CREATIVITY THROUGH EFFICIENCY

Automated Predictive Analytics
Quick User's guide **v2.5**

Company headquarters:
Address: Chemin des 2 Villers, 11 - 7812 Ath (V.N.D.) - Belgium
Phone (global): +32 479 99 27 68
Phone (Americas): + 57 300 675 13 69

Creation date: January 2008
Last Edited: October 2021

1. Introduction

Welcome to the Quick user's guide to TIMi Modeler.

This document will guide you through the process of creating and interpreting a predictive model using TIMi Modeler. This guide is designed for beginners, and while the vocabulary used may sometime be a little bit technical, you don't need to be a specialist with 10 years of training in intricate mathematical abstractions to understand this document.

Discovering new insights about your customers should be fun and easy, and the TIMi Suite has been designed with this goal in mind. The TIMi Suite lets you easily explore terabytes of data to extract some useful knowledge. There is a whole new world waiting to be discovered, hidden inside your databases that you can now easily explore with TIMi Modeler.

The "TIMi Suite" consists of four tools: Anatella (TIMi's ETL), Startdust (TIMi's Data Visualization and Segmentation tool), Kibella (unlimited BI) and TIMi Modeler, the fastest automated predictive modeling tool currently available. This document focuses only on TIMi Modeler, please refer to the appropriate tutorial for the other tools.

During the course of this document we will analyze together a dataset named "census-income" (in statistics data-tables are usually named "datasets"). This dataset contains information about the financial characteristics of residents of the United State of America. Here is an extraction of this dataset:

key	**Is taxable income amount above $50K ?**	age	education	marital stat	race	sex	country of birth	weeks worked in year
1	0	73	High school graduate	Widowed	White	F	USA	0
2	0	58	Some college but no degree	Divorced	White	M	USA	52
3	0	18	10th grade	Never married	Asian	F	Vietnam	0
4	0	9	Children	Never married	White	F	USA	0
5	0	10	Children	Never married	White	F	USA	0
6	0	48	Some college but no degree	Married-civilian	Indian	F	USA	52
7	0	42	Bachelors degree(BA AB BS)	Married-civilian	White	M	USA	52
8	1	28	High school graduate	Never married	White	F	USA	30
9	0	47	Some college but no degree	Married-civilian	White	F	USA	52
10	0	34	Some college but no degree	Married-civilian	White	M	USA	52
11	0	8	Children	Never married	White	F	USA	0
13	0	51	Some college but no degree	Married-civilian	White	M	USA	52
14	1	46	High school graduate	Divorced	White	F	Columbia	52
15	0	26	Bachelors degree(BA AB BS)	Never married	White	F	USA	52
16	0	13	Children	Never married	Black	F	USA	0
17	0	47	Bachelors degree(BA AB BS)	Never married	White	F	USA	52
18	0	39	10th grade	Married-civilian	White	F	Mexico	0
19	0	16	10th grade	Never married	White	F	USA	0
20	0	35	High school graduate	Married-civilian	White	M	USA	49

Table 1: Data Struture

During the course of this tutorial, we will explore the relationship between the column "Is taxable income amount above $50K ?" and all the other columns of the dataset (age, education level, race,…) . The column "Is taxable income amount above $50K?" is the "column to explain" inside our dataset. The "column to explain" is named, in technical term, the "Target Column".

As it is often the case in machine learning problem, only a small percentage of this population belongs to the target group, and each individual (each record, or each observation) within this group is named "a target". In terms of data, the "Targets" (i.e. all the people with a taxable income amount above $50K) are identified with a value of '1' in the "Target Column".

Within the "census-income" dataset, the "Target Column" contains only two different values: 0 or 1. This is called a "Binary Target". Note that TIMi Modeler can analyze datasets with three types of targets:

- "Binary Targets",
- "Continuous Targets" and
- "Multi-class Targets".

In this document, we will focus primarily on the census-income dataset, which contains a "Binary Target". However, in section 8 we will extend the notions learned on a "Binary Target" problem to a "Continuous Target" problem. In this section, we will focus on the prediction of the weight of a person using only various body circumference lengths.

The "census-income" dataset contains another special column: the "primary key" column or primary key *variable*. This "primary key" contains a unique value for each line of the dataset. This allows us to uniquely identify each record (i.e. each row or observation) in our dataset. The concept of "primary key" is well known in the database world: Should you require additional information on the topic, we recommend you to read any introductory books on the "data management/data base" topic, or simply ask anyone in your IT department. The "primary key column" in our dataset is named "key", and we recommend (although it's not mandatory) using this convention for automatic type recognition.

Modeler is able to process datasets stored in many formats. These datasets can be stored inside Anatella Gel files (.gel_anatella files ; this is the preferred format), relational databases (like Oracle, Teradata, Microsoft SQL server, Informatix, MySQL,...), or simple "flat files" (text files). The preferred storage format for Modeler is a .gel_anatella file which offer a good compression algorithm and the fastest reading speed (.csv files compressed in RAR are also good).

2. TIMi Modeler Installation

TIMi Modeler is part of the TIMi Suite. To install the TIMi Suite on a machine, the easiest way is to download and run the automated installer available here:

https://timi.eu/downloads/

For more details, you can also refer to the document "TIMi_Deployment.pdf" available here:

http://download.timi.eu/docs/TIMi_Deployment.pdf

Along with the TIMi Suite, there are some other useful third party softwares included in the package:

- **IrFanView:** "TIMi Modeler" generates many charts and graphics that allows you to explore your data in a colourful-user-friendly way. Most of these graphics are PNG files. The "IrFanView" software allows you to rapidly browse through thousands of PNG-graphic-files generated with TIMi. This is a nice complement to the picture viewer integrated inside Windows. Visit http://www.irfanview.com/ for more information.
- **EditPadLite7:** Some reports generated with TIMi can be extremely large files. The "EditPadLite" software is one of the few TEXT editors that is able to open and edit 100 MB unicode text reports. This is very handy when working with dataset files containing more than 10,000 columns because, then, the TIMi-reports can become really huge (more than 80 MB size). An even better alternative to EditPadLite is EmEditor: Visit https://www.emeditor.com/ for more information about EmEditor.

By default, The TIMi Suite installation software also installs some sample datasets ("Census-Income" and other) inside your "{My_documents}\TIMi\DATASETS" folder or inside "c:\soft\TIMi\AnatellaDemo\datasets". Inside this tutorial, we will analyze with TIMi Modeler the dataset "Census-Income".

3. Directory Structure

When working with TIMi Modeler, we suggest to adopt the following directory structure:

- **A data lake: One "Central Dataset Repository" directory**

  The "dataset" directory is a central repository that will contain all the datasets that you will analyze. Dataset files are usually very large files (several gigabytes) and it's better to avoid duplication of these files in various directories on your system. To prevent duplication, we use a central repository that contains the "one and only" copy of our dataset files.

  If you are working on a distributed file system, you'll most certainly have a "central-network-drive" shared by all the analysts. I suggest you to create the dataset repository on the shared drive to prevent duplication of the data. TIMi Modeler has been designed to minimize the load on the network and it will read your data files at the lowest possible frequency. Typically, once the first "read" is completed, the work happens locally which reduces the load from the central server.

  Note that you can easily change this directory by simply selecting another one in the standard Mode Interface.

- **<u>One "working directory" for each different analysis</u>**

  The first thing to do when starting a new analysis with TIMi Modeler is to create a working directory that will contain all the reports and, in general, all the results of the data mining process. The TIMi Suite installation software automatically creates the directory "{My_documents}\TIMi\ INCOME", which will be used as "working directory" during the course of this tutorial.

By default, the "Central Dataset Repository" is the directory:

- C:\soft\TIMiPortable\AnatellaDemo\datasets when using the standard TIMi installer.
- {My_documents}\TIMi\DATASETS when using the Wizard-based off-line installer.

Here is a screenshot of our default demo "Central Dataset" Repository:
(It contains our demo-dataset: "Census-Income.rar")

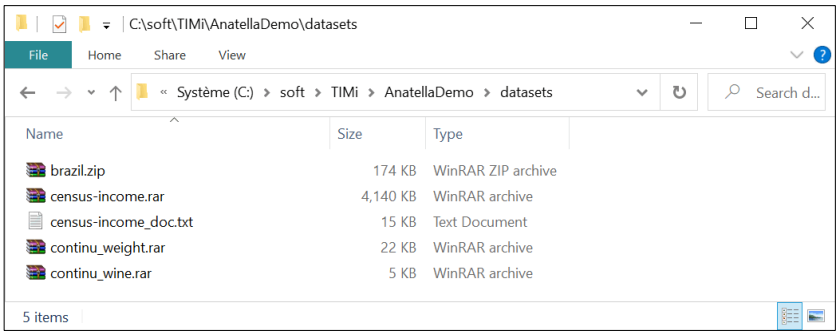


Figure 1: Central Dataset Repository Content

As you see in the sceenshot above, our "Central Dataset Repository" directory contains several ".rar" files. Each ".rar" file actually contains one compressed Text/CSV files (see appendix A about compressed CSV files). TIMi is able to read natively compressed CSV files: The compression formats that are supported are ".rar", ".zip", ".gz" (the compressed archive must always contain one unique Text file). When TIMi uses these file formats, it does NOT decompress the files on the Hard Drive: TIMi decompresses the dataset "on-the-fly" in central core memory, thus reducing:

- the load on the hard drive
- the hard-drive consumption required to do the analysis.

The "Census-Income" dataset has been prepared for you by an expert data miner so that it's directly "*ready to use*" for predictive analytics purpose.

You can use Anatella (included inside the "TIMi suite") to easily build such dataset. Technically, all we need is the creation of a target column, although many transformations are often required to generate high quality models. Please refer to the following document to have more information about the (many) data preparation steps and how to construct a good Dataset:

http://download.timi.eu/docs/DataPreparation_Churn.pdf
http://download.timi.eu/docs/DataPreparation_PropensityToBuy.pdf

4. The TIMi Modeler Modeling Process

As mentioned earlier, the "Census-Income" dataset originates from the American Census Bureau (it's an open source dataset freely available online). Each line represents a person. The ***target column*** is a binary variable, that is true (1) when the income level for a person is above $50K and false (0) otherwise. The objective of our modeling exercise is to identify which USA residents will have "Target= TRUE", using all the information available.

With TIMi modeler, the process to create a new predictive model consists of three steps:

1- Step 1: Where is my dataset?

We need to set the location of the dataset to analyze. Typically, our dataset will be inside the "Central Dataset Repository" directory. This dataset can be a text file, a .gel_anatella file, the result of a query on a SQL database, or any file format we mentioned above. The information about the location of your dataset is stored inside a ".DSourceXML" file.

At the end of this first step, TIMi Modeler attempts to guess the type of each column/variable in the dataset based on some heuristics. Based on these guesses, it will produce a ".TypeXML" file (see next step to know more about this file)

2- Step 2: What are the types of the columns inside the dataset?

We now need to set the type of the columns. There are basically five types of columns:

a. **Value type.** Examples are: Age, Size, Cost, Price,…
b. **Nominal type.** Examples are CarLabel, Region, Sex, …
c. **Binary type.** Examples are: isMale, isForeigner,…
d. **Target type.** What is the "Target Column"?
e. **Key type.** What is the "Primary Key" column?

The information about the type of the columns is stored inside a ".TypeXML" file. This file is the end-result of the previous step (step 1). You (normally) have to carefully check it do the necessary changes if some column's types were not guessed properly by Modeler during the step 1.

At the end of this step, TIMi Modeler generates several reports about the data quality of the dataset, some statistics about the content of every column, and some charts. It also generates a ".CfgXML" file (see the next step to know more about this file).

3- Step 3: Who are my targets?

We now need to set the selection of lines and of columns inside your dataset that we want to analyze. Most of the time, no sub-selection is needed as you will analyze all the lines and all the columns of your dataset, so you don't have to provide anything here (You can leave the default values "as is"). Your selection is saved inside a ".CfgXML" file.

At the end of this step, Modeler generates an "analyst" report that explains how we can identify the "Targets" (in our example: how to recognize somebody that has an income level above $50K). This "analyst" report contains information about the exact profile of a US resident with an income level above $50K.

TIMi Modeler also generates a "predictive model". This model is using all the information contained inside the columns of your dataset to guess if a person is "*inside the target*" or not.

Typically, Modeler constructs a "predictive model" using no more than 15 to 25 columns of the dataset to perform the guess. Why such a small number of Columns? Because, usually, the columns that are ignored by TIMi Modeler simply do not contain any other additional relevant information compared to the columns already used by the model.

