Cortana for Dummies

Marvin Meeng, Wouter Duivesteijn

December 4, 2012

1 What is Cortana, and what does it do?

Cortana is a tool which, given a dataset, performs Supervised Descriptive Local Pattern Mining:

Local Pattern Mining: a run of Cortana strives to find *subsets* of the dataset where *something interesting* is going on. We try to pinpoint local exceptionalities in your dataset.

Descriptive: subsets delivered by Cortana are not just any subsets of the data, but *coherent* subsets, defined on attributes. Such subsets are called *subgroups*.

Supervised: the interestingness is measured with respect to a user-defined *target concept*: a subset of attributes that we are particularly interested in. Cortana finds subgroups for which the target concept is substantially different than the target concept over the entire dataset.

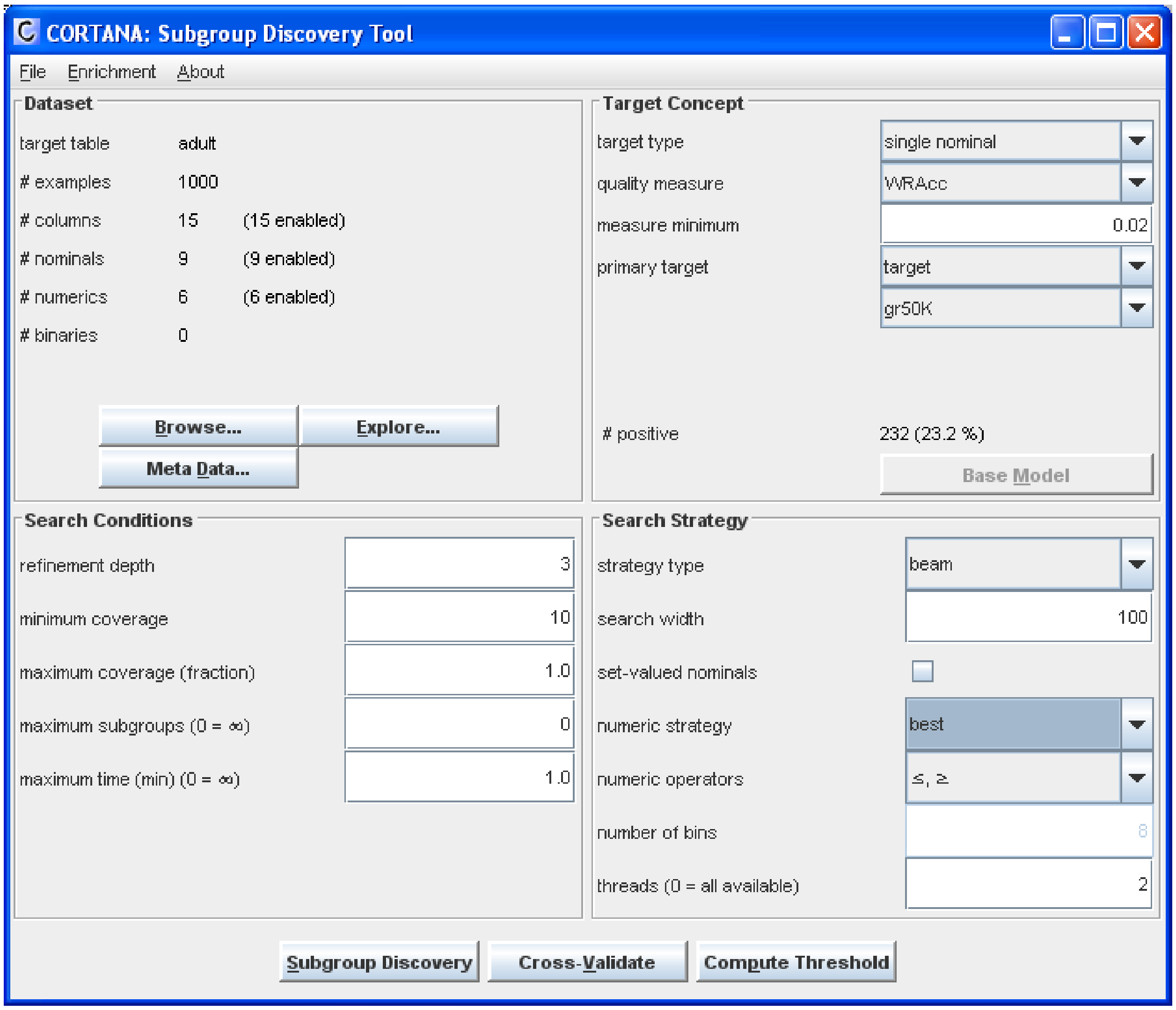
As an example, suppose a dataset about people, and let the target concept be the distribution of just one binary attribute: whether the person develops lung cancer or not. Cortana can find subgroups like *smoking = true* (the smokers, which have a substantially higher incidence of lung cancer), and *weekly walking kilometers* ≥ 50 (athletes, which have a substantially lower incidence of lung cancer). [TODO: might need a better example here]

