Contextual anomalies in medical data

Conference Paper · June 2013

DOI: 10.1109/CBMS.2013.6627869

3 authors, including:

Pedro Pereira Rodrigues
University of Porto
96 PUBLICATIONS **1,052** CITATIONS
SEE PROFILE

João Gama
University of Porto
325 PUBLICATIONS **4,000** CITATIONS
SEE PROFILE

Contextual Anomalies in Medical Data

Daniela Vasco
LIAAD-INESC TEC,
University of Porto
baia.daniela@gmail.com

Pedro Pereira Rodrigues
LIAAD-INESC TEC,
University of Porto
pprodrigues@med.up.pt

João Gama
LIAAD-INESC TEC,
University of Porto
jgama@fep.up.pt

1. Introduction

Anomalies in data can cause a lot of problems in the data analysis processes. Thus, it is necessary to improve data quality by detecting and eliminating errors and inconsistencies in the data, known as the data cleaning process [1]. Since detection and correction of anomalies requires detailed domain knowledge, the involvement of experts in the field is essential to the success of the process of cleaning the data. However, considering the size of data to be processed, this process should be as automatic as possible so as to minimize the time spent [1].

2. Related Work

The application of statistical methods in data cleaning, to analyze the data quality as well as to correct deficiencies is very common [2]. These methods are often used but they become a rather basic given that there are many other tools that are more effective and efficient. More sophisticated methods incudes: AJAX, a flexible and extensible framework which the main purpose is to transform the existing data of one or more datasets in a targeted scheme and eliminate duplicates within this process [3]; FraQL [4] a declarative language that supports the specification of a data cleansing process. It eliminates duplicates, fills in missing values and eliminates invalid tuples through the detection and removal of outliers and noise in the data; Potter's Wheel [5] an interactive data cleaning system that integrates the data transformation and error detection using a spreadsheet-like interface. This system considers syntactical anomalies and violation of domain restrictions.

Our work is based on GritBot [6]. Gritbot is a commercial tool that detects inconsistencies in the data set as a precursor to data mining. Although there is no technical description of the methods used by GritBot, we can guess from the results and studies published by Quinlan that the GriBot generates various rules and in each iteration, considers an attribute from a subset of n attributes as an objective (dependent) attribute. Then, each tuple that violates a certain rule is assigned the probability that the value anomaly can occur by chance and not by error.

3. Case Study

Of all the tools presented in the previous section, the GritBot is the one that divides the data into subgroups, this tool seems to be quite interesting when applied to medical data. To illustrate the operation of this tool, the GritBot was applied to a data set representing all cases where there is at least one diagnostic code for disorders of the heart valves, either primary or secondary. The data was collected from nationwide admissions between 1993 and 2009, resulting in 160,853 observations, including information for 63 different variables. As dependent variable we considered the fact that vascular disease was or was not the main diagnosis associated with each admission, and 491 rules were obtained that identify different types of anomalies. In order to facilitate the interpretation of rules, the diagnosis-related group (DRG) is a system to classify hospital cases into homogeneous diagnosis group of each admission, from which are, for example, defined the payments made to the hospital. These codes can be generally clustered into medical or surgical type, thus variable GDHTIPO encodes the corresponding type of DRG (M: medical & C: surgical).

Some of the obtained rules consider the whole data set, as exemplified in the example below:

```
case 37937: (label -259)  [0.001]
        CLTOTDIAS = -257 (156084 cases, mean 9,
        99.99% >= -20)
```

The rule above tells that to the observation 37937, the variable CLTOTDIAS takes the value -257, which is not compatible with the rest of the data, since the 156,084 cases in which this variable has a value, the average is 9 and 99.99% of the cases have a value greater than or equal to -20. This observation is considered an anomaly because it considers the total days of hospitalization as negative.

Other rules consider subsets of the data set. These subgroups may contain one or more variables with exact values or range of values.