At this stage, the model creation process is complete. However, several steps must be taken to fully deploy the models and put it into production. Such steps are usually completed using Anatella. A TIMi predictive model can be:

- …used directly inside Anatella (this is the easiest for the deployment of your models and also the recommended way to put your models in production).
- …used directly on the command-line (in a batch script)
- …exported:
  - …to "SAS base" code
  - …to simple 'SQL" code that runs in almost any database engine.
  - …to Python
  - …to VBA (to run your models inside an Excel Sheet)
  - …to HTML and Javascript (to run your models inside your Internet Browser)

5. Running TIMi Modeler.

Let us now begin our analysis of the "census-income" dataset

After the installation process of TIMi is completed, you should have on your desktop the following icon:

Double-click the above TIMi icon and the "Main Window" of TIMi Suite appears:

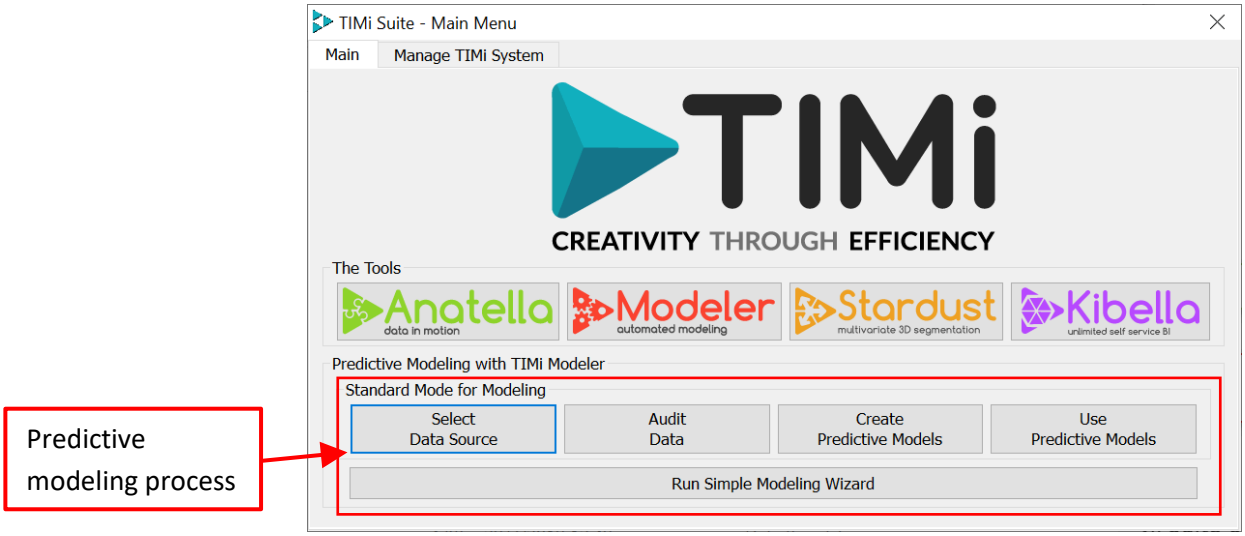


Figure 2: The TIMi Suite Main Menu

The TIMi Suite offers three different interfaces. Each interface is adapted to the degree of technicity of the user. Note that no previous knowledge of algorithm design or statistical modeling is required for either of the three levels.

We will focus on the Predictive Modeling Process area, which consists of 4 tasks:

- Data selection
- Data Audit and validation
- Model Creation
- Model Application

5.1 Three levels of user's interface

1. The « Simple WIZARD » Interface

This simple interface is designed for users with limited experience (or no experience) in datamining/Modeling.

The user is guided through a standard wizard in which all the steps of the model building process are included, and the creation of the directory structure is automated. At each screen, the user answers a simple question and then clicks « NEXT ». At the end, a predictive model is generated, with the accompanying report. These predictive models can then be used "in production" through a set of automated tools.

This interface allows you to solve 90% of the « classical » datamining problems.

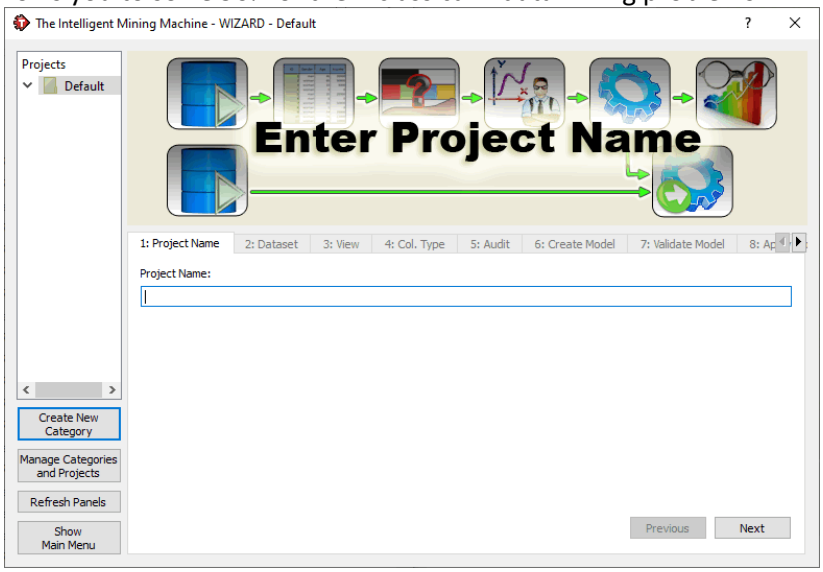


Figure 3: Simple Wizard Interface

2. The « Standard Mode » Interface.

This interface requires a bit more experience in terms of data structure, yet gives the user a bit more freedom and control when steering the wheel. You can easily create predictive models for binary targets (risks models, fraud detection, appetency, etc.), for continuous targets (profitability models, share-of-wallet, etc.) and for multi-class targets.

With TIMi Modeler, you can spend your time understanding your data instead of spending time understanding the parameters of the datamining software that analyzes your data!

All the parameters available in this mode have a "business" sense that is easy to understand (i.e. there are no "*impossible to understand*", intricate parameters linked to an incomprehensible internal algorithm that you did know it existed). Using the « Standard Mode » interface, it's almost impossible to wrongly set the parameters of TIMi Modeler: We always obtain a relevant predictive model (with the corresponding analysis report).

This current document (TIMi Modeler Quick User's Guide) gives a detailed explained of the « Standard Mode » Interface.

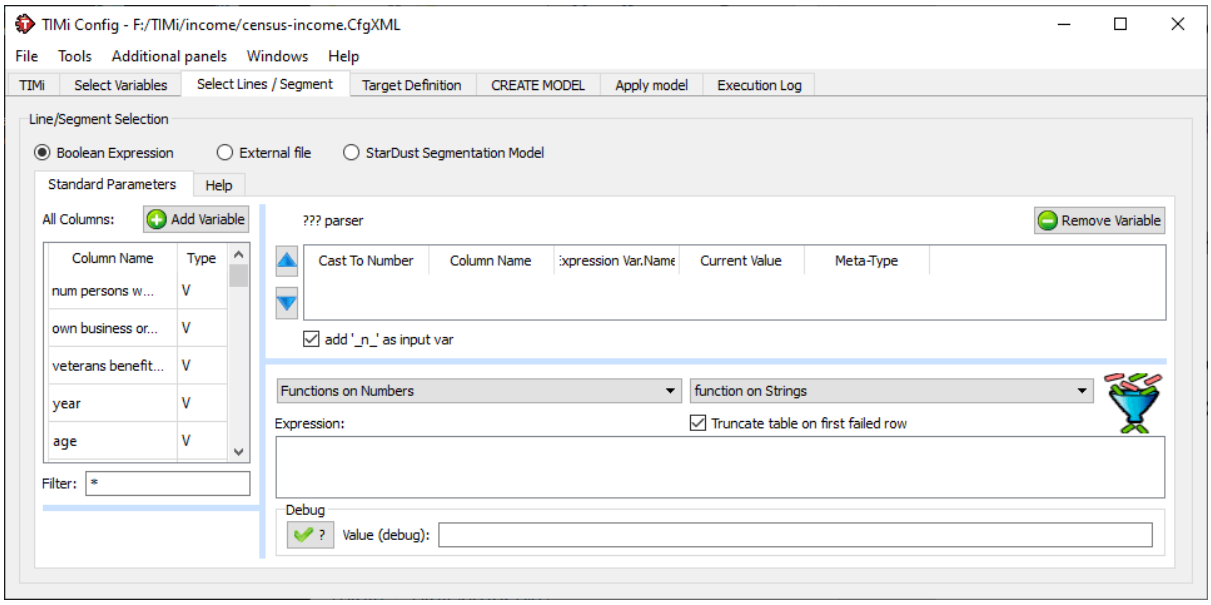


Figure 4: Standard Mode Interface

3. The « Expert Mode » Interface.

This interface allows you to modify almost all the internal parameters of the algorithms used by TIMi Modeler. A detailed knowledge of the algorithms used by TIMi Modeler is required to understand and properly adjust these parameters.

In most situations, the defaults values of the parameters already generate very efficient predictive models. If you want to "extract" a "few percent more" from your data and create slightly better predictive models, you can adjust these meta-parameters but, in general, the gain is really marginal, unless rare problems of over-fitting arise.

Please refer to the TIMi Modeler "Advanced User Guide", to have more information on the algorithms used by TIMi Modeler and to know how to parameterize them correctly.

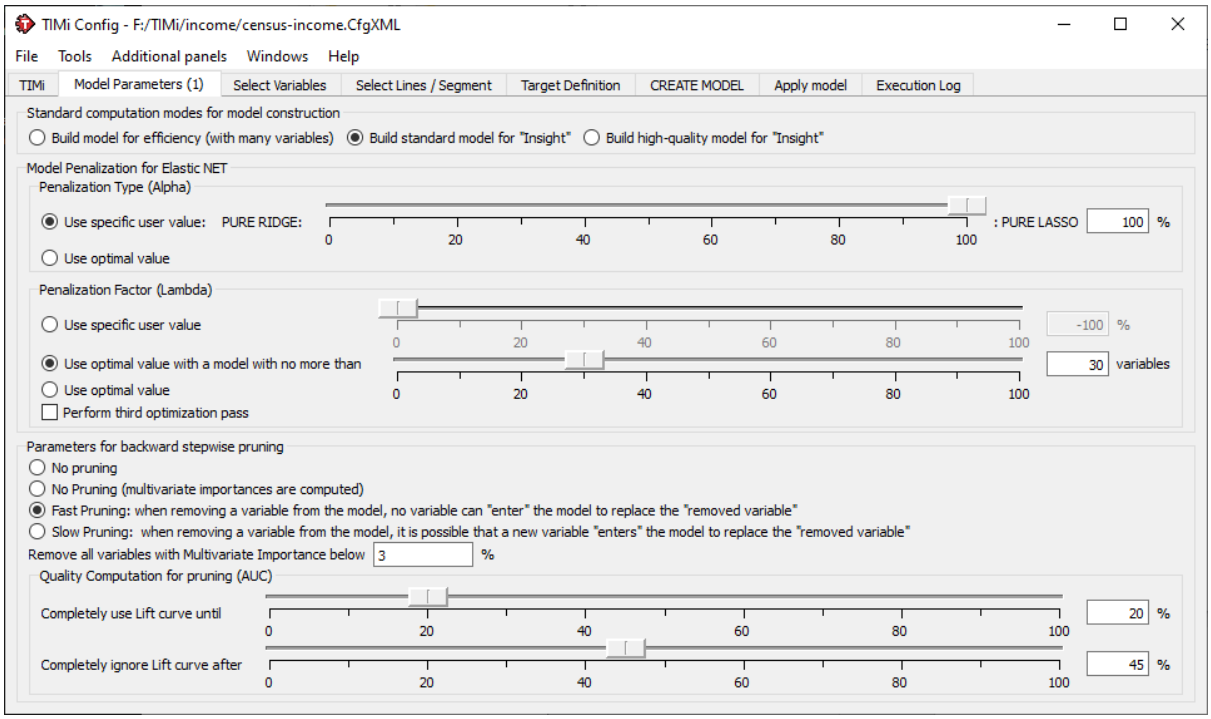


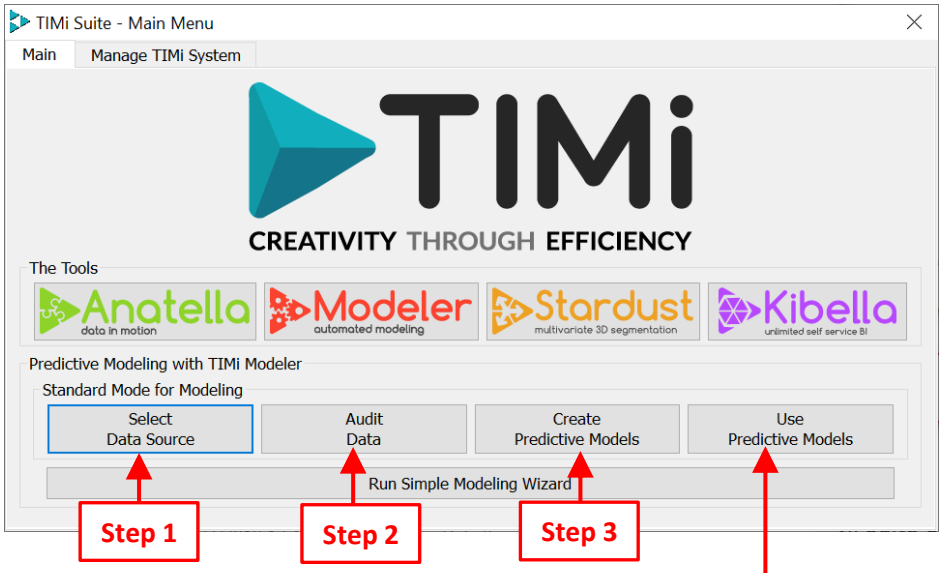
Figure 5: Advanced Mode Interface

Let us now focus in details on the « Standard Mode » Interface. There are three main components inside this interface:

- The DataSource Editor. (Step 1)
- The Type Var file Editor. (Step 2)
- The Config File Editor. (Step 3)

Those three components correspond to:

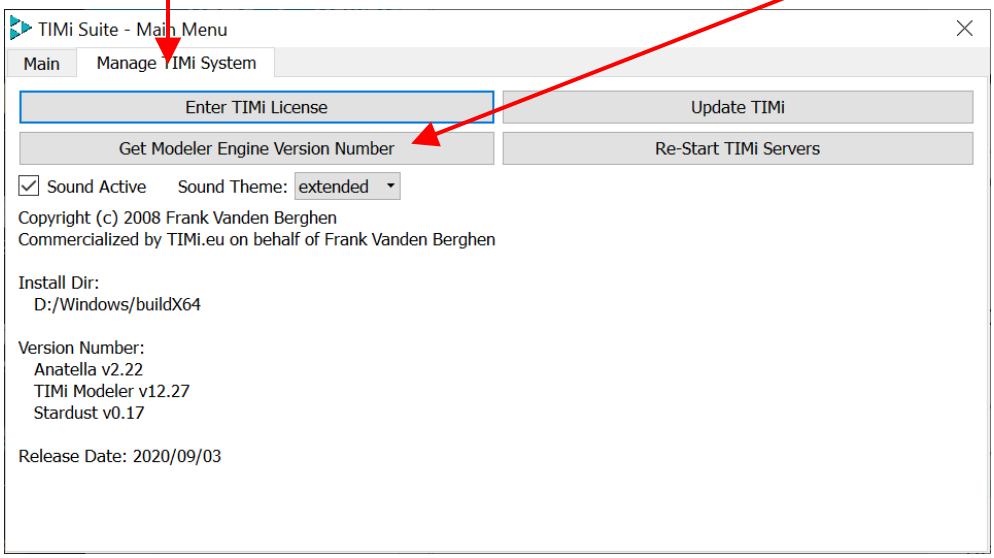
1. The three steps described in the introduction section.
2. The first three buttons on the Data Mining Process on the TIMi Main Menu:



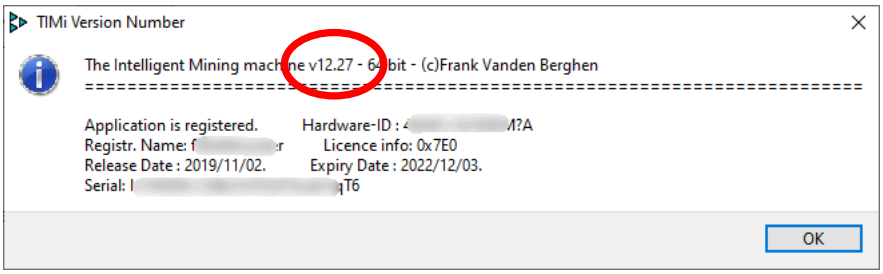
We'll also talk about the fourth, very important, step: the "Apply Model" Step:

5.2 Command-Line Help Menu

You can check your version of TIMi in the following way: inside the “main window of TIMi” click on the Manage TIMi System tab and on the "Get Modeler Engine Version Number" button:

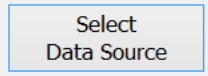


The version number is inside the new window:



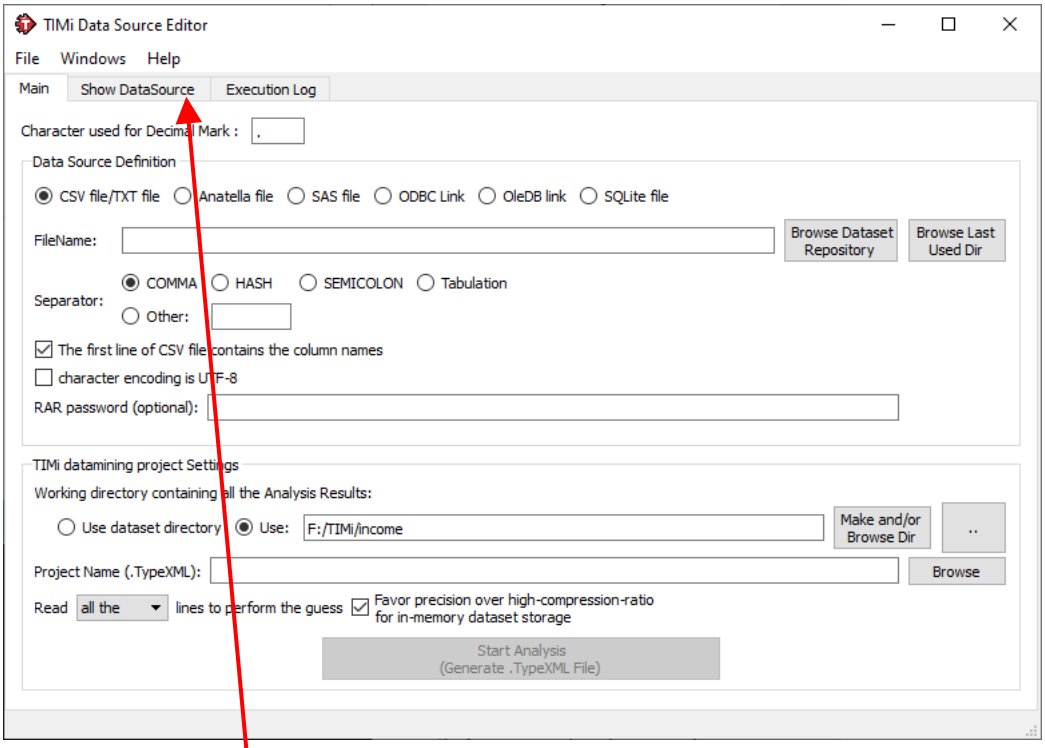
6. Predictive Modeling with TIMi Modeler

6.1 Data Selection: The DataSource Editor (Step 1)

Click on the 

button of the TIMi main window.

The following window appears:



The first step of the analysis is obviously to open the dataset file on which we will work on. Using the DataSource editor, you can specify the location of the dataset that you want to analyze. The default storage is CSV/Text file (CSV="Comma Separated Value" text file).

It is important to properly set the separator character when working with text file. It can be a comma ",", a Hash "#", a Semicolon ";"a TAB character, or any other which you may specify. When in doubt, click on "Show Datasource" and TIMi Modeler will display the first 100 lines.

Please also note the presence of the parameter named "*character used for decimal mark*". The default setting is ".". It means that a column containing "3,14" won't be recognized as a "value" column but rather as a "nominal" column because of the comma. If you want that TIMi Modeler sees "3,14" as a number you must:

1. Change the parameter named "*character used for decimal mark*" to the value ",".
2. In particular: This will **NOT** work:

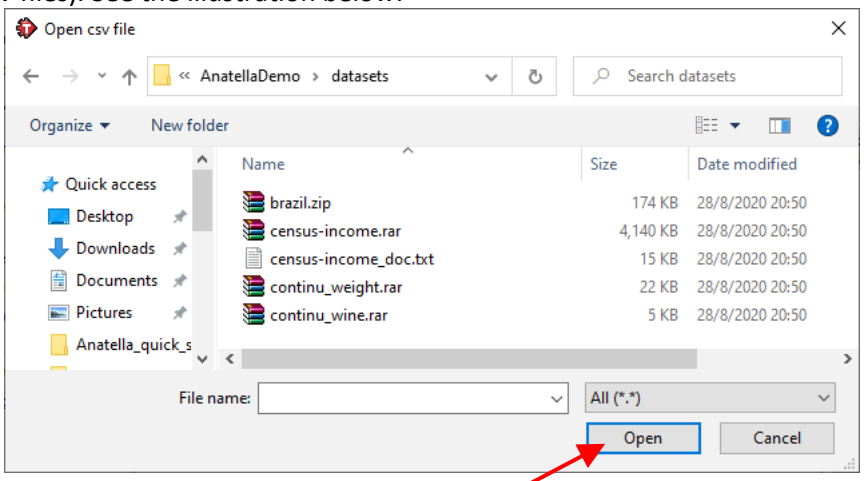
   Manually changing the type of the variable from "nominal" to "value" inside the "Type Var Editor" (see next section) will **not** work.

Alternatively, a better solution might be to use Anatella to replace the "," comma character with a "." dot character (using the ReplaceStrings action).

Note about UNICODE:
Ideally, the character encoding of your text file is Unicode UTF-8. Unicode allows you to manipulate datasets containing any kind of characters: Cyrillic, Chinese, Greek, etc. If unicode is used, TIMi Modeler automatically produces unicode version of the reports in MSWord and MSExcel.

If the first 3 bytes of your text dataset are `0xEF,0xBB,0xBF` (This is the Byte-Order-Mark for utf-8 encoded text file) then TIMi Modeler automatically switches to the Unicode mode. By default, Anatella always produces utf-8 unicode text files with a correct utf-8 BOM (Byte-Order-Mark) header (so that everything is automated and transparent for the end-user when using Anatella).

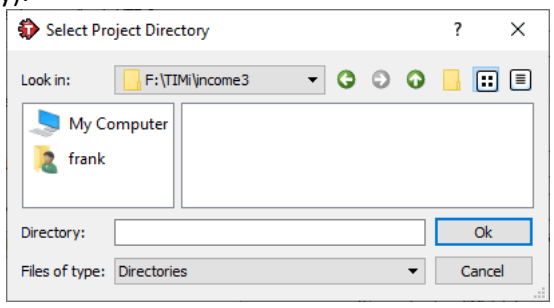
Let us go back to the "census-income" example: Click on the Browse Dataset Repository button, go to the "Central Dataset Repository" directory, and select the "census-income.rar" file (see appendix A about compressed CSV files). See the illustration below:



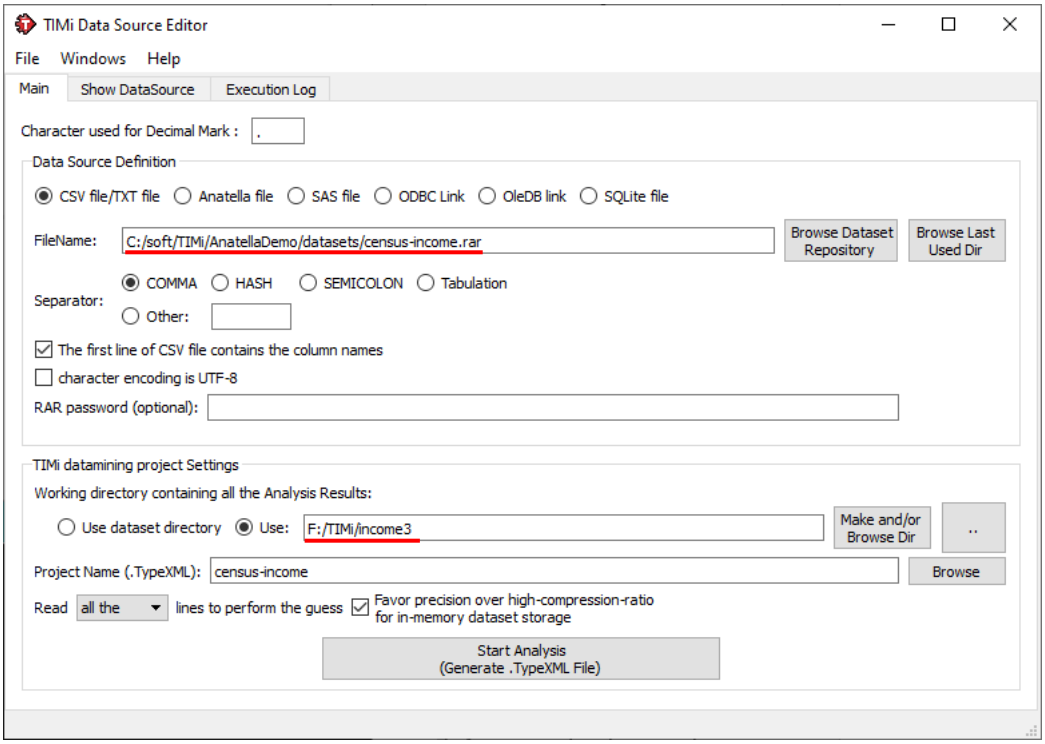
Once you click "Open" the connection is established.

Note that the dataset can also be stored inside an Anatella .gel file, a .sqlite database file or inside any database accessible through ODBC or OLEDB technique.