2 Launching Cortana

After obtaining and unpacking a copy of Cortana from `http://datamining.liacs.nl/cortana.html`, navigate to the Cortana directory where you will find a cortana.jar file. One could start Cortana by double clicking the jar. However, it is recommended to start Cortana from the command line, since this will feed back information information about the Subgroup Discovery process to the user. To start Cortana this way, open a terminal or command window, navigate to the Cortana directory containing cortana.jar, and type: *java -jar cortana.jar*. Alternatively, one can use either cortana.bat (for Windows) or cortana.sh (Bash shell script). Be sure to read Appendix C if your computer displays memory issues while running Cortana.

After starting Cortana you are prompted for the file containing the dataset to be analysed, after which Cortana's main screen is shown.

Cortana can handle two kinds of files: ARFF files and plain (comma seperated) text files. An ARFF file combines the data with definitions of the type of each attribute. For text files, Cortana wil try to automatically infer these types. Unfortunately, this may not always deliver the desired outcome. We will discuss how to correct wrong assessments in Section 3.1.3, but in order to prevent the problem, we recommend the use of ARFF files when this option is available.



Dataset

target table	adult	
# examples	1000	
# columns	15	(15 enabled)
# nominals	9	(9 enabled)
# numerics	6	(6 enabled)
# binaries	0	

Target Concept

target type	single nominal
quality measure	WRAcc
measure minimum	0.02
primary target	target
	gr50K
# positive	232 (23.2 %)

Search Conditions

refinement depth	3
minimum coverage	10
maximum coverage (fraction)	1.0
maximum subgroups (0 = ∞)	0
maximum time (min) (0 = ∞)	1.0

Search Strategy

strategy type	beam
search width	100
set-valued nominals	☐
numeric strategy	best
numeric operators	≤, ≥
number of bins	8
threads (0 = all available)	2

Figure 1: Cortana's main window.

3 Main Screen

The main screen of Cortana is devided into four major panels, *Dataset*, *Target Concept*, *Search Conditions*, and *Search Strategy*.

3.1 Dataset Panel

The dataset panel contains information about the dataset that is currently loaded:

target table shows the name of the data file used. For text files this is the filename, and for ARFF files this is the name defined in the '@relation' field;

# examples shows the number of examples in the dataset;

# columns shows the number of columns in the dataset. Remember that columns are also be referred to as attributes.

Finally, there is a number of fields that indicate the number of attributes from each type: nominal, numeric, and binary. Notice that Cortana cannot handle ordinal data; such attributes can be handled as either nominal or numeric, depending on what kind of subgroups you would want to define on the attribute.

In addition to the fields described above, there are three buttons present on the **Dataset** panel: **Browse. . .** , **Meta Data. . .** and **Explore. . .** .

3.1.1 Browse Window

Clicking the Browse button will open a Browse Window, displaying the data in its current state. You can sort the data by clicking on the column head of an attribute, and save the current state of the data to a file, which is useful if you have made modifications via the Meta Data Window (see Section 3.1.3).

