more importantly, novice users. These tools allow to discover complex correlations and to test hypotheses and differences between various populations in an entirely visual manner with just a few clicks, unfortunately, often ignoring even the most basic statistical rules. For example, in a recent user study that we performed, we have asked people to explore a dataset containing information about different wines and report their findings. Using histograms showing American vs. French wines, most subjects came to the conclusion, that French wines earn higher critic ratings on average. At the same time, almost none of the participants used a statistical method to test if the visually observed difference from the histogram is actually meaningful. Similar, none of the participants, even the more statistically savvy ones, did consider that the arbitrary exploration and attempts to find interesting facts actually increases their chance to find random occurrences of seemingly significant correlations.

While these concerns are often waived under arguments such as "scientists are in desperate need for better tools, so any help is better than none", it is often not recognized that these tools not only greatly increase the likelihood of spurious discoveries but in many cases, make it also impossible to control the number of false discoveries later on. For example, recent visual recommendation systems, such as SeeDB [34] or Data Polygamy [5], are potentially checking thousands of hypotheses in just a few seconds and are smoking guns hiding as water pistols. SeeDB tries for example to find interesting visualizations and while a visualization per se does not seem like a hypothesis test, it should be treated as one. Why otherwise should a visualization be considered interesting, if the effect shown by the visualization is not relevant?

As a result, by testing thousands of visualizations it is almost guaranteed that the system will find something "*interesting*" regardless of whether the observed phenomenon is actually statistically relevant or not. Similar, Data Polygamy tries to find interesting correlations between time series data. Hypothesis tests are therefore actually performed using MC-methods relying on a fixed threshold for the p-values, without providing a correction for multiple hypotheses. Even more astonishingly, while the authors do discuss p-value adjustments using the Bonferroni correction, they then – surprisingly – disregard it. Let us assume the system has to test 100 potential correlations, 10 of them being true. Assuming a p-value of 0.05 (as suggested in [5]), and that our test has a statistical power of 0.8 (common values for a single statistical test), Data Polygamy on average will find 13 correlations of which 5 ($\approx 40\%$) are "*bogus*" (i.e., they are *false positives*). Even worse, without knowing how exactly the system tried to find the "interesting" correlations and how many correlations it tested, it is later on impossible for the user to determine what the expected false discovery rate will be across the whole data exploration session.

In this paper, we outline our vision and initial results for QUDE, the first system to Quantifying the Uncertainty in Data Exploration, which is part of Brown's Interactive Data Exploration Stack (BIDES). In order to better quantify the severity of the problem, we analyze existing visual recommendation systems, namely SeeDB and Data Polygamy, and discuss how they are prone to find wrong insights using simulated and real-world data. We further quantify the risk for data exploration systems like Vizdom/IDEA, which is not as severe as for automatic insight finders, but still persists because of its capability to quickly test many different hypotheses.

8th Biennial Conference on Innovative Data Systems Research (CIDR '17) January 8-11, 2017 , Chaminade, California, USA.



(a) Interesting visualization (b) Uninteresting visualization

Figure 1: Examples of interestingness as defined in [34].

Afterwards, we discuss QUDE and how it achieves a more sustainable data discovery process based on techniques for controlling the False Discovery Rate (FDR) [2]. Furthermore, we also show how QUDE tries to automatically infer the tested hypotheses based on the user interactions, and how we plan to incorporate user feedback, as well as, warn the user about potential risk factors. While QUDE is designed mainly to avoid the risk of multi-hypothesis testing, it also includes techniques to tackle other risk factors such as missing data or visually misleading results (e.g., Simpson paradox).

Our main contribution is twofold: first, we demonstrate the risk of false discoveries by using three example systems, Vizdom/IDEA [8, 7], SeeDB [34], and Data Polygamy [5] (Section 2). However, the insights gained from these systems apply to a large range of commercial (e.g. [16]) and research prototypes (e.g. [34]). Secondly, in Section 3, we present our vision of QUDE and initial techniques we use to control the amount of false discoveries.

2. THE RISK WITH TODAY'S TOOLS

Modern tools for interactive data exploration enable domain experts and novice users alike to efficiently analyze large amounts of data. At the same time, if not used carefully, these tools can significantly increase the risk of making spurious discoveries. In this section, we analyze different tools for data exploration and discuss how these tools amplify the risk of false discoveries. Later, in Section 3, we present techniques based on a statistical concept called "*False Discovery Rate*" (FDR) and how we adopt these techniques for data exploration to control the risk factors.

2.1 Visual Data Exploration

Visualizations are arguably the most important tool to explore, understand and find insights in data. As part of interactive data exploration, visualizations are used to skim through the data and look for interesting patterns. It comes therefore at no surprise that the database research community over the last few years focused on developing techniques (e.g., adaptive indexing, approximate query processing) to better support interactive exploratory workloads [17]. Visualization systems such as Vizdom [7] are capable of visualizing large-scale data with interactive speed. While interactivity is key to the usability of advanced analytical tools [25], using them unfortunately also significantly increases the risk of making spurious discoveries. Such risk has two aspects:

(1) The statistical significance of the visualized results is unclear.
(2) The growing number of hypotheses being tested during exploration increases with every single visualization.