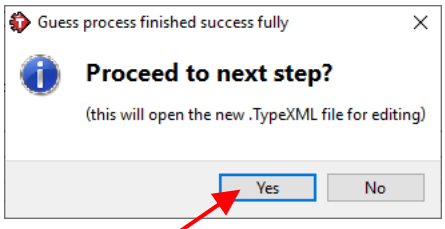
The next step is to set the working directory that contains the analysis results. You can either choose to use the current directory (where the source file is located) or any directory of your choosing. Click on the Make and/or Browse Dir button and create/select a working directory (here below, we selected the "f:\TIMi\income3" directory):



You can also change the name of the project to something more meaningful, like "demoIncome" (the same "working directory" can contain several projects, if they have different names). You should now have something like this:

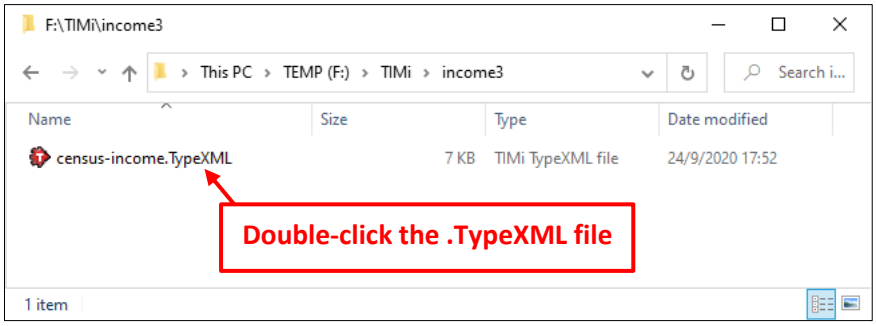


Click the Start Analysis (Generate .TypeXML File) button and wait for a few seconds (it can be a few minutes on very large dataset files). You should pretty soon see the following message:



The first step of the analysis is now completed. TIMi Modeler performed a first analysis of your dataset in an attempt to guess what's the type of each column of your dataset. Based on this guess, it produced a ".TypeXML" file that appeared inside the "working directory".

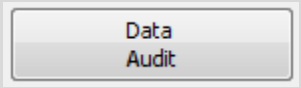
Click the yes button: The TIMi Modeler ".TypeXML file Editor" appears. Alternatively, you can also double-click the "demoIncome.TypeXML" file inside your working directory:



6.2 Review and Audit Data: The Type Var file Editor (Step 2)

The goal of this step is to validate the type of the variables inside our dataset. These variables are used for the whole remainder of the analysis, so we'd better check carefully their type!

There are three ways to start the .TypeXML file Editor:

1. Create a new .TypeXML file using TIMi Modeler and open it directly after creation (this is normally what you just did in the previous section)
2. Click the 

button inside the main menu of TIMi.
3. Double-click on a "*.TypeXML" file inside a "MSWindow File Explorer" window.

6.2.1 Variable Types: Review and Implications

Using the "Type Var Editor", you can specify the type of each column of the dataset.
There are basically five types of columns:

a. **Value type.** Examples are: Age, Size, Cost, Price, …
This type of column can only contain numbers and exhibits an ordering property. For example, a six-year old boy is younger than a twelve-year old boy and a twelve-year old boy is younger than an eighteen-year old boy. There is an order. So, a column containing the age of a person should be of type "value". On the contrary, a column containing the zip code of the house of a person should be of type "nominal" because the zip code 1050 is NOT smaller or bigger than the zip code 1210. There is no order inside the zip code.

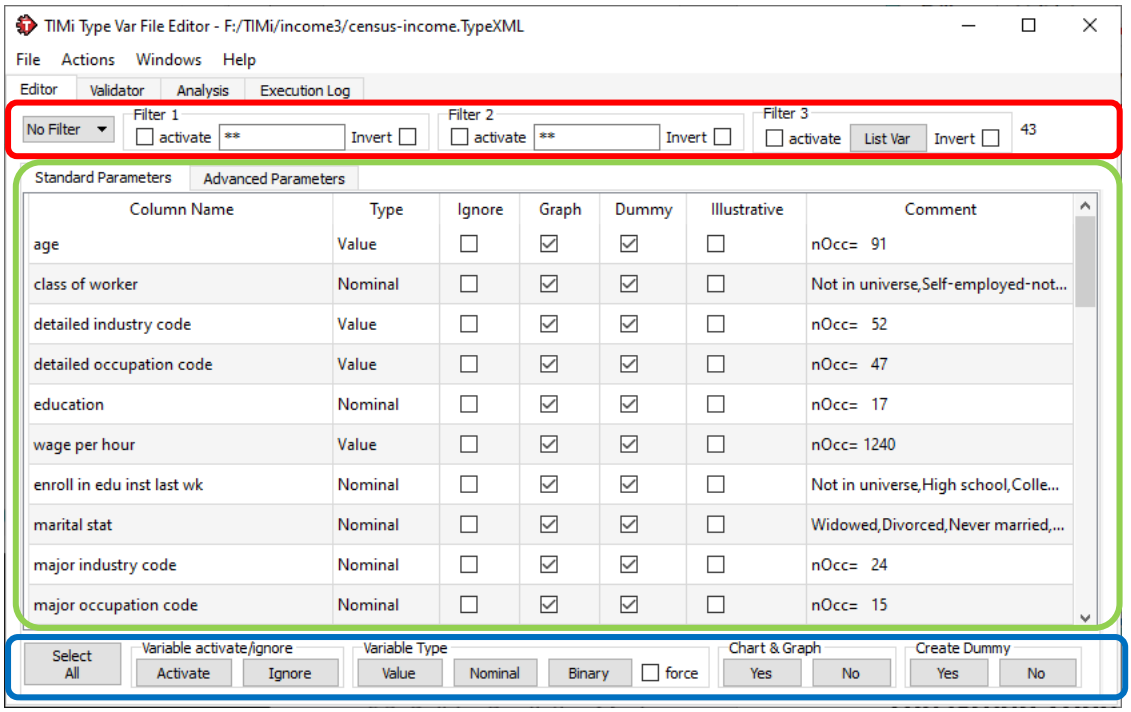
b. **Binary type.** Examples are: isMale, isForeigner, …
This type of column can only contain a "true/false", "yes/no" semantic. These columns contain only two modalities (T/F) or three modalities (T/F/missing).

c. **Nominal type.** Examples are CarLabel, Region, …
This type of column contains anything that is not "value type" or "binary type".

d. **Target type.** What is the "target column"?

e. **Key type.** What is the "primary key column"?

The Type Var Editor window is divided in three **zones**: See the illustration below:



The Green Zone contains information about all the columns of your dataset.

Inside this zone, you can see the column "**Type**" that contains all the column's type inside your dataset. The dataset column's type is based on a guess done by TIMi Modeler, and (as it is the case with everything that's coming out of a heuristic) guesses can sometime go wrong. For example: a postal code will be detected as "value type", but it is in reality a "Nominal type". You should check the column's type to see if the guess was OK.

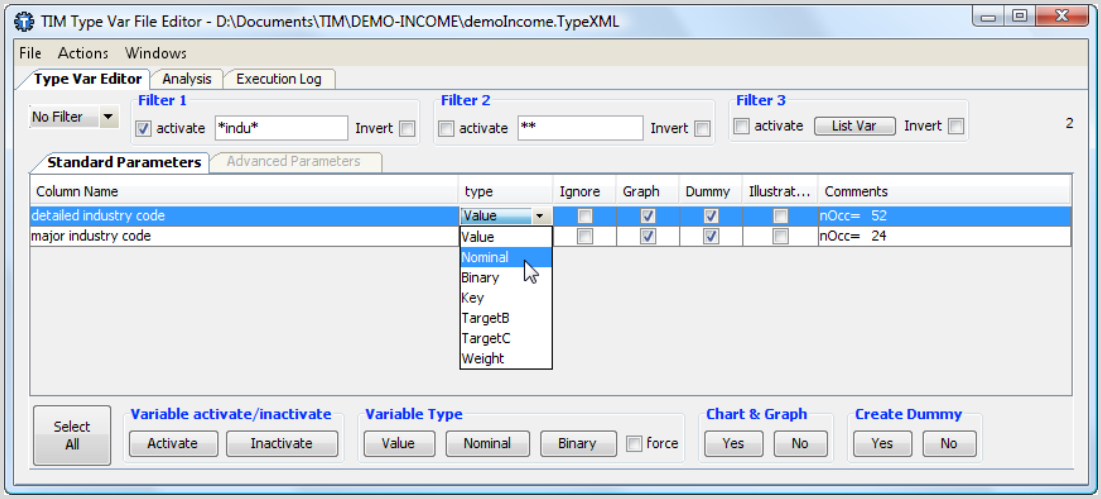
If a column containing the number "3,14" is incorrectly detected as the "nominal", you must go back to the DataSource Editor and change the "*character used for decimal mark*" parameter to ",". Alternatively, a better solution might be to use Anatella to replace the "," comma character with a "." dot character (using the ReplaceStrings action).

Note that TIMi Modeler will work fine even if some types are wrongly defined. However, if there are too many errors in the column's type then the quality of the predictive model (the quality of the lift) might be a little bit lower.

Usually, TIMi Modeler's guesses are fine (except for zip and other numeric "code" type of variable). So, don't worry if you do not want to look at the type of ten thousand variables. You can proceed directly to the next step to construct a "predictive model".

Remark: Once I have a predictive model, I check carefully the type of the few columns (usually no more than 25) that are actually really used to perform the prediction. As a general rule of thumb, it's never a good idea to spend a lot of time cleaning and "tuning" all the dataset's columns. You should focus only on the columns that are used by the predictive model. TIMi Modeler is pretty robust to un-cleaned data, so don't worry too much about having a "perfectly" correct .TypeXML file.

You can edit directly the table to change the column's type. For example, the column "detailed industry code" is a code and thus should be a "nominal value". Change column's type as illustrated below:



Please note that the manipulation described in this note is only an example. This manipulation has NOT been performed in the rest of this document.

The red zone: This zone contains some filter that allows you to search easily inside the central table in the green zone. Some examples:

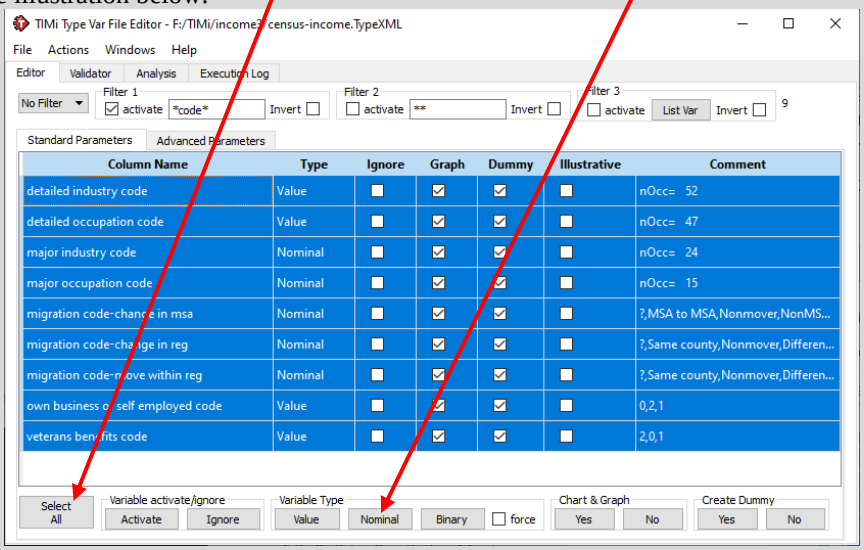
- If you enter "*code" inside the Filter 1, you will get all the columns with a name that ends with "code".
- If you enter "code*" inside the Filter 1, you will get all the columns with a name that starts with "code".
- If you enter "*code*" inside the Filter 1, you will get all the columns with a name that contains "code".

The blue zone: This zone is used to apply changes on the selected lines of the central table in the green zone.

An example of usage of the red and blue zone: We want to set to the "nominal" type all the columns that contain "code":

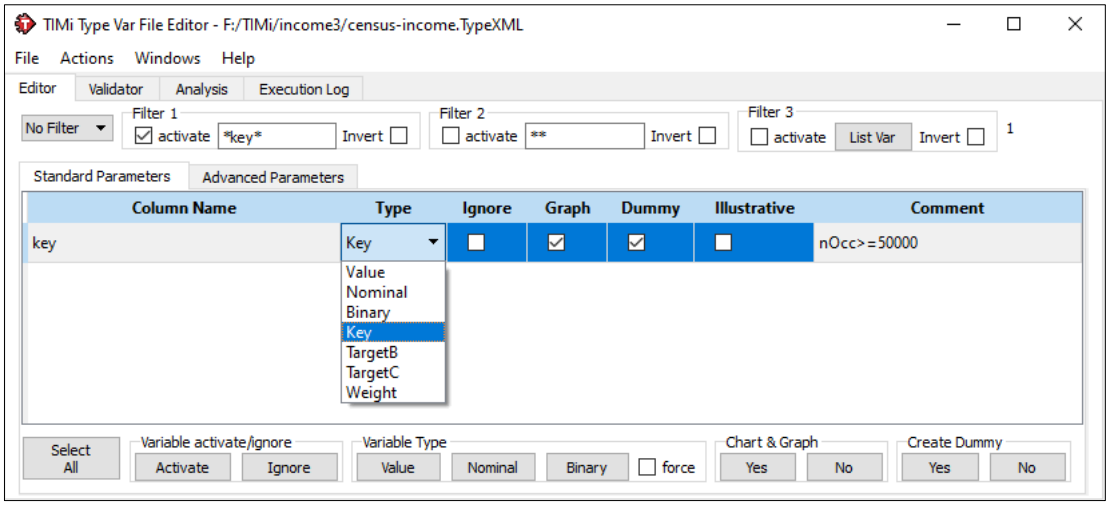
1. Enter "*code*" inside the Filter 1.
2. Click the Select All button and then the Nominal button.