3.1.2 Explore Window

Clicking the Explore button will open an Explore Window, which allows you to examine the distribution of single attributes, and cross-examine two attributes. This section of Cortana is independent of the Local Pattern Mining for which Cortana was developed; it exists merely to allow you to learn more about the makeup of your data.

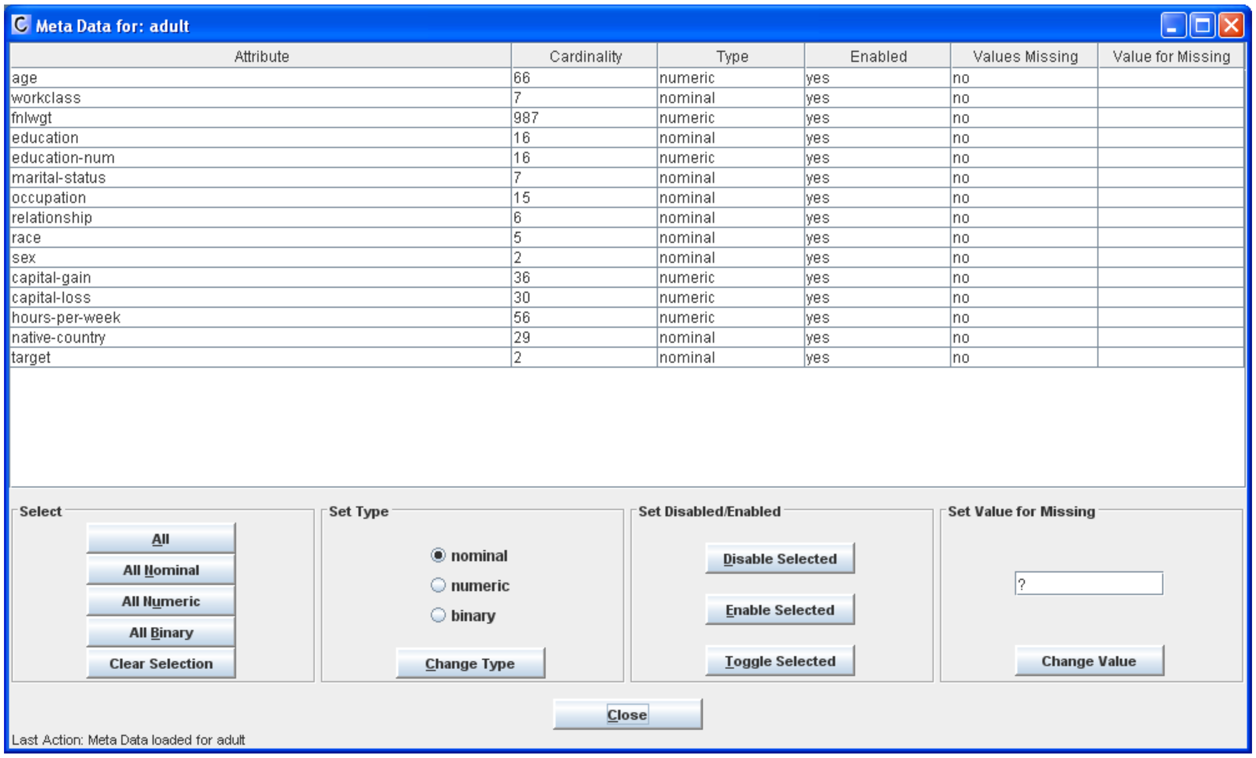
3.1.3 Meta Data Window

Clicking the Meta Data button will open a MetaData window, which displays information on the attributes of the current dataset and allows changing some of the attribute properties. The upper part of the window shows a table with six columns. The lower part contains a number of panels that allow modification of the data as it is in memory. Note that no modifications are made to the original data file. First the properties shown in the table in the upper part will be described, the purpose of the various data manipulations will be explained after that.

Meta Data Table The upper part of the MetaData window displays a table containing the current attribute information. The first column in this table, *Attribute*, lists the attribute names. *Cardiality* gives the number of distinct values for the attribute. *Type* shows its attribute type. *Enabled* indicates whether the attribute is enabled or disabled; this setting affects the search for subgroups, which will be discussed in Section 3.4. *Values Missing* indicates whether values for the attribute are missing in the dataset, and if so, *Value for Missing* shows the value that is currently used for these missing values. If not, this field is blank.