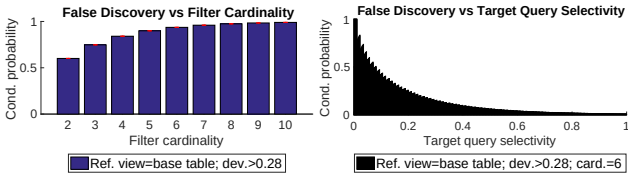
The first aspect of risk is important because visualizations have the power to influence human perception and understanding by the rich information they may carry. Suppose that a salesperson of an ice cream company is exploring a data set about the sales. As the first step, she wants to get a yearly distribution of the sales figures. So she compares the sales of the last five years using a histogram of sales per year. In the second step, she is interested in learning if the the sales differ significantly across different states. She thus compares sales per state over the last five years.

Suppose the histogram shows that sales in Vermont were higher than in Rhode Island. Consider how tempting it is for an unsophisticated user to conclude that Vermonters buy more ice cream just based on the visualization. Although a statistically inclined user would formally analyze this observation by using hypothesis testing, she would have to redirect her attention to work with a different statistical tool (e.g. R [12]) before proceeding to the next data exploration step. After such efforts on context-switching, the insight might turn out completely due to random noise. At scale, the division of labor between data exploration and hypothesis testing will cause even more waste of human efforts on such spurious insights. Thus, *if a visualization provides any insight, the insights should be immediately tested for their significance*. If that would not be the case, the value of the visualization would be very limited as the user would not be allowed to make any conclusions based on the visualization. Thus if we consider a visualization as something more than a pretty picture presented to the user (i.e., more than just a listing of facts), we should always test the insight the user gains from the visualization for its significance and inform the user about it. A central challenge is of our work is the understanding of the hypothesis derived by the user given a certain data visualization. With respect to the previous example, the hypothesis derived by the user could be: (1) Vermonters buy more ice cream than Rhode Islanders, (2) Rhode Islanders buy more than Vermonters, or (3) they buy ice cream in the same amount.

The second aspect of risk is arguably even more severe. With every additional hypothesis test the chance of finding a false discovery increases. This problem is known as the "*multiple comparisons problem*" and has been studied extensively in the statistics literature [3, 1, 15, 23]. While only a decade ago it was an art left to experts, data analytics has become more and more accessible to a broader range of users with the advent of open-source data analysis systems. What has clearly changed is how easy it has become to test an hypothesis. For instance, if ten years ago it took a scientist one day to do a single test (including determining the right test statistic, collecting and cleaning data, etc.), it is now very easy to do 20 or more in a few minutes on a system such as Vizdom [7]. Assuming a significance level of 5%, for 20 tests the risk to falsely reject at least one true null hypothesis increases to $1 - (1 - 0.05)^{20} = 64\%$.

Data exploration on systems such as Vizdom [7] not only increase the risk of false discovery, but also change the way how statistical tests are applied. Suppose in the previous example the salesperson explores various relationships in the sales dataset through visualizations until she sees a visualization that she deems useful (e.g. significantly more ice cream sales to males in Massachusetts compared to California). With some statistics background, she validates this insight by using an appropriate test with a significance level of 5%. Suppose the observed p-value is below the significance level, she rejects the null hypothesis and believes that there is only a 5% chance that she incorrectly rejected the null hypothesis in case it was true. However, this way of applying statistical test is wrong. What the user ignores is that before she did the test she had already searched through the dataset for a while, and had observed different insights and implicitly their corresponding hypotheses, albeit untested. Thus, by the time the user applied the statistical test, she was already inadvertently trapped into the multiple comparisons problem, because the data exploration tool provided her the illusion that data exploration was not a sequence of hypotheses.

To conclude, without considering the risk of false discoveries, current interactive data exploration tools have the propensity to significantly exacerbate the problem of considering random occur-



(a) Multiple Target Queries (b) Single Target Query

Figure 2: The risk analysis of false discoveries in SeeDB [34]

rences as insights. Rather than limiting data exploration exclusively to statisticians, we believe in empowering both unsophisticated and advanced users with more intelligent systems with automatic risk control, where data-driven insights can be drawn both efficiently and safely. While it is clearly not the tool's fault that false discoveries happen, in the end it is the user's, tools like Vizdom, Tableau and many others purposefully target a broader audience of users. That is, more users without a sufficient statistic background will be using these tools and not understand the risk factors. Furthermore, even trained statisticians struggle to fully control the multi-hypothesis problem, which in theory requires keep track of every single insight every user ever made over a given dataset.

Therefore, with QUDE we plan to build a system which actively makes the user aware of these problems during the data exploration process and controls the risk of false discoveries automatically. Unfortunately, traditional techniques for multi-hypothesis testing, such as the Bonferroni correction, are both too pessimistic and require the system to know the number of hypotheses the user can explore upfront, which make them inadequate for data exploration.