See the illustration below:

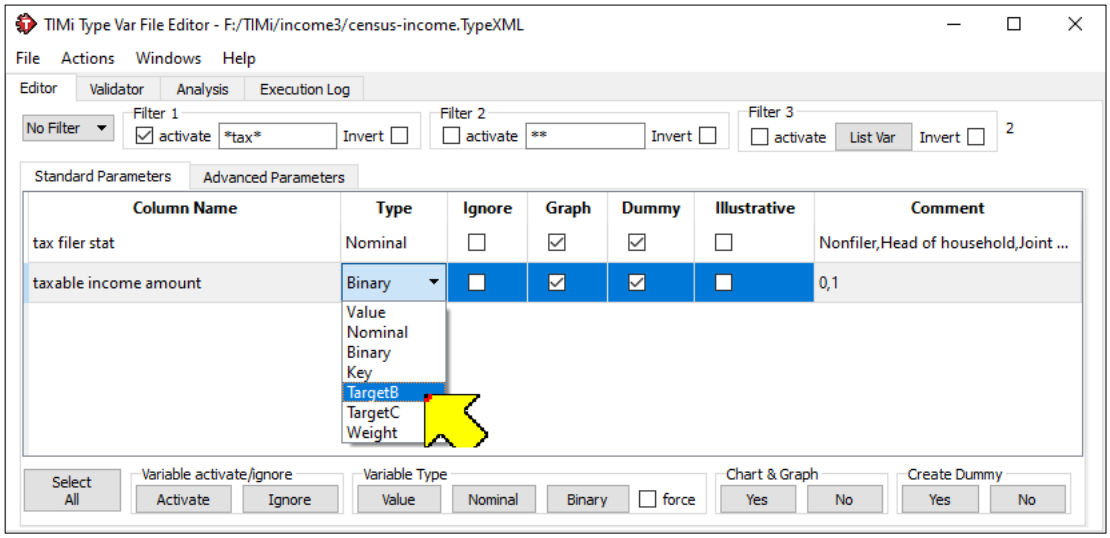


Please note that the manipulation described in this note is only an example. This manipulation has NOT been performed in the rest of this document.

You must define the column that contains the primary key. In our example the primary key is the column “Key”. See illustration:



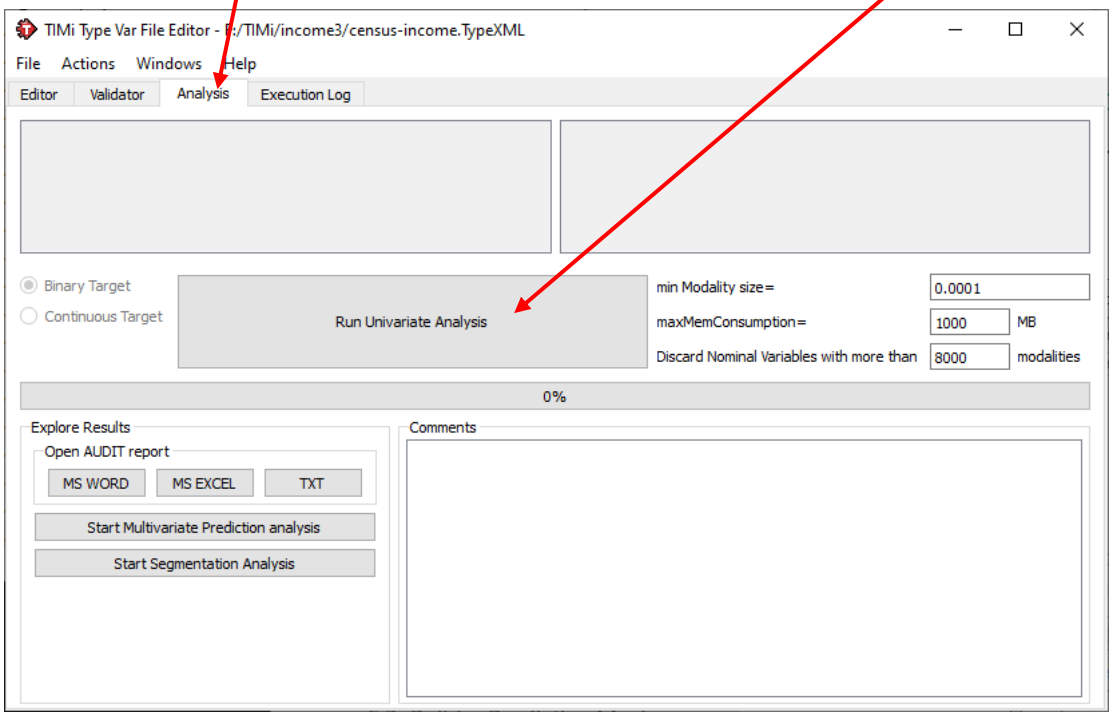
We also have to define the column containing the target. In this example the target is of type Binary True/False: We select inside the .TypeXML the column “taxable income amount” as TargetB (“B” stands for “Binary” target). See illustration:



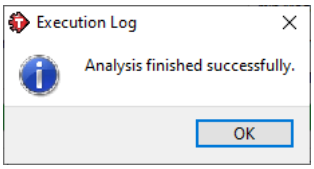
The other parameters available inside the .TypeXML File editor are not important at this time. You can leave them at their default value.

6.2.2. Generate the Data Audit

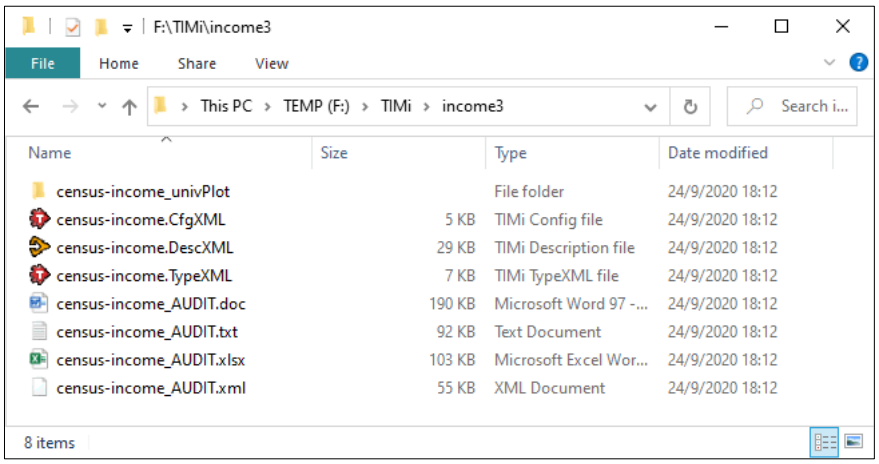
Click on the “Analysis” tab: …and then on the “Run Univariate Analysis” button:



After a few seconds you obtain:



Your working directory now contains:



The “census-income.CfgXML” file will be described in the next section.

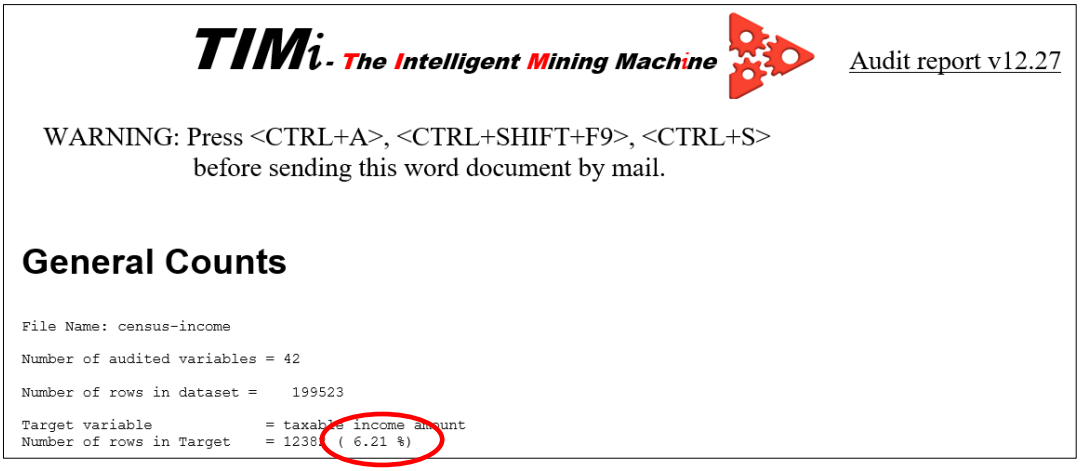
The “census-income.DescXML” file is required to do a segmentation and a prediction analysis. The “.DescXML” files are not editable and contain only information used internally by TIMi Modeler and

StarDust. The ".DescXML" files contain information about the distribution and the recoding of the different variables that is required by TIMi Modeler and Stardust to correctly process the data.

The "census-income_AUDIT.doc", "census-income_AUDIT.txt" and "census-income_AUDIT.xml" files have all the same content. The only difference between these files is: « *Most of the time, the .doc file is the one you are interested in. The .txt file is faster to open but it does not contain any images. The ".xml" is only useful for the automatic generation of commercial reports.* ». Let us have a first look at our dataset! Click on the "MS-Word" button here:

This button is a simple shortcut to open the .docx file. Alternatively, you can double-click on the "census-income_AUDIT.doc" file!

You should see something like this inside Microsoft Word:


TIMi - The Intelligent Mining Machine

Audit report v12.27

WARNING: Press <CTRL+A>, <CTRL+SHIFT+F9>, <CTRL+S>
before sending this word document by mail.

General Counts

File Name: census-income

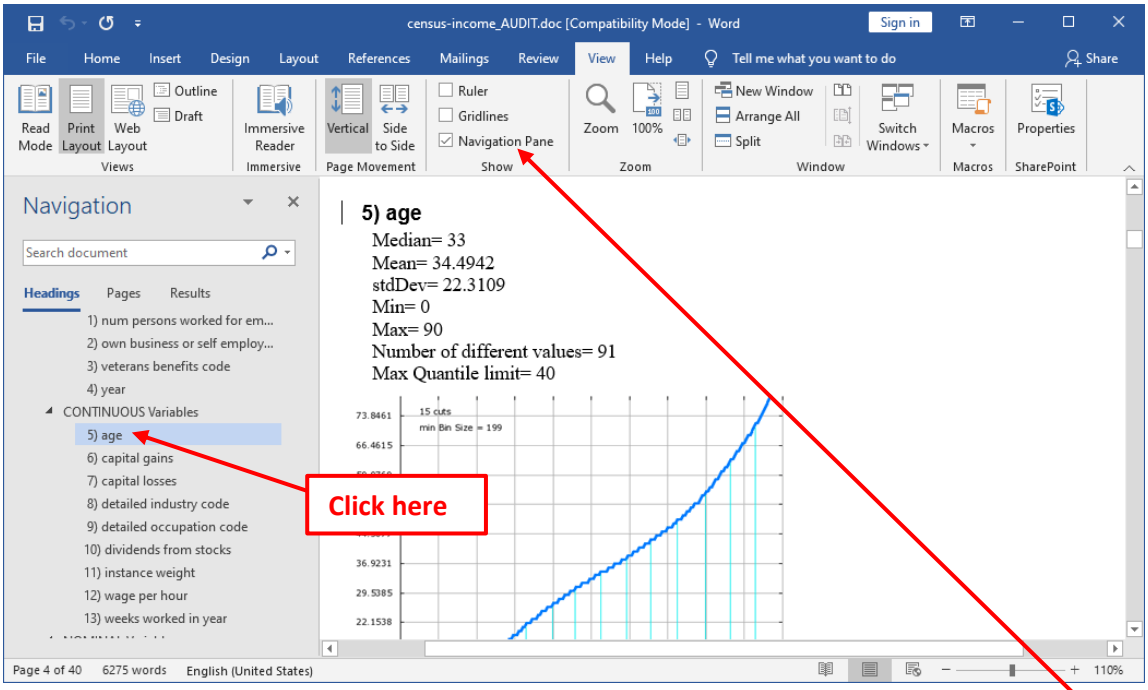
Number of audited variables = 42

Number of rows in dataset = 199523

Target variable = taxable income amount
Number of rows in Target = 1238[illegible] (6.21 %)

The important information here is the percentage of target inside the dataset: 6.21%. It means that if you pick at random a row (a person) inside your dataset, you have 6.21% of chance of to find somebody that is a target (i.e. that has an income level above $50K). "6.21%" is named the "***a priori probability***" (to be a Target) or "***natural density***" of targets.