978-1-4799-1053-3/13/$31.00 ©2013 IEEE

In the following rules, only one variable is considered:

```
case 129252: (label 0) [0.000]
        GDHTIPO = M (4807 cases, 99.98% `C')
                SRG1 = 3522
```

At the observation 129252, the variable GDHTIPO takes the value "M" which is a value of its domain, however, when the variable SRG1 (Procedures) takes the value 3522, which occurred in 4807 cases, 99.98% of observations had the value "C" for variable GDHTIPO. In medical terms, the rule means that Procedure 3522 in GDHTIPO Medical is an anomaly.

```
case 58386: (label 0)  [0.004]
        GDHTIPO = M  (1110 cases, 99.73% `C')
            ADMTIP = 6
```

The interpretation of this rule is: admissions for additional production of surgery encoded with medical DRG. A possible explanation is that the patient did not actually had the surgery, for some reason, hence requiring coding with medical DRG. This is considered a possible anomaly because in 99.73% of the cases which has admissions for production of additional surgery they are encoded with clinical DRG.

Also rules with two, three or four variables were obtained.

The variables may also be defined by a set of values. In the rules below three possible types of these rules are exemplified:

```
case 85036: (label 0)  [0.002]
        DDXBin = no (1154 cases, 99.83% `yes')
            CLIDADAN > 81 [85]
            ADMTIP = 2
            DRG = 135
```

The medical interpretation of this rule is: Patient with more than 81 years, not scheduled admission, coded with DRG 135, was not encoded as the main diagnostic for a valvular heart disease.

```
case 34461: (label 0)  [0.011]
        DDXBin = yes  (1847 cases, 99.73% `no')
            ADMTIP = 2
            CLTOTDIAS <= 9 [6]
            DRG = 122
```

This rule means that: Not scheduled for admission, inpatient less than 9 days and DRG 122 with a valvular disease as main diagnostic is considered a possible anomaly.

The last rule we presented here consider a observation 13526 as a possible anomaly because in this case, admission is not scheduled and a patient aged over 59 years is hospitalized for more than 5 days and DRG 135 has no valvular disease as the main diagnostic.

```
case 13526: (label 0)  [0.013]
        DDXBin = no  (2862 cases, 99.06% `yes')
            CLIDADAN > 59 [74]
            ADMTIP = 2
            CLTOTDIAS > 5 [36]
            DRG = 135
```

As we can observe, by the interpretation of the rules, they can be a great asset not only for the detection of anomalies, but also in identifying rare cases.

4. Conclusions

In this work we use the data cleaning software GritBot, designed to identify anomalous cases in data. The most relevant characteristic of GritBot is the ability to identify the contexts where some of the attribute-values are anomalous. The contexts are rules that specify regions in the instance space. Doing so, Gritbot identify local anomalies that are observations with suspicious attribute-values in a particular region of the instance space. We should point out that these anomalies might not be suspicious at global level. it was realized how much this tool distanced itself from other tools studied for its ability to detect potential anomalies in subsets of data instead of the usual strategy of finding anomalies over all the data. This is the main advantage of GritBot. However, it cannot guarantee to find all anomalies in the dataset and the reported cases are just possible anomalies. Therefore is very important a domain expert to analyze the results. In the case reported here, the anomalies pointed out where confirmed by the expert. The main advantage of this software is the ability to define the context where and why the anomaly occurs.

Acknowledgments: The authors acknowledge ACSS and Alberto Freitas for data availability and coding interpretation and financial support of ERDF through the COMPETE Programme and FCT by the project FCOMP - 01-0124-FEDER-022701 and KDUS (PTDC/EIA/098355/2008).

References

[1] Müller, H., & Freytag, J.-C. (2003). Problems, Methods and Challenges in Comprehensive Data Cleaning *Technical Report HUB-IB-164*. Berlin, Germany: Humboldt University.

[2] Maletic, J. I., & Marcus, A. (2000). *Data Cleaning: Beyond Integrity Analysis.* Conference in Information Quality, Cambridge, MA, USA.

[3] Gallhardas, H., Florescu, D., & Shasha, D. (2000). AJAX: An extensive data cleaning tool. *ACM SIGMOD on Management of Data.* Dallas, TX, USA.

[4] Sattler, K.-U., Conrad, S., & Saake, G. (2000). *Adding Conflict Resolution Features to a Query Language for Database Federations.* Proceedings 3rd International Workshop on Engeneering Federated Information Systems, Dublin, Irland.

[5] Raman, V., & Hellerstein, J. M. (2001). *Potter's Wheel: An Interactive Data Cleaning System.* 27th Very Large Data Bases, Roma, Italy.

[6] Quinlan, R. (2007). GritBot:An Informal Tutorial, from http://www.rulequest.com/gritbot-unix.html