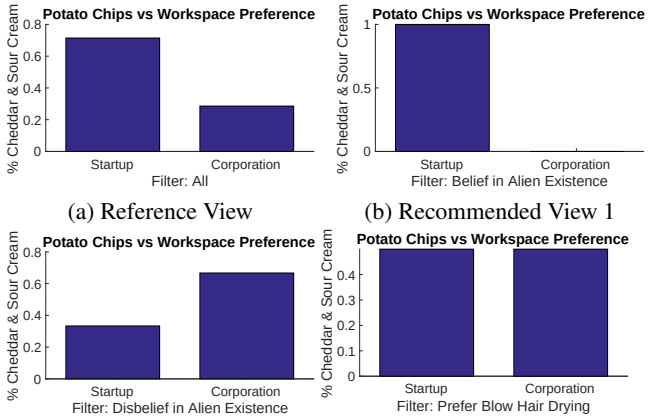
2.2 Visual Recommendations

To automate data-driven discoveries at scale, *visualization recommender systems* such as Scagnostics [27], SeeDB [34], VizDeck [21], or Voyager [36] have been proposed. None of them however considers the risk of false discovery.

In a nutshell, visualization recommendation systems automate two dimensions of the visual data exploration process: (1) Recommend new visualizations with the goal to provide new insights (2) provide better representations for a given visualization. We focus on the first type of recommendation systems as they are similar to the systems discussed in Section 2.1, except for the fact that the system itself becomes the explorer of the data and is capable of checking thousands of hypotheses in just a few seconds. Without controlling the risk of false discoveries, these systems systematically increase the risk of spurious discoveries at scale. For the remainder of the discussion we use SeeDB [34] as an example, though similar observations can be made for other systems.

SeeDB considers the current query and according visualization of the user (i.e., the *reference query*) and offers recommendation (i.e., the *target query*) by adding/changing filtering and group-by attributes etc. To rank the recommendations, SeeDB recommends to the user the most interesting target queries based on the deviations from the given reference query. SeeDB assumes a larger deviation indicates a more interesting target query. Figure 1a and Figure 1b show examples of "*interesting*" and "*uninteresting*" target views. Furthermore, SeeDB truncates uninteresting visualizations if the deviation value is below a certain threshold.

Unfortunately, the deviation in SeeDB may just result from random noise, and thus carries no statistical significance. In [34], the authors report that the discovery rate for interesting visualizations with SeeDB is three times higher than for manual exploration tools such as Tableau [16] or Vizdom [7]. This increase of efficiency is troubling because the false discoveries also increase in an uncontrolled manner.



(a) Reference View (b) Recommended View 1

(c) Recommended View 2 (d) Recommended View 3

Figure 3: An example of SeeDB [34] on survey data.

2.2.1 *False Discoveries on Random Data*

As a first step to show how visual recommendation systems such as SeeDB [34] suffer from false discoveries, we use a probabilistic model to analyze how likely it is that SeeDB finds a large deviation from the reference view on random data without any real correlations. We focus on SeeDB's capability of adding and changing a single filtering condition to the reference query. We ignore more advanced variations (e.g. multiple filter conditions, adding group-by's, etc.), as all of these would further increase the chance of false discoveries. For simplicity our model makes the following assumptions: (a) the aggregate function is `SUM`, the aggregate column has zero variance and the group-by column has uniformly distributed binary values; (b) the filter column is a sequence of Bernoulli trials; (c) the selectivity of attribute values on the filter column is drawn from a multinomial distribution; (d) all columns are independent; (e) we used the same deviation distance as in [34] for the recommendation threshold.

Figure 2 shows the risk of the SeeDB model making recommendations based on the random effects. The first simulation in Figure 2a varies the number of unique values in the filtering attribute (called filter cardinality), which corresponds to the number of target queries per reference query. The support size (i.e., number of selected tuples) of each target query is kept constant at 100. Given higher filter cardinality, more target queries are compared against the reference query, and thus the risk of false discovery increases. The second simulation in Figure 2b uses 1000 records, keeps the filter cardinality constant at 6 but varies the selectivity of the predicate and with it the support size. The figure shows — not surprisingly — that the lower the selectivity, the higher the chance of a false discovery because of the reduced support size.

2.2.2 *False Discoveries on Survey Data*

As a second step to verify what spurious recommendations SeeDB would make, we collected 104 answers for 69 (mostly unrelated) multiple-choice and 17 fill-in-the-blank questions on Amazon Mechanical Turk [18]. Questions range from *Who is Mike Stonebraker?* to *What is your eye color?* and *Have you ever been in a Sauna?*. Each answer is treated as an attribute.

We implemented the SeeDB recommendation algorithm, and used simple queries with one aggregated and one group-by attribute but without any filtering condition as the reference views. The deviation threshold was set according to the example in [34]'s Figure 1. As a result, our SeeDB implementation generated $2,078,608$ target views (i.e., potential recommendations) based on $9,996$ reference views, among which a stunning $708,109$ were recommended,

# Records	100
# Attributes	11
# Datasets	2
Extreme data prob.	20%

(a) Random input

# Trials	50
Significance level	0.05
# Incorrect rejection	43
# Correct acceptance	7

(b) False discoveries

Figure 4: Data Polygamy [5] on random extreme points.

though many of which are statistically insignificant.