Let us scroll down a little inside the AUDIT document. Let us look at the information extracted from the column "age". You can use the "Navigation Panel" to directly access the variable "age". You should see the following:

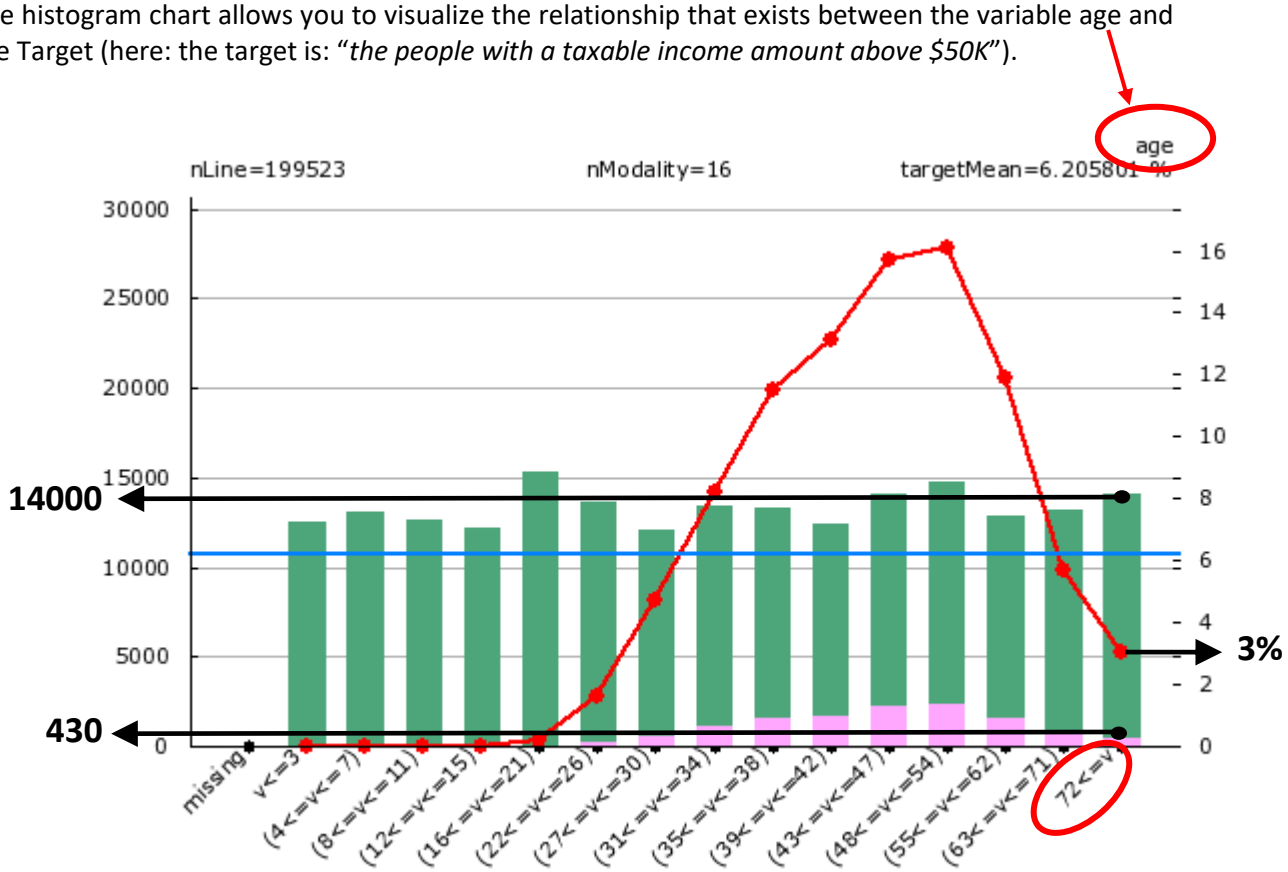


If the "Navigation Panel" is not visible, you can activate it in the "View" menu by clicking here:
When you scroll down a little further, you see a red/green histogram chart.

6.2.3. Interpretation of the red/green histogram charts.

Let us go back to the variable "age".

The histogram chart allows you to visualize the relationship that exists between the variable age and the Target (here: the target is: "*the people with a taxable income amount above $50K*").



What do we see on the above chart?
TIMi Modeler automatically discretized the variable "age": it regrouped together the values inside the column 'age' into small "bins" or "modalities". In the chart above, there are 15 modalities. Let's focus our attention on the modality "72<=v" (that includes all the individuals with an age greater or equal than 72).

You can see on the chart (looking at the **green** bar), that there are approximately 14 000 records matching this condition. Amongst those 14 000 people, there are 430 targets (look at the very small **pink** bar). It means that if you pick randomly somebody that is older than 72, there is 430/14 000=.03= 3% chance that this person is a target. The **red** line on the graph represents the probability that a person is a target in function of its age (for the statisticians: it's the conditional probability P(target=1| a<=age<=b)). For example, looking at the red line above, we can see that, for the modality "72<=v", the probability to be a target is 3% (the **red** and **blue** lines refer to the vertical axis on the right and the **green** and **pink** bars refer to the vertical axis on the left).

The **red** line is the most important part of this chart. It allows you to see "trends" and "pattern" linking some specific variables to the target.

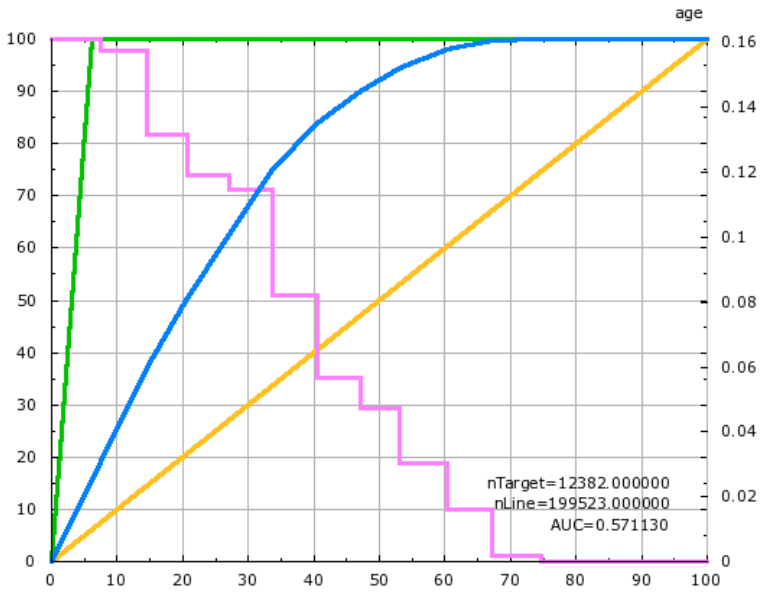
The **blue** line represents:

- The "***a priori probability***" to be a target or the "***natural density"*** of the target.
- The probability to be a target if you pick somebody at random.

What can we conclude?
In USA, the wealthiest people are between the ages of 47 and 53 (inclusive) (because the red line is the highest in this range). Between the ages of 29 to 61, the red line is above the blue line. It means that, between the ages of 29 to 61, you are wealthier than the *common, average guy*.

6.2.2. Introduction to the lift curves.

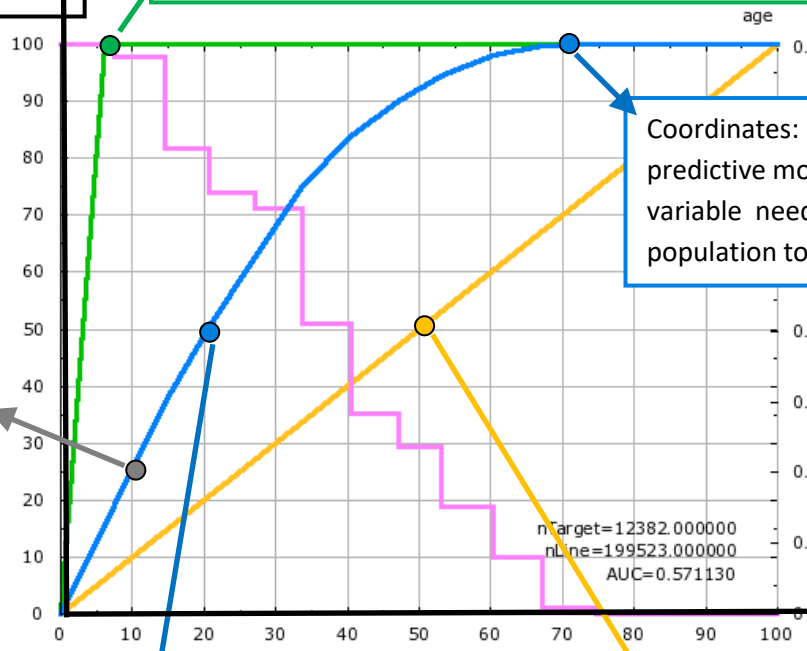
Inside the AUDIT report, TIMi Modeler also reports many LIFT charts: i.e. One lift chart for each different variable in the dataset. These lift charts represent graphically the accuracy of a "univariate" predictive model that is built using the "current" variable. A "univariate" predictive model is a predictive model that uses only one variable to compute its predictions. For example, for the column "age", the LIFT chart is:



To explain the lift chart, let us assume that we have a "perfect" predictive model that makes no mistakes. This "perfect" predictive model detects without doing any error, all the targets inside your population. In our example, here, the target size is 6% of the population (i.e. there is 6% of the population that earns more than $50K per year). This "perfect" predictive detects 100% of our targets selecting only 6% of the population. The quantity of "detected targets" is displayed on the Y axis while the size of the selection is displayed on the X axis. This "perfect" predictive model is illustrated with the **green** curve. This **green** curve goes through the point of coordinates (6% ; 100%) meaning that it's able to predictive detects 100% of our targets selecting only 6% of the population:

Coordinates: (6% ; 100%): The perfect predictive model needs to select only 6% of the population to find 100% of the targets.

Vertical Axis= Percentage of targets found



Coordinates: (70% ; 100%): The univariate predictive model built using the "age" variable needs to select 70% of the population to find 100% of the targets.

Lift Index is 2.4

Horizontal Axis= size of selection

Coordinates: (20% ; 50%): The univariate predictive model built using the "age" variable needs to select 20% of the whole population to find 50% of the targets.

Coordinates: (50% ; 50%): The random predictive model needs to select 50% of the population to find 50% of the targets.

A "normal" predictive model is forced to "recruit" a lot more than the minimum 6% of the population to find 100% of the target. For example, the univariate predictive model built using the "age" variable must select 70% of the population in order to find all the targets:

If we only want to find "half the target" (i.e. 3% of the population that are inside the target) using the above univariate predictive model, we will need to select 20% of the population.

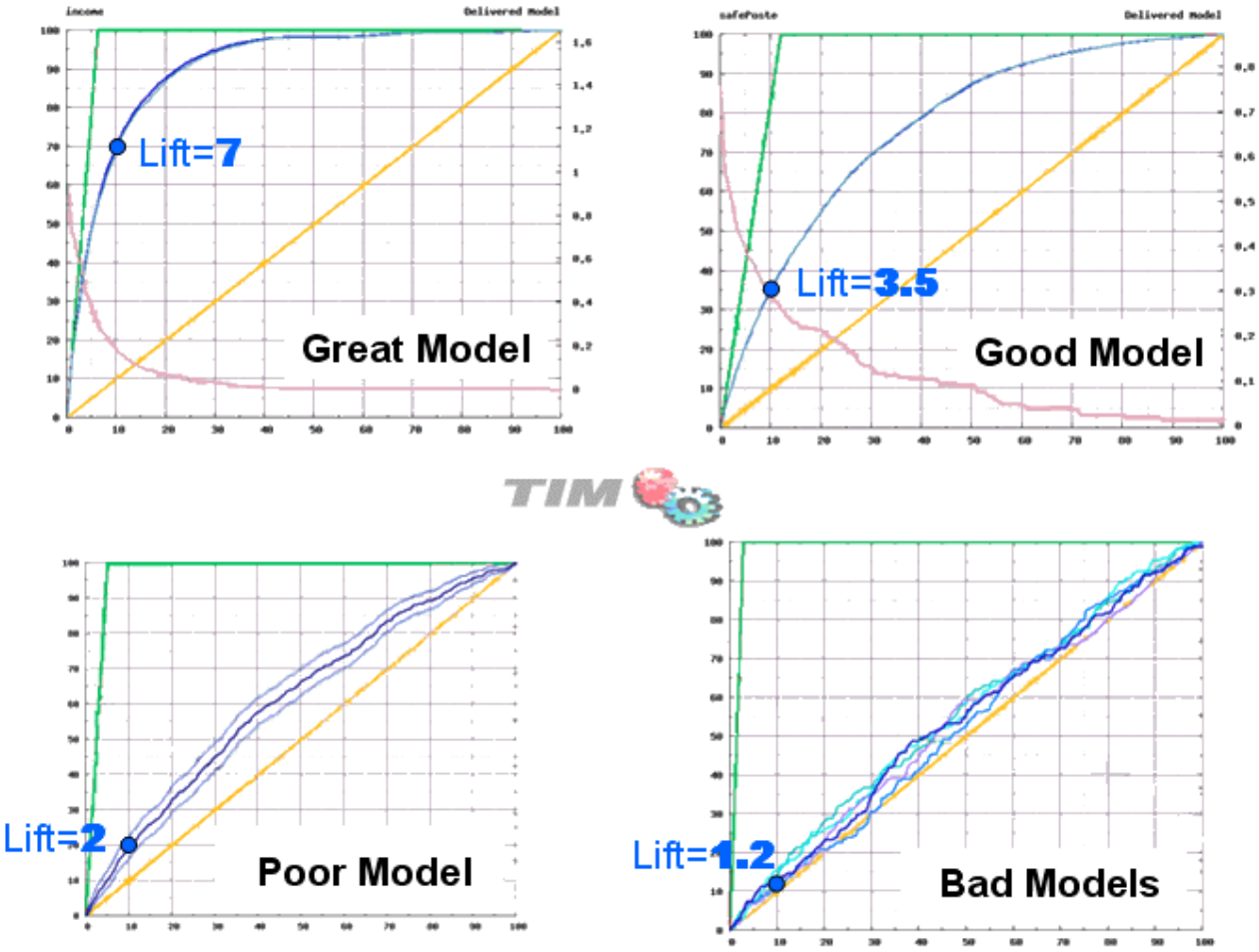
The worse predictive "model" that we can create would "select" people at random. It is illustrated with the yellow line in the lift chart above. This "random" model needs to select 50% of the population to "find" 50% of the targets (it finds them by pure luck).