Meta Data Functions The panels in the lower part of the *Meta Data Window* all allow selecting or changing the data. The first, *Select*, allows selecting all attributes of a certain type. This is a convenience method to be used in combination with the functionalities available in other panels.

Set Type allows changing the attribute type. Cortana makes an educated guess which type each attribute should have. Occasionally you will want to change this assessment, and if possible this panel will allow it. Select the attributes in the table in the upper part of the MetaData window, choose the radio button corresponding to the desired type, and click the Change Type button. If the requested action can be



Meta Data for: adult

Attribute	Cardinality	Type	Enabled	Values Missing	Value for Missing
age	66	numeric	yes	no	
workclass	7	nominal	yes	no	
fnlwgt	987	numeric	yes	no	
education	16	nominal	yes	no	
education-num	16	numeric	yes	no	
marital-status	7	nominal	yes	no	
occupation	15	nominal	yes	no	
relationship	6	nominal	yes	no	
race	5	nominal	yes	no	
sex	2	nominal	yes	no	
capital-gain	36	numeric	yes	no	
capital-loss	30	numeric	yes	no	
hours-per-week	56	numeric	yes	no	
native-country	29	nominal	yes	no	
target	2	nominal	yes	no	

Set Type: [✓ nominal] numeric binary

Set Value for Missing: ?

Last Action: Meta Data loaded for adult

Figure 2: Meta data window.

performed, Cortana will do so, and if not (for instance: if you try to make an attribute binary while it has more than two distinct values), Cortana will do nothing.

Set Disabled/Enabled allows to disable or enable an attribute. The relevance of this choice will be discussed in Section 3.4; by default all attributes are enabled.

Set Value for Missing can be used to change the value that is currently used as a proxy for missing occurrences of this attribute in the data. Details can be found in the Full Cortana Manual.

3.2 Target Concept Panel

The settings in the Target Concept panel determine how the Cortana process is *supervised*, as described in Section 1. A Cortana run strives to find subgroups for which a target concept is substantially different than the target concept over the entire dataset. In the Cortana Light Manual, we will focus on the case where this target concept is the distribution of a single attribute. Cortana can also find subgroups for more complex target concepts; these are described in the Full Cortana Manual.

[TODO: edit from here]

The first drop-down menu in the Target Concept panel is **target type**, that is used to change the target type for which one wants to do Subgroup Discovery. Selecting another target type from the drop down box will force Cortana to consider other attributes as target concept. For example, when changing the target type from the default *single nominal* to *single numeric*, the first attribute having the numeric attribute type is selected, and subsequently displayed in the **primary target** drop down box.

Based on the target type selected, a number of quality measures is available. These are listed in the **quality measure** drop down box. After changing the target type, the quality measures listed in the **quality measure** drop down box are automatically updated, to fit the new target type.

The text field next to **measure minimum** will, show a minimum threshold value for the selected quality measure. One might consider lowering this value if a Subgroup Discovery experiment, run with the default value, returned too few subgroups. On the other hand, one can increase this value to force Cortana to only report subgroups that have a sufficiently quality.

The items described in the remainder of this section will not be available in every target type setting, as the various target types are related to the attribute type of the target concept. For each item, it will be mentioned to which attribute type it applies.

primary target is available in the following target type settings: *single nominal*, *single numeric* and *double correlation*. The **primary target** drop down box lists all attributes available in the dataset.

target value is available only in the *single nominal* setting. The **target value** drop down box lists all values of the attribute set as *primary target*, which can be of the *nominal* and *binary* attribute type.