Figures 3a-3d show example spurious recommendations. Suppose the user analyzes the preference of Potato Chips (Cheddar vs. Sour Cream) based on the Workspace Preference using the reference view in Figure 3a, which is already a questionable finding on its own. Still SeeDB recommends three of the top-ranked target views shown in Figure 3b-3d (ranked by the deviation beyond the threshold as in [34]'s example), which are even more questionable and do not hold up in our statistical test. For instance, the recommendation in Figure 3c shows that the disbelief in aliens reverts the trend compared to the reference view, though the correlation between disbelief in aliens and preference of potato chips is insignificant (p-value of 0.59). On the other hand, SeeDB also recommends views based on statistically significant yet questionable correlations, such as the correlation between Saunas and Stonebraker from the abstract, which even passes our post mortem statistical test (p-value of 0.036). *Thus, even if the user would perform a statistical test after seeing the visualization, she might wrongly assume that the insight is significant as she would certainly never consider the risk the visualization recommendation system introduced by searching for an "interesting" visualization.*

2.3 Automatic Correlation Finders

As a last class of system, we analyze recent recommendation engines, which not just suggest visualizations but try to automatically find insights through automatic hypothesis testing. One of such systems is Data Polygamy [5], which searches for statistically significant correlations in temporal-spatial datasets. Such correlations may exist at certain time or location. For example, the wind speed may not correlate with the number of taxi trips during the year, but it may when the hurricane strikes. Data Polygamy first identifies extreme data points, then uses the F1 score to measure the relationship strength, and performs Monte Carlo permutation test to determine the statistical significance given a predefined significance level [5].

Unfortunately, Data Polygamy ignores the problem of multiple comparisons and therefore its method is only sensible for a *single* compared relationship. Suppose there are two datasets of 5 attributes each, resulting in 25 pairwise relationships to test. With a significance level of 0.05, on average at least about one such relationships would pass the significance criteria even on random input. However, data variety, on the other hand, is increasing quickly. The NYC Urban data collection has 228 features on weather monitoring, and over 1,300 data sets in the span of two years have been collected by the government agencies in NYC [5] [13]. Thus recommendation systems without controlling for multiple comparisons are not suitable for real-world datasets.

We downloaded the code of Data Polygamy and studied the number of false discoveries over random data with randomly introduced extreme data points, as summarized in Figure 4. Each extreme data point was sampled independently with 20% probability from a distinct uniform distribution than the normal data. With 100 records and 11 attributes per dataset, Data Polygamy found a total of 43 "bogus" relationships in 50 independent trials. Thus, without considering the risk of multiple comparisons, *Data "Polygamy can be bad for you"; it is literally an automatic p-hacking system.*

2.4 Automatic Model Finding

Finally, systems for automatic model building and tuning in data mining or machine learning (e.g. MLBase [24]) are also victim of the risk of multiple comparisons. To demonstrate the complexity of this problem, suppose that we evaluate a sequence of 20 possible models $\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_{20}$ for our observed data. We test each model using cross validation on different holdout sets, and accept a model if its estimation (prediction) error satisfies our requirement with significance level ≤ 0.05 (i.e., the probability that the model achieved that smaller level of estimation error on a random data is bounded by 0.05). However, this also implies that at least one such model on average would pass our criteria even on random data.

3. QUDE: A SYSTEM TO QUANTIFY THE UNCERTAINTY IN DATA EXPLORATION

As discussed in the previous section, the multi-hypothesis pitfall is a core problem affecting many recent systems for interactive data exploration, recommendations for visualizations and insights, as well as, automatic model building. With QUDE (pronounced "*cute*") we are building the first system to automatically **Q**uantify the **U**ncertainty in **D**ata **E**xploration. QUDE is part of Brown's Interactive Data Exploration Stack (BIDES) and consist of a risk assessment engine as well as a user facing component integrated into Vizdom, BIDES' user interface. While the main focus of QUDE is on the control of the risk of false discoveries due to the testing of multiple hypotheses, QUDE will also be able to detect other risk factors as explained at the end of this section.

3.1 Controlling the Exploration Risk

When a user is exploring a larger number of hypotheses based on the data, either explicitly, indirectly through visualizations, or automatically through recommendation engines, there is a growing risk of flagging a random (i.e., non "*statistically significant*") fluctuation in the data as a significant discovery. Any sustainable data exploration system should therefore effectively control the risk of such "*false discoveries*".

3.1.1 Multi-Hypothesis Evaluation

The risk of false discovery is known as the problem of multiple comparisons, or multi-hypothesis evaluation Two main fundamental challenges arise when attempting to automatically quantify the risk: (1) the traditional techniques do either not scale well with the number of hypothesis or can not be used in an interactive environment and (2) in many cases it is not clear which hypothesis is currently being tested through a visualization by the user (i.e., the "*user intent*"). In the following, we describe various multiple-hypothesis control techniques and how well they work to address the first challenge, whereas in Section 3.1.2 we discuss how we plan to address the user intent challenge.