Inside TIMi Modeler, the accuracy of the (binary) predictive models is always illustrated using a lift curve (in the chart above: look at the **blue** curve). The (blue) lift curve of the predictive model is always "in-between" the "perfect model" curve (in **green**) and the "random model" curve (in **yellow**). The **green** curve and the **yellow** curve thus represent the upper and lower bounds of the accuracy reachable by TIMi Modeler. ***The higher the (blue) lift curve, the higher the accuracy of the predictive model*** (but you can never go "above" the **green** curve).

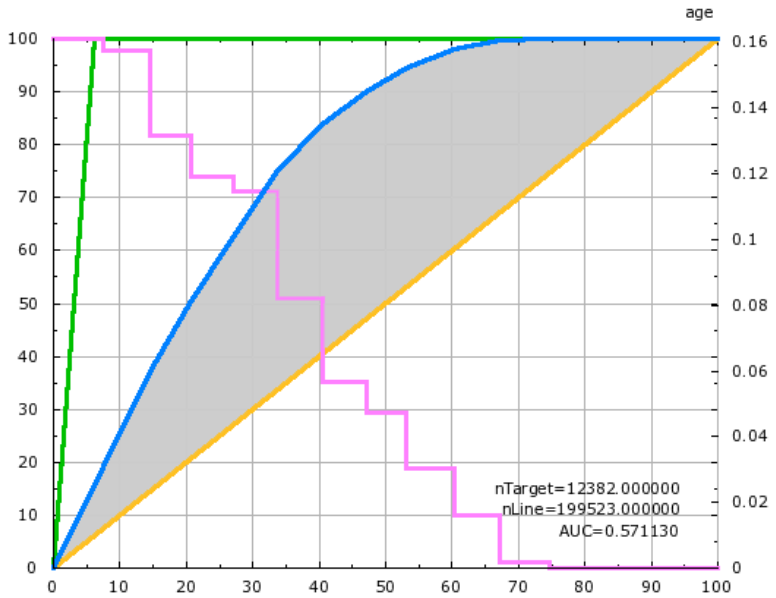
6.2.3. How to Measure the Accuracy of a predictive model

Out of the lift chart, we can extract several indexes that characterize the accuracy of the predictive model.

The most common index is the "lift index" or, in short, the "lift". In the chart above, the "lift index" is 2.4 (=24/10): If we use our predictive model to select 10% of the population, our selection is 2.4 times better than the "random model" selection because we will "find" 24% of the targets (instead of only 10% of the targets for the "random model"). The "lift index" represents "*how much better*" you are doing compared to the "random model" selection. By convention, it is always expressed for a selection of 10% of the dataset. Here are some examples:

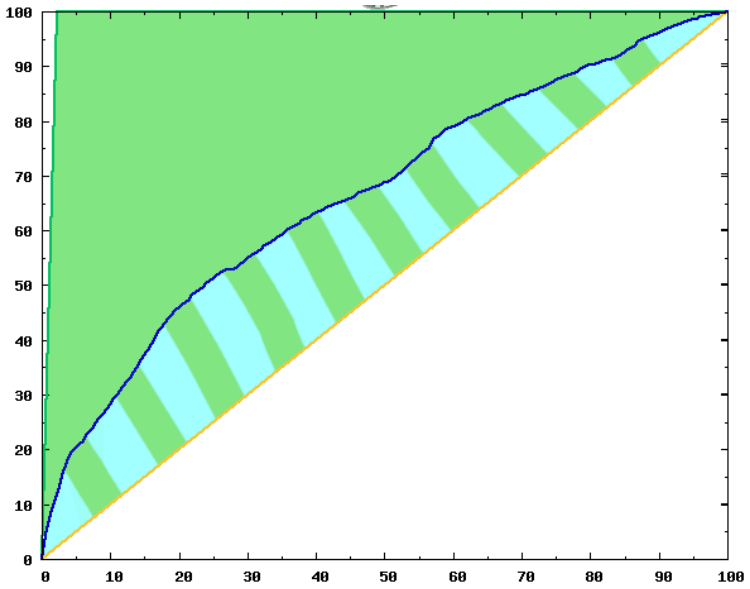


The “lift index” is very often used but it’s not very representative of the accuracy of the predictive model because it only represents one point of the lift curve (the point for the “10% selection”). There exists a better index named AUC: “Area Under Curve”. The AUC is the measure of the area “in between” the **blue** lift curve and the **yellow** line. This area is illustrated in grey in the following chart:

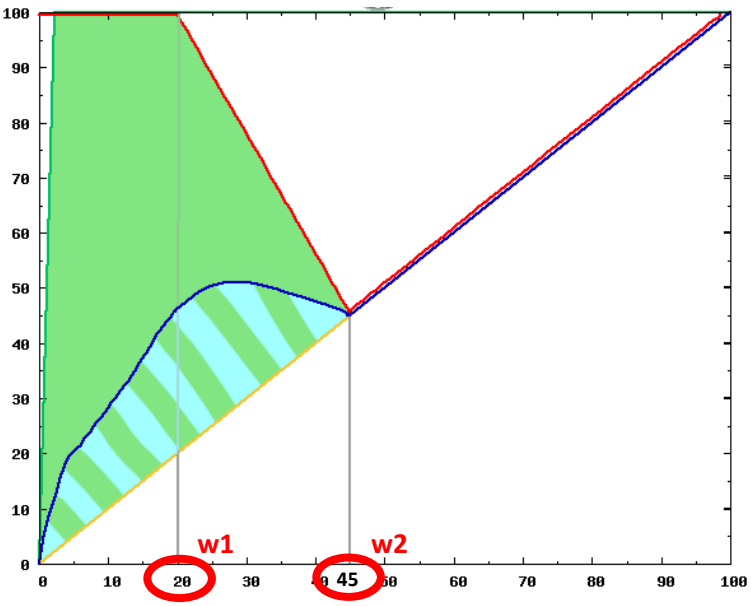


By definition, the AUC of the “perfect” predictive model is AUC=100%.
By definition, the AUC of the “random” predictive model is AUC=0%.
In the chart above, the AUC of the univariate predictive model built using the “age” variable is 57.1% (the area below the **blue** curve is 57.1% of the area between the **green** curve and the **yellow** line).

For many applications (e.g. cross-sell models, up-sell models, churn models), there is only one part of the lift curve that is of any interest: the very beginning of the lift curve, on the left. This is why, inside TIMi, we introduced a new notion named the “AUCtop”. Here is the definition of the “AUCtop”:



$$AUC = \frac{\text{blue area}}{\text{green area}}$$ **(Classical definition of the AUC)**



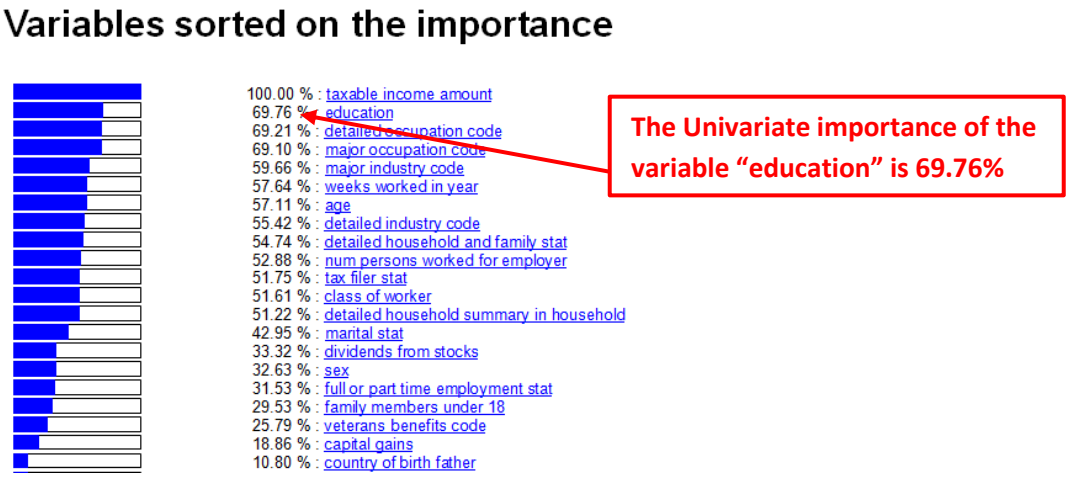
$$AUCtop = \frac{\text{blue area}}{\text{green area}}$$ **(improved definition)**

The “AUCtop” only appears inside the TIMi Modeler Analyst Reports: you won’t see it inside the Audit reports. The AUCtop of a model is defined using two parameters that are named “w1” and “w2”. These two parameters are defined inside the panel “Pruning” inside the “.CfgXML” editor (by default, this panel is hidden). To know more about the “AUCtop” of a predictive model, the parameters “w1” and “w2”, please refer to the TIMiModelerAdvancedGuide.pdf.

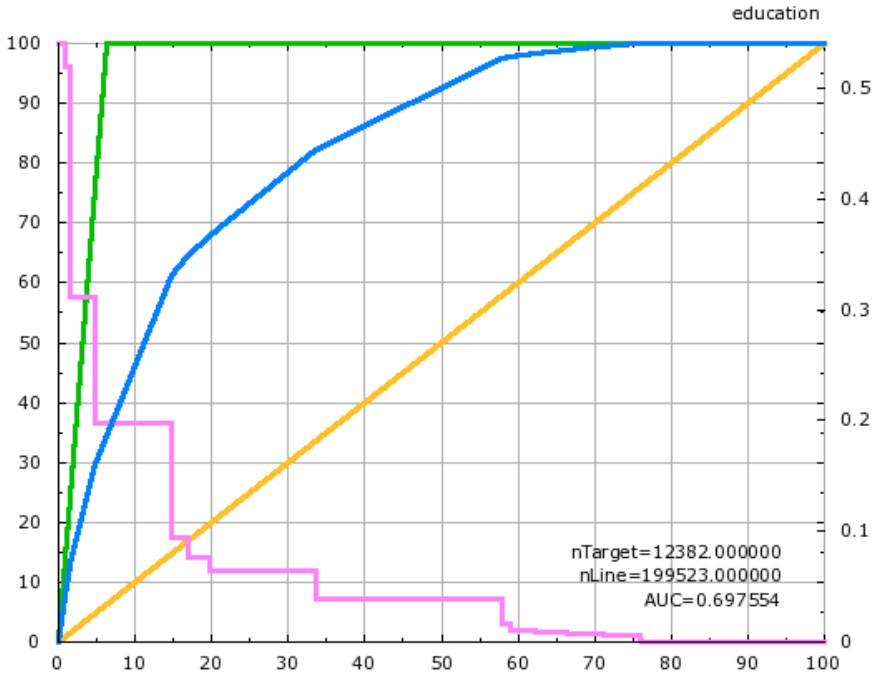
6.2.4. Univariate Importance of a Variable

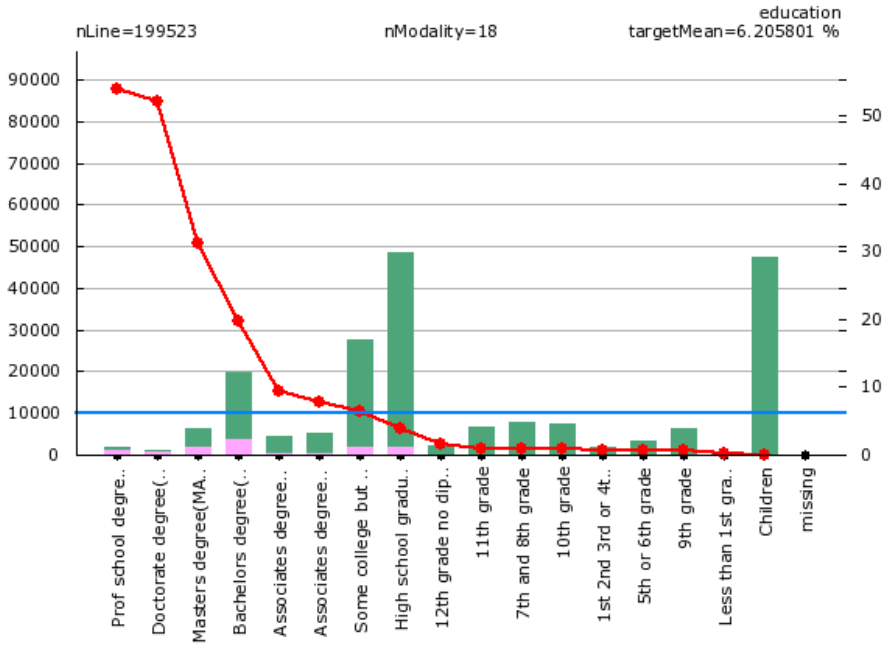
We will define the “Univariate Importance” of a variable X as the AUC of the univariate predictive model built using only the variable X. The “Univariate Importance” of the “age” variable is thus 57.1%.