3.3 Search Conditions Panel

The fields on this panel allow setting the search conditions used in the Subgroup Discovery process. For all fields the default values are also given.

refinement depth controls the number of conditions that is used to create subgroups. A refinement depth of 1 would lead to conditions like $x \leq 9.11$, while a refinement depth of 2 would allow for the creation of conditions like $x \leq 1.2 \wedge y \geq 3.4$. It is not recommended to set the refinement depth to very high values. In Subgroup Discovery, often a depth greater than 4 or 5 does not lead to significant improvements of the measure score any more, and just increases both the risk if overfitting, and the computational time needed to calculate the result.

minimum coverage sets the lower bound for the size of the subgroups that should be reported by Cortana, meaning all reported subgroups have at least this size. The default value is set to 10% of the total dataset size, which is shown as **# examples**, see section **??** above.

coverage fraction sets the upper bound for the size of the subgroups that should be reported by Cortana, meaning all reported subgroups have at most this size. The default value is set to 1.0, which indicates the entire dataset.

maximum subgroups is used to control the maximum number of subgroups in the result list generated by Cortana. First, this means that the Result Window (section **??**) will show at most this number of subgroups.

But also, as alluded to in the *Dataset* section (**??** on *measure minimum*, this number also controls the number of intermediate results retained by Cortana to form the pool out of which the generation for the next *refinement depth* is formed. As such, this *Search Condition* parameter has significant influence on the computational demands, as it directly controls the number of combinatorial candidates to be tested during Cortana's Subgroup Discovery process. Without going into the actual algorithm, a small example probably suffices to make this clear. For a dataset consisting of only *nominal* attributes, assume $n =$ *maximum subgroups*, and $m =$ *number of attribute-value pairs*, then the number of candidates c to be tested by Cortana for the next *refinement depth* iteration is: $c = n \cdot m$.

maximum time (min) will determine the maximum time, measured in minutes, Cortana is allowed to search for new subgroups. After this periode Cortana will abort the Subgroup Discovery process and report all subgroups found up until that point. This means that especially for the *search strategy* (section **??**) *strategy type depth first* a large of the search space will not be explored at all, if the search proces is aborted. More precisely, only the first attributes will have been addressed at that time.

3.4 Search Strategy Panel

4 Result Window

Appendices

A Glossary

binary An attribute that can have only two distinct values. These values will be represented in Cortana with 0 and 1. Cortana will form subgroups *attr*=0 and *attr*=1.

disabled An attribute that will not be considered for generating candidate subgroups.

enabled An attribute that will be considered for generating candidate subgroups.

nominal An attribute that can have any countable number of distinct values. No ordering is assumed between these values. Cortana will form subgroups *attr*=*value*, and *attr* $\in$ {*value1*,*value2*,...} when the set-valued nominals checkbox in the Search Strategy Panel is selected (see Section 3.4).

numeric An attribute that can have values from a continuous range. Cortana will form subgroups *attr*$\leq$*value* and *attr*$\geq$*value*. You can control how many and which values will be selected in the Search Strategy Panel (see Section 3.4).

ordinal An attribute that can have any countable number of distinct values. A natural ordering exists between these values. Cortana cannot handle this attribute type. Depending on which kind of subgroups you would want to consider, you can make Cortana handle such an attribute as either nominal or numeric.

type Determines the range of values an attribute can have. In Cortana, an attribute can have types binary, nominal, or numeric. Attributes in Cortana cannot be ordinal; for these attributes you can either select the nominal or numeric type, depending on what kind of subgroups you would like to see.

B Autorun

C Memory Issues

Although Cortana is written in Java, and therefore platform independent, it will behave slightly different on different operation systems and/or platforms. These differences arise from small variations in the Java Virtual Machines, used in different situations. The main issue is with 32-bit operating systems (OS). On such systems the maximum amount of memory the Java Virtual Machine (JVM) can use is around 1600 MegaBytes. However, the actual amount depends on the amount of RAM available. The cortana.bat and cortana.sh file included in the Cortana.zip set the maximum amount of memory the JVM can use to 1600 MegaBytes, through the -Xmx option. The value should, at most, be set to half the amount of available RAM, meaning eg. for a 2GB machine to *-Xmx1000m*. For 64-bit OSes no such limit exists, and it should be save to remove the -Xmx. Note that the above means that, especially for 32-bit OSes, not all datasets will fit into memory.