Family Wise Error Rate (FWER): Traditionally, frequentist methods for multiple hypothesis testing focus on correcting for modest numbers of comparisons. A natural generalization of the significance level to multi-hypothesis testing is the *Family Wise Error Rate (FWER)*, which is the probability of incurring at least one Type I error (i.e., false positive: the null hypothesis is true, but is rejected) in any of the individual tests. The Bonferroni correction [3] was proposed to control FWER for m hypothesis tests at an upper bound α. The Bonferroni correction tests each null hypothesis with significance level α/m. However, this method is too conservative in that the power of the test is too low (i.e., the accepted significance level becomes extremely small) when m is large, resulting in many false negatives. Several methods have been proposed to improve the

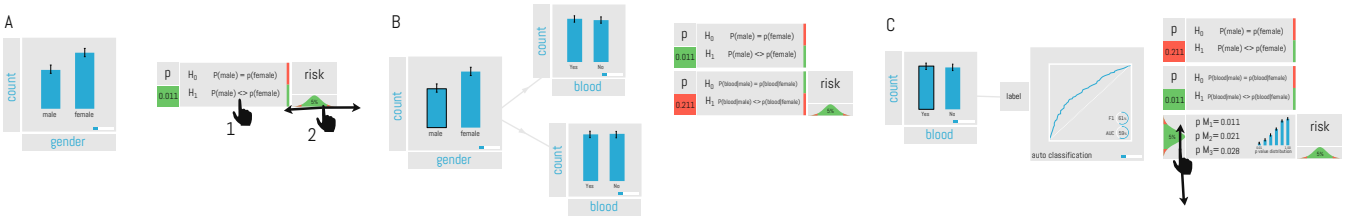


Figure 5: Storyboard of how a "risk controller" could look like.

average power of the Bonferroni method for small to modest m; but for large number of hypotheses, all of these techniques lead to tests with low power. A review of these techniques is given in [32].

False Discovery Rate (FDR): The *False Discovery Rate (FDR)* was introduced by Benjamini and Hochberg [1] as an alternative and less conservative approach to control errors in multiple hypothesis tests. Let V be the number of Type I errors in the individual tests, and let R be the total number of null hypotheses rejected by the multiple test. FDR is defined as the expected ratio of erroneous rejections among all rejections, namely FDR $= E[V/R]$, with $V/R = 0$ when $R = 0$. Designing a statistical test that controls FDR is not simple as the FDR is a function of two random variables that depend both on the set of null hypotheses and the set of alternative hypotheses. Building on the work in [1], Benjamini and Yekutieli [2] developed a general technique for controlling the FDR in multi-hypothesis tests. Furthermore, for entirely random data, FDR controls the same error rate as FWER. That is, FDR and FWER result in the same expected number of mistakes over random data. This property makes FDR easy to explain to users (though, admittedly understanding the differences between FWER and FDR is harder in the presence of real correlations). Most importantly, recent work by Foster and Stine [11] allow to incrementally run the hypothesis tests and thus provide a starting point for controlling the risk in interactive data exploration.

Uniform Convergence and (Structural) Risk Minimization: The uniform convergence paradigm is often used to control the risk in model selection (i.e., machine learning) and is a great candidate technique to control the risk for automatic recommendations (e.g., visualizations as in SeeDB or correlations as in Data Polygamy) and model tuning (e.g., as in MLBase). In this approach the complexity of the predefined class of all possible hypotheses under consideration is analyzed, and based on this complexity it is possible to compute an upper bound to the sample size that is sufficiently large to simultaneously evaluate the expected error of all hypotheses in the class. The approach was first proposed by [33] as the theoretical foundation for statistical learning, but it has been shown to provide practical solutions to some important data analysis problems [29, 30, 31]. The method of structural risk minimization prioritizes less complex models by assigning weights to different hypothesis classes (model) corresponding to the user's preferences. We say that a set of functions has the uniform convergence property if we can use one finite sample to estimate the expectation of all the functions in the set, with a uniform bound on the gap between the empirical mean and the true expectation that hold simultaneously over all the functions in the set. Formally, a set of functions $\mathcal{F}$ has the *uniform convergence* property with respect to a domain Z if there is a function $m(\epsilon, \delta)$ such that for any $\epsilon, \delta > 0$, $m(\epsilon, \delta) < \infty$, and for any distribution D on Z, a sample $z_1, \ldots, z_m$ of size $m = m(\epsilon, \delta)$, drawn independently & identically distributed (i.i.d.) from D satisfies

$$\Pr(\sup_{f \in \mathcal{F}} |\frac{1}{m} \sum_{i=1}^{m} f(z_i) - E_{\mathcal{D}}[f]| \leq \epsilon) \geq 1 - \delta.$$

Other Approaches: An alternative to the previous methods is a *hold-out dataset*, i.e., randomly dividing the dataset into an exploration D_1 and a validation D_2 dataset [37]. While a feasible approach for very large datasets, it can be shown that this approach significantly lowers the power (i.e., the chance of finding a real insight) especially for smaller datasets or subsets of the data. For example, if the user tries to find what distinguishes her top 100 customers from the rest, this method leads to significantly more false negatives. *Permutation tests* [14] can also be used to achieve similar control for multi-hypothesis testing. While well suited for small datasets, permutation tests on large datasets are usually too computationally intensive to be executed interactively.