The last section of the audit report generated with TIMi Modeler contains a table with all the “Univariate Importance” of all the variables in the dataset:



It means that, if we use the information contained in the column “education” alone to predict the target, we will get a lift with a AUC=69.75%. Let us have a look at the lift chart of the (univariate) model built using only the column “education”:



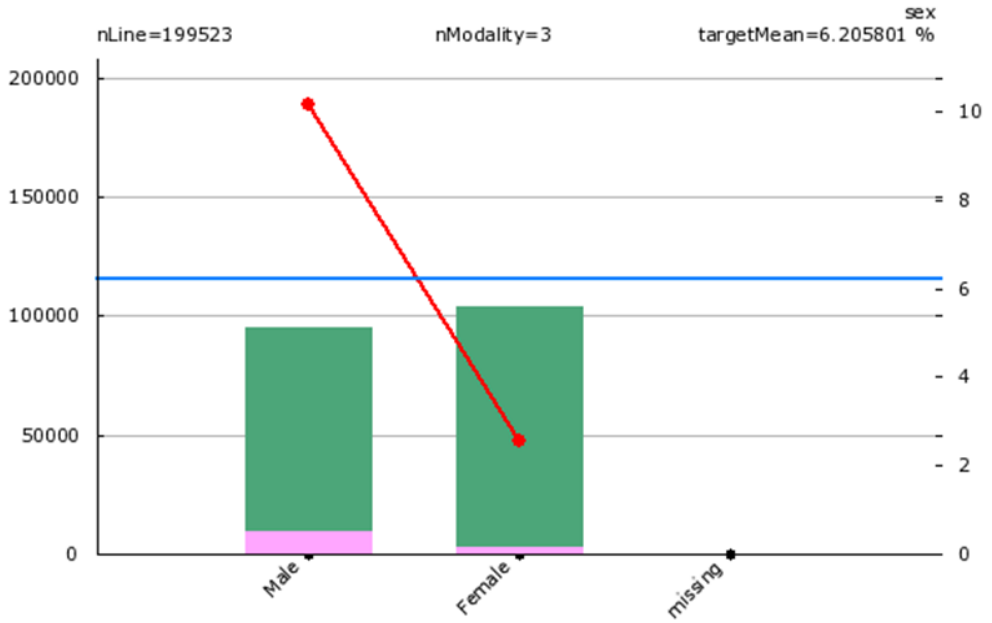


What can we conclude about the education level of a person?

A higher education means a higher chance to be wealthy on adult age. Interestingly, the “Doctorate degree” (the second from the left) study does not seem to pay off at the end. So, if you want to be wealthy, you should study for a long time and stop before the “Doctorate degree”. Apparently, too much studies fries your brain! 😉

Is the gender of the person related to its income?

Let us have a look at the column “sex”. We observe:



Univariate AUC= 32.63 %	Count Dataset	% DataSet	Count Target	Proba Target	% Index
Male	95539	47.9	9719	10.2	163.9
Female	103984	52.1	2663	2.6	41.3

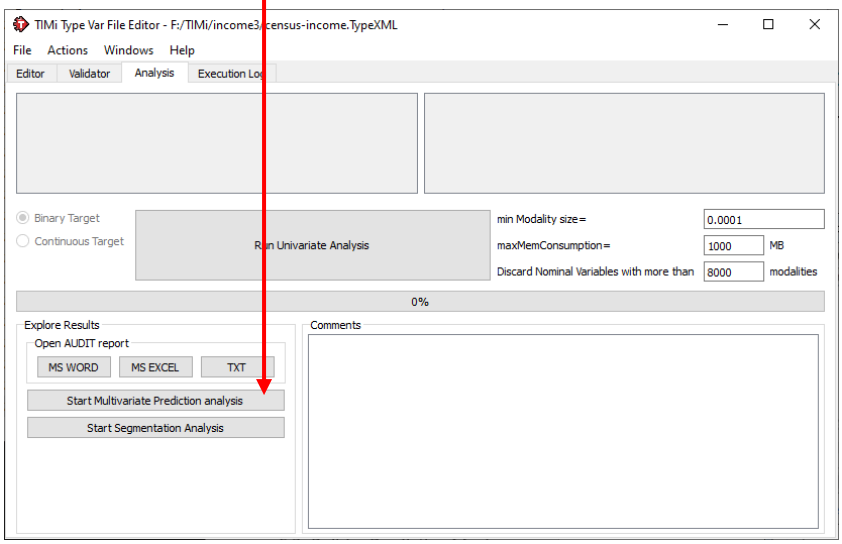
It appears that the “Males” have a lot more chances to be wealthy than “Females”.

We could say this chart contradicts the hypothesis of gender equality quite a bit!

We can continue to analyze the dataset, column by column, for a long time. In one way or another, almost every variable has some impact and could be (more or less) well suited to predict if a person is wealthy or not (i.e. to predict if a person is “*inside the target*”). In reality, only a small subset of the 42 columns is really needed to predict accurately the target. In this “demo” situation, we only have 42 columns but most real-life datasets often have hundreds or thousands of columns/variables. In the CRISP-DM methodology, a careful review of every variable is required prior to a predictive modeling exercise. With TIMi Modeler, while are careful review of *some* variables is still needed, the process tends to be much faster.

To figure out which variables are really important, the easiest way is to find out which can be included inside a predictive model.

Let us then proceed to the next step: open the “Config File editor”: click on the Start Multivariate Prediction analysis button:



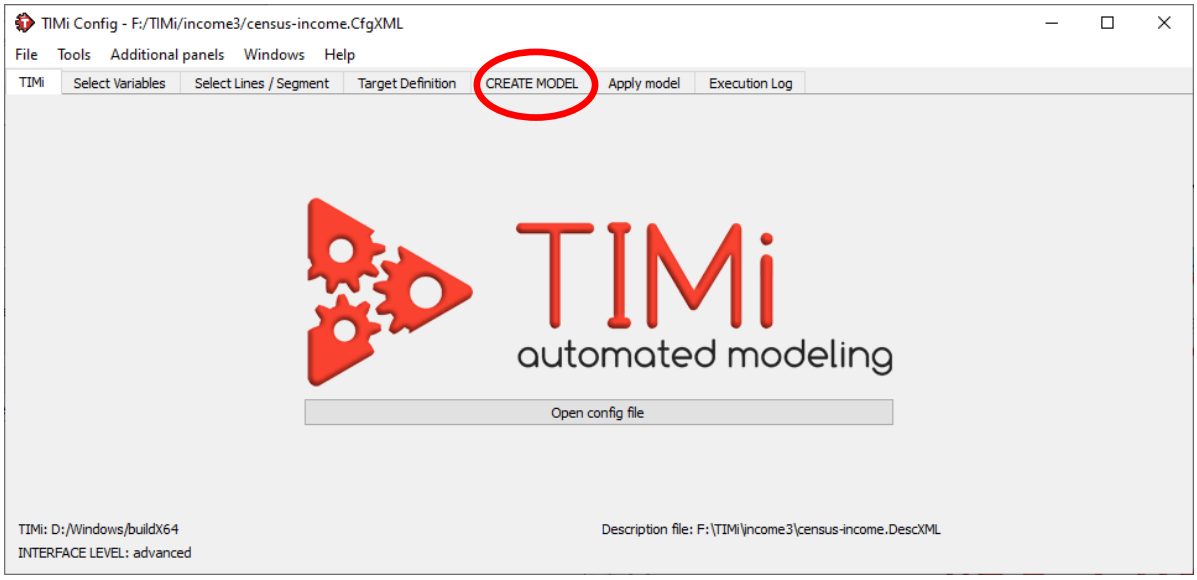
You can now close the “Type Var Editor” application and the “Microsoft Word” application (the file “census-income_AUDIT.doc” was still visible inside Microsoft Word).

6.3. Build a Predictive Model: The Config File Editor (Step 3)

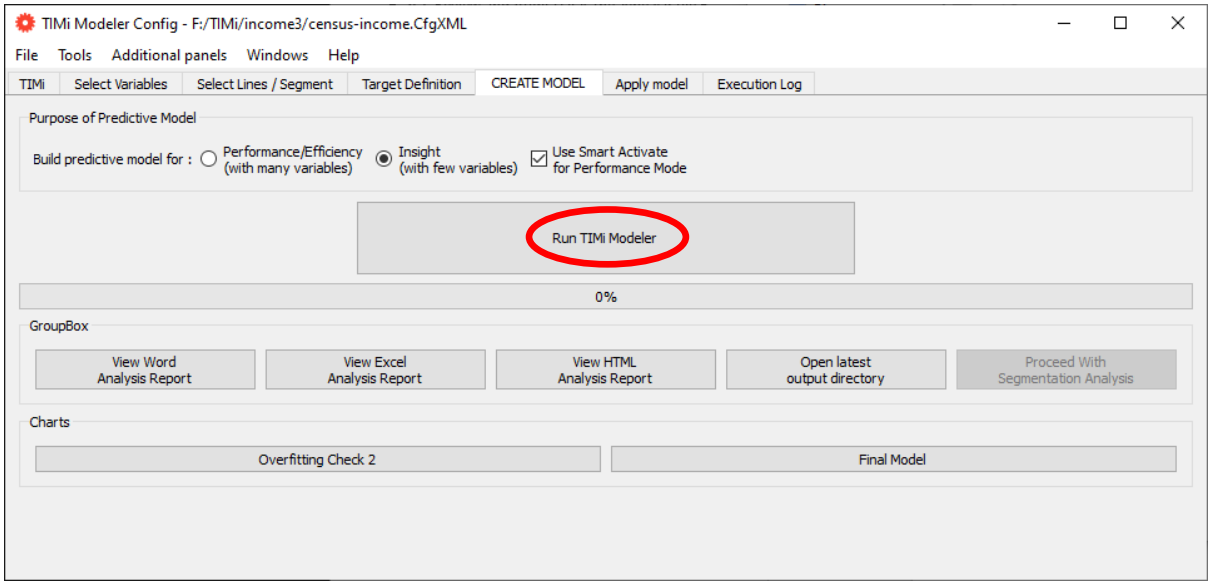
There are three ways to start the Config File Editor:

1. Create a new Config File using TIMi Modeler and open it directly after creation (this is normally what you just did)
2. Click the Create Predictive Models button in the main menu of TIMi.
3. Double-click on a “*.CfgXML” file inside an explorer window.

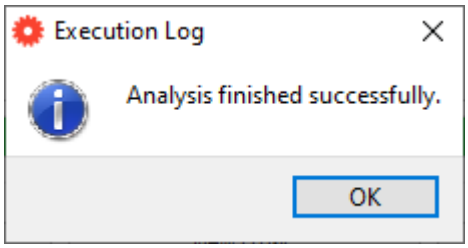
Here is an illustration of the “Config File editor”. You can directly Click on the “CREATE MODEL” tab:



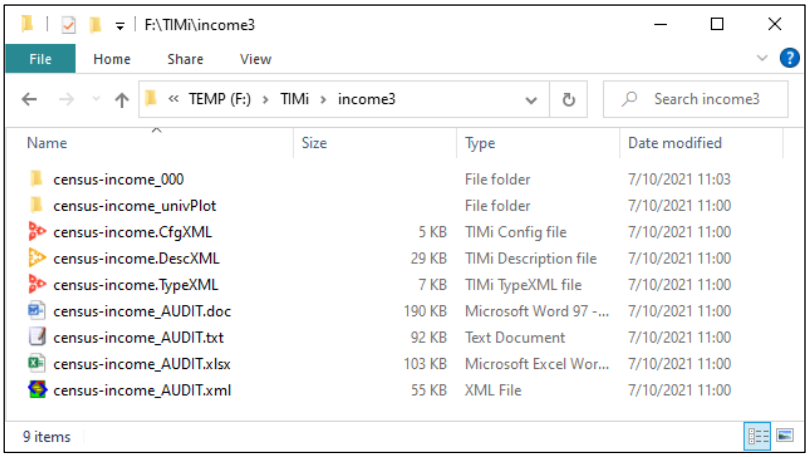
… then click on the “Run TIMi Modeler” button:



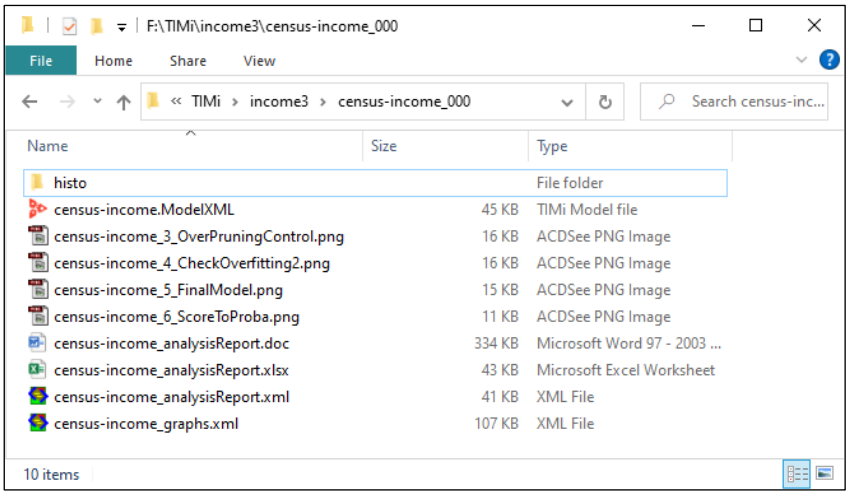
After a few seconds of computation, you should see this message:



The working directory now contains one additional subdirectory named “census-income_000”: see the illustration below:

