WizSoft®

Data Mining for Forensic Investigators

Abraham Meidan, Ph.D.

One of the main tasks of auditors, forensic investigators and data-quality managers is revealing fraudulent cases and errors in data. *WizRule* can help in carrying out this task. *WizRule* is a data-auditing tool based on data mining technology. It performs an analysis of the data revealing inconsistencies and "strange" cases to be investigated.

The standard method for revealing fraudulent cases and errors uses reports that filter and sort the data. For example, one such report may list all the transactions in which the discount percentage is above a certain threshold. There are several tools that can be used in order to issue these kinds of reports.

WizRule does not compete with these tools. Rather it complements them.

By definition the above-mentioned reports can only reveal frauds and errors that the report was designed to find. For example, if the investigators suspect that some transactions include fraudulent discounts, they may issue a report that list the transactions having the highest discount percentage. But if they don't suspect a problem related to the discount, the reports issued by the standard auditing tools will not discover a fraudulent discount. And since there are an enormous number of possible frauds it is impractical to generate a report for each of them.

This is where *WizRule* can help. *WizRule* works automatically – the user just selects the data and *WizRule* does the analysis. *WizRule* checks all the relationships among the values within the various fields and reports unexpected and unlikely cases. Therefore, *WizRule* reveals fraudulent cases missed by the standard auditing tools.

How does *WizRule* work?

WizRule is a data mining tool. Data mining programs reveal interesting patterns in data.

Usually data mining tools reveal the patterns in data registered in the past and use these patterns to issue predictions for new cases. For example, a bank may apply a data mining program in order to reveal the patterns of customers that did not pay their loans. Then, when a new customer asks for a loan these patterns are used to calculate the probability of default on this loan. This approach is also used for revealing frauds. For example, credit cards companies use data mining programs in order to discover the patterns of known fraudulent cases and apply these patterns when checking new transactions. But this approach cannot be used when one searches for fraudulent cases without having previous examples.

This is where *WizRule* is relevant. Instead of issuing predictions for new cases *WizRule* points out cases deviating from valid patterns. *WizRule*'s approach is based on the following assumption: ***In many cases, frauds are exceptions to the rule.*** For example, if in all sales transactions to a certain customer the salesperson is Dan, and there is a single transaction in which the salesperson is someone else, who is usually connected with other customers, then this is a suspected case that should be investigated.

In creating a software application that discovers exceptions to the rule, the program first needs to discover all the rules (patterns) in a given data set. In other words, the software should do a "reverse engineering" of the rules that created the data. This is precisely *WizRule*'s strong point. *WizRule* is based on a mathematical algorithm that is capable of revealing all the rules governing a data set within a very short span of time. The output of a *WizRule* analysis is a list of records that are unlikely in reference to the discovered rules. These records are *suspected cases,* or at least *cases to be investigated.*

When using *WizRule*, you simply select the data that you wish to analyze and the software does all the rest. Within a short time, the analysis report is displayed on the screen.

When analyzing the data, *WizRule* performs the following operations:

It first reads the data. *WizRule* can read all the standard databases. You are then given the opportunity to "fine-tune" the analysis parameters such as "*minimum probability of if-then rules*" and "*minimum number of cases of a rule.*" You can also define exactly which types of rules *WizRule* should search for and whether some fields should be ignored.

Within a short time, *WizRule* reveals the rules governing the data and points out the cases deviating from the discovered rules. Each deviation is displayed along with the rules from which it deviates.

What kind of rules does *WizRule* reveal?

WizRule analyzes the data by revealing four types of rules:

- Formula rules
- If-then rules
- Outstanding rules
- Spelling rules

As mentioned, the user does *not* enter the rules. Rather all the rules are discovered automatically.

An example of a *formula* rule is:

A = B * C

Where: **A = Total**
B = Quantity
C = Unit Price

Rule's Accuracy Level: **0.99**
The rule exists in **1890** *records*

The "*Accuracy Level*" in formula rules indicates the ratio between the number of cases in which the formula holds and the total number of relevant cases. The cases in which the formula holds are those cases where the formula matches the data exactly except for deviations that may result from a rounding.

WizRule reveals arithmetical formulas with up to 5 variables that hold in the data. To avoid revealing uninteresting formulas and false alarms, formulas where **A** is 0 or 1 are ignored.

Obviously if a formula rule holds for all the records in the data except for just a few records, then these deviating records should be investigated.

An example of an *if-then* rule is:

If **Customer** is **<u>Summit</u>**
and **Item** is **<u>Computer type A</u>**
T*hen*
Price = **<u>765</u>**

Rule's probability: **0.998**
The rule exists in **1002** *records*
Significance level: error probability < 0.001

The "*Probability*" in if-then rules designates the ratio between the number of records in which the condition(s) and the result hold, and the corresponding number of records in which the condition(s) hold with or without the result.

The "*Significance Level*" indicates the degree of the rule's validity. It is equal to 1 minus the "*error probability*", which quantifies the probability that the rule exists accidentally in the data under analysis.

WizRule reveals all the if-then rules with any number of conditions (as determined by the user).

Once again, a deviation from a highly valid rule might point to a fraud.

An *outstanding rule* lists unexpected *rules* (contrary to unexpected *records*). These rules can discover a fraud that occurred many times and as a result created a rule.

Consider the following example:

The program discovered the following rules:

(1) *If* **Customer** is **<u>Summit</u>**
T*hen*
% Discount = **<u>20</u>**

(2) *If* **Customer** is **<u>Dan</u>**
T*hen*
% Discount = **<u>10</u>**

(3) *If* **Customer** is **<u>Marry</u>**
T*hen*
% Discount = **<u>10</u>**

Suppose also that there are many more rules like (2) and (3) where the customer discounts is 10%. In such a case rule (1) is outstanding since it deviates from the other rules: it is the only rule where the discount is 20%, while in all other rules the discount is just 10%.

In other words, the rule is outstanding since another rule was expected. Following the other rules that relate between the Customer and the % Discount fields, one would expect the following rule:

> *If* **Customer** is **<u>Summit</u>**
> T*hen*
> **% Discount** = **<u>10</u>**

But actually, the above-mentioned outstanding rule says that the % discount is 20% (rather than 10%).

An example of a *spelling* rule is:

> *The value* **<u>Summit</u>** *appears* **2080** *times*
> *in the* **Customer** *field.*
> *There are* **2** *case(s) containing similar value(s)*

These rules are presented mainly in order to reveal cases of misspelled names. A name is suspected as misspelled if (a) it is similar to another name in this field, and (b) the frequency of the first name is very low, while the frequency of the second name is very high. For example, if the name **Zummit** appears only one time (in the Customer field) it will be presented as a deviation to be examined.

How does *WizRule* avoid False Alarms?

Following the discovery of the rules that govern the data, *WizRule* checks the deviations from these rules. However, not every deviation from a rule is a case to be examined. Suppose *WizRule* reveals the following *if-then* rule:

> *If* Customer is <u>Summit</u>
> Then
> Salesperson is <u>Dan</u>
> *Rule's probability:* 0.98
> *The rule exists in* 1003 *records*
> Significance level: error probability < 0.001

Since the rule's probability is 0.98 and the rule exists in 1003 records, then there are about 20 records in which the salesperson deviates from this rule. Reviewing each of these 20 records is quite tedious and many of these deviations might be false alarms.

To avoid such false alarms *WizRule* checks whether the deviation is explainable by another rule that holds in the data. If the answer is positive the case is not a suspected one. For example, it may be the case that if the **item** is **computer** then the **salesperson** is **John**, and this rule may explain some of the above-mentioned deviations. Since these deviations are explainable they are not considered ~~as~~ cases to be investigated.

WizRule also checks the frequency of the *then* value in the deviated case. Its frequency under the rule conditions should be lower than its overall frequency in the data. If it does not, then once again the case is not considered as a case to be investigated. For example, if the **salesperson** in two of the deviated records is **Frank** and these are *the only two cases* in the entire data where Frank is the salesperson, then these cases are not considered as cases to be investigated. However, if Frank is usually the salesperson of other customers, then the above-mentioned deviations are *indeed suspected cases.*

When the *then* field is numeric, *WizRule* also lets you reduce false alarms by only displaying deviations where the *then* value deviates from the expected value by at least one standard deviation. Smaller deviations are ignored.

When viewing the suspected cases, you can sort the cases by the *then* field, and sort the values within this field. This sorting is mainly relevant when the report lists many deviations. By sorting the deviations, you can concentrate on the most interesting cases.

Applying all these methods *WizRule* avoids almost all the false alarms and points out only the cases that need to be investigated.