3.1.2 *Automatic Risk Control in QUDE*

Our goal is to provide the user with accurate risk estimates for different types of interactions in data exploration. Our work as in QUDE is built upon the line of research on efficient FDR bounds for massive data explorations and the application of uniform convergence [23, 29, 30]. The core idea of QUDE is to assume a standard null hypothesis for any exploration the user performs, while allowing the user to customize the null hypothesis with her domain-specific prior knowledge. We are therefore currently developing a heuristic based on our user study to determine the intention of the user. For example, when the user observes that there are an equal number of men and women in the database, but the distribution of men and women is unbalanced when considering only individuals with income over 50k, QUDE assumes the user executes a test to evaluate the significance of this difference. At the same time, QUDE's interface presents this standard hypothesis to the user and allows to overwrite the null hypothesis, if the user chooses to adjust the null hypothesis. Our assumption is that it is not crucial to be always correct with the default hypothesis, but rather that the default hypothesis is used to actively make the user aware that the insight he might have gotten should be tested for significance and to actively seek the user's feedback. Then, as the user continues exploring the data set, the system continuously calculates the risk of false discovery based on the FDR method. If a shown difference is not significant, the user is automatically warned about it. Furthermore, if the user trains a model or uses a visualization recommendation, we apply the same principle by providing an upper bound on the expected number of false recommendations using Uniform Convergence and (Structural) Risk Minimization.

While with our current QUDE we are already able to quantify the risk factors for simple workflows, we also discovered several challenges which we still need to address. Most importantly, for each interaction we need to identify the appropriate null hypothesis to compute the corresponding p-value. Besides using a default hypothesis and allowing the user to overwrite it, we plan to explore alternative approaches such as (1) asking the user what information she is looking for; (2) learning from past action of users on similar data; and (3) apply a precalculated upper bound on the number of different hypothesis answered by a given chart (otherwise a histogram with 20 bars would create over 190 tests).

Similarly, we found that current FDR methods are still not well suited for the iterative process of data exploration. The standard FDR control methods evaluate all the null hypothesis and select a subset to reject. In the iterative process instead we want a stopping criteria that depends only on the actual hypothesis evaluated by the user. There has been some preliminary work towards this direction [15, 11]. Our work builds upon [11] to provide incremental and interactive risk control of data exploration.

A last challenge is to identify classes of hypothesis (e.g., for recommending certain visualization) for which we can compute practical and efficient bounds on their sample complexity using structural risk techniques, as in our recent work [29, 30, 31] for controlling the risk in frequent itemsets mining. The idea is to group sets of primitive hypothesis into classes, and evaluate the total error with respect of the number of different classes used by the users.

3.1.3 QUDE User Interface

Integrating the user feedback in the data exploration process is a key factor towards avoiding false discoveries: (1) the system needs to understand the user intents to better quantify the risk and (2) the system needs to adequately warn the user about potential risk factors so that the user understands the risk of her actions. The core idea of our approach is to use an automatically derived default null hypothesis in order to obtain user feedback and (potentially) as a pessimistic lower bound. Figure 5 shows a storyboard of QUDE's way to to convey the risk factors to the users. In this example, **(A)** Eve drags out a histogram and the system immediately displays a"*risk controller*" on the left-hand side where the results of a default hypothesis test for this histogram are displayed: rejected null hypotheses highlighted in green, accepted null hypotheses highlighted in red. It confirms what Eve intuitively observed from the visualization: there seems to be a significant difference between the number of female and male patients in this dataset. (1) Tapping on this default hypothesis allows Eve to manual adjust and correct what's being tested (e.g., she might want to change from two-sided to one-sided). (2) Eve can also use the "*risk*" slider to change the amount of false discoveries she is willing to accept. **(B)** Eve adds more histograms and connects them to the previous one and the system again automatically runs the appropriate hypothesis tests. Again this confirms what Eve sees visually, the chances of having a blood disease does not seem to be dependent on the gender of a person. **(C)** Afterwards, Eve decides to build a classifier that predicts if someone has a blood disease. Our system's risk controller automatically adds an entry in the risk list, which allows Eve to define a false discovery rate budget for the model search and which is then, for example, controlled with the structural risk minimization technique. Furthermore, Eve can see the p-values of the top 3 models as well as the distribution of p-values for all models that have been tested.

3.2 Detecting Common Statistical Pitfalls

While our main goal of QUDE is to control the multiple hypothesis error, we are also planning to implement tools to detect other common statistical errors/mistakes. In the following, we discuss some of these pitfalls and initial ideas of how to make users aware of them.

3.2.1 Simpson's Paradox

A well-known phenomenon in statistics is the Simpson's Paradox [22] in which a trend reverts when splitting a data set into multiple subgroups. One of the most famous examples is the gender bias among graduate school admissions to University of California, Berkeley. The overall admission figures of 1973 showed that men were more likely to be admitted than women. However, when looking at the largest departments the trend actually reverted for the majority of departments.

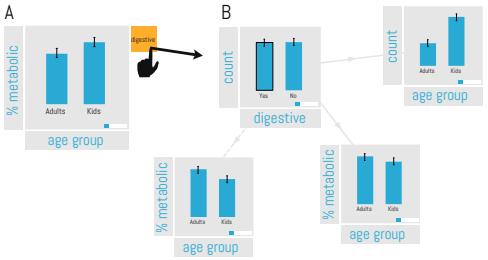


Figure 6: Storyboard of a Simpson's Paradox Warning

For QUDE, we therefore integrated algorithms that detect a Simpson's Paradox online as the user explores a data set. Figure 6 shows a storyboard of how a exploration session of QUDE looks like: Eve already has filtered down her dataset to look at patients from a particular demographic and with certain types of blood testing result values. Eve is now interested to see percentages of such patients that have blood diseases grouped by age groups (kids and adults). From looking at the histogram in **(A)** it seems like kids are more prone for these types of diseases. Afterwards, Eve however notices that the system displays a warning (yellow box). **(B)** By dragging out the warning, the system presents a set of visualizations showing that when accounting for the lurking variable "*digestive disease*" the trend reverses (i.e., kids are less prone for blood diseases for both, with digestive diseases and without).

Testing a dataset for the Simpson's Paradox is quite challenging as it requires to test many different attribute combinations while controlling the risk of finding a Simpson's Paradox by chance. For dealing with both these issues, we are currently developing techniques based on novel index structures and the FDR method.

3.2.2 Other Hypothesis Testing Issues

So far we only focused our attention on Type I error on homogeneous data. However, the Type II error can be as important and (if possible) should be quantified as well. Furthermore, many test can fail on non-homogeneous data and ideally, QUDE should warn the user in those cases or suggest different types of tests. In the context of ML, similar issues, such as the *Base Rate Fallacy* or the *Imbalance of Labels*, can significantly disturb the result if the system/user does not control for it. Similar, *Pseudoreplication*, a very common problem with data collected in life-sciences, may lead to detect a false statistical significance. While it is not possible to automatically detect all of these issues as they might depend on the semantics of the data itself, it might be possible to derive some of the issues automatically based on the schema, analyzing the general data statistics (e.g., for the base rate fallacy), or testing for correlations.

3.3 Data Quality Issues

Finally, many issues can also arise from dirty data in form of missing, duplicate, or inconsistent records. Unfortunately, all data cleaning techniques are expensive, in both time and money (e.g., to pay humans to correct errors) [26], are often not adequate for interactive data exploration, and in almost all cases it is unrealistic to assume that a data-set is perfectly cleaned upfront.

3.3.1 Estimating Remaining Errors

For data exploration it is often more important to understand how many errors a dataset contains and whether these errors are systematic or random rather than trying to correct all errors. This would allows an analyst greater insight on the both the data set and the potential risk factors. While a simple question at first, it

is actually extremely challenging to define data quality without knowing the ground truth [28, 4, 9, 10, 20].

A naïve approach would be to "perfectly" clean a small sample as the gold-standard data (as in [35]) and extrapolate the insight of the cleaning process to the entire data set. For example, if we found 10 new errors in a sample of 1000 records out of 1M records, we would assume that the total data set contains 10000 total errors. A very small sample of cleaned data may however not be representative of the entire dataset. Further, how can the analyst determinate whether the sample itself is actually perfectly clean without a quality metric? As part of QUDE, we have therefore started to develop alternative methods to the naïve estimator, which consider the entire data set – albeit when it is imperfectly cleaned. Our core insight is that almost any cleaning technique has diminishing returns, that is, every additional error is more difficult to detect.

3.3.2 Automatically Repairing Errors

As a second step, we are exploring the possibility of using our insight for the remaining errors in order to automatically correct query answers and models. For example, in previous work [6] we developed and analyzed techniques to estimate the impact of the missing data (a.k.a., "*unknown unknowns*") on simple aggregate queries. The key idea is that the overlap between different data sources enables us to estimate the number and values of the missing data items. Our main techniques are parameter-free and do not assume prior knowledge about the distribution. For future work, we plan to develop similar techniques to correct for a broader range of analytical queries and to learn repair procedures for other errors based on the history of user interactions as well as data characteristics that can either be applied automatically or simply suggested to the user during exploration.

3.4 Current State of QUDE

QUDE currently performs risk evaluations using default hypotheses for simple workflows. We implemented QUDE as part of Vizdom and currently evaluate different types of user feedback and warnings as outlined in our storyboards. We also already developed techniques for quantifying the impact of the unknown unknowns [6], currently evaluate the data quality metrics with several real world use cases, and developed new approximation algorithms for detecting the Simpson's Paradox. However, by no means do we claim that we solved all open issues. Rather, we believe that QUDE is just a first step towards a potential new research area focused on the control of various risk factors in all type of analytics.

4. CONCLUSION

We demonstrated that recent recommendation systems such as SeeDB [34] and Data Polygamy [5] significantly increase the risk of making false discoveries. We further presented our vision and initial ideas for QUDE, a system for automatically controlling the various risk factors in interactive data exploration, automatic model building, and insight recommendation. The goal of this work is, on one hand, to point out that the risk of false discoveries can not be ignored, and on the other, to outline possible solutions with the hope to foster a new line of research around tools for sustainable insights.

5. ACKNOWLEDGEMENTS

This research is funded in part by NSF grant IIS-1562657, NSF grant IIS-1514491, NSF grant IIS-1453171, NSF grant IIS-1247581, NIH grant R01-CA180776, the Intel Science and Technology Center for Big Data, and gifts from SAP, Oracle, Google, and Mellanox.

6. REFERENCES

[1] Y. Benjamini et al. Controlling the false discovery rate. *Journal of the Royal Statistical Society, Series B*, 57(5), 1995.

[2] Y. Benjamini et al. The control of the false discovery rate in multiple testing under dependency. *Ann. Statist.*, 29(4), 08 2001.

[3] C. E. Bonferroni. *Teoria statistica delle classi e calcolo delle probabilita*. Libreria internazionale Seeber, 1936.

[4] Y. Cheah et al. Provenance quality assessment methodology and framework. *J. Data and Information Quality*, 5(3):9:1–9:20, 2015.

[5] F. Chirigati et al. Data polygamy: The many-many relationships among urban spatio-temporal data sets. In *SIGMOD*, 2016.

[6] Y. Chung, M. L. Mortensen, C. Binnig, and T. Kraska. Estimating the impact of unknown unknowns on aggregate query results. In *SIGMOD '16*, pages 861–876, New York, NY, USA, 2016. ACM.

[7] A. Crotty et al. Vizdom: Interactive analytics through pen and touch. *PVLDB*, 8(12), 2015.

[8] A. Crotty et al. The case for interactive data exploration accelerators (ideas). In *HILDA@SIGMOD*, page 11, 2016.

[9] A. Even et al. Dual assessment of data quality in customer databases. *J. Data and Information Quality*, 1(3), 2009.

[10] W. Fan et al. Discovering conditional functional dependencies. *IEEE Trans. Knowl. Data Eng.*, 23(5):683–698, 2011.

[11] D. P. Foster and R. A. Stine. α-investing: a procedure for sequential control of expected false discoveries. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(2):429–444, 2008.

[12] T. R. Foundation. The r project for statistical computing. https://www.r-project.org/.

[13] J. Freire et al. Exploring what not to clean in urban data: A study using new york city taxi trips. *IEEE Data Eng. Bull.*, 39(2), 2016.

[14] P. Good. *Permutation tests: a practical guide to resampling methods for testing hypotheses*. Springer Science & Business Media, 2013.

[15] M. G. G'Sell et al. Sequential selection procedures and false discovery rate control. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(2), 2016.

[16] P. Hanrahan. Analytic database technologies for a new kind of user: the data enthusiast. In *SIGMOD*, 2012.

[17] S. Idreos et al. Overview of data exploration techniques. In *SIGMOD*, 2015.

[18] A. Inc. Amazon mechanical turk. https://www.mturk.com.

[19] J. P. A. Ioannidis. Why most published research findings are false. *Plos Med*, 2(8), 2005.

[20] K. Keeton et al. Do you know your iq?: a research agenda for information quality in systems. *SIGMETRICS Performance Evaluation Review*, 37(3):26–31, 2009.

[21] A. Key et al. Vizdeck: self-organizing dashboards for visual analytics. In *SIGMOD*, 2012.

[22] R. Kievit et al. Simpson's paradox in psychological science: a practical guide. *Frontiers in psychology*, 4, 2013.

[23] A. Kirsch, M. Mitzenmacher, A. Pietracaprina, G. Pucci, E. Upfal, and F. Vandin. An efficient rigorous approach for identifying statistically significant frequent itemsets. *Journal of the ACM (JACM)*, 59(3):12, 2012.

[24] T. Kraska et al. Mlbase: A distributed machine-learning system. In *CIDR*, 2013.

[25] Z. Liu et al. The Effects of Interactive Latency on Exploratory Visual Analysis. *Visualization and Computer Graphics, IEEE Transactions on*, 20(12), 2014.

[26] A. Marcus et al. Crowdsourced data management: Industry and academic perspectives. *Foundations and Trends in Databases*, 6(1-2):1–161, 2015.

[27] D. T. Nhon et al. Transforming scagnostics to reveal hidden features. *IEEE Trans. Vis. Comput. Graph.*, 20(12), 2014.

[28] L. Pipino et al. Data quality assessment. *Commun. ACM*, 45(4):211–218, 2002.

[29] M. Riondato et al. Efficient discovery of association rules and frequent itemsets through sampling with tight performance guarantees. *TKDD*, 8(4), 2014.

[30] M. Riondato et al. Mining frequent itemsets through progressive sampling with rademacher averages. In *KDD*, 2015.

[31] M. Riondato et al. ABRA: approximating betweenness centrality in static and dynamic graphs with rademacher averages. *CoRR*, abs/1602.05866, 2016.

[32] J. P. Shaffer. Multiple hypothesis testing. *Annual review of psychology*, 46, 1995.

[33] V. Vapnik et al. On the uniform convergence of relative frequencies of events to their probabilities. *Theory Probab. Appl.*, 16(2), 1971.

[34] M. Vartak et al. SEEDB: efficient data-driven visualization recommendations to support visual analytics. *PVLDB*, 8(13), 2015.

[35] J. Wang et al. A sample-and-clean framework for fast and accurate query processing on dirty data. In *International Conference on Management of Data, SIGMOD 2014, Snowbird, UT, USA, June 22-27, 2014*, pages 469–480, 2014.

[36] K. Wongsuphasawat et al. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Trans. Vis. Comput. Graph.*, 22(1), 2016.

[37] A. F. Zuur, E. N. Ieno, and C. S. Elphick. A protocol for data exploration to avoid common statistical problems. *Methods in Ecology and Evolution*, 1(1):3–14, 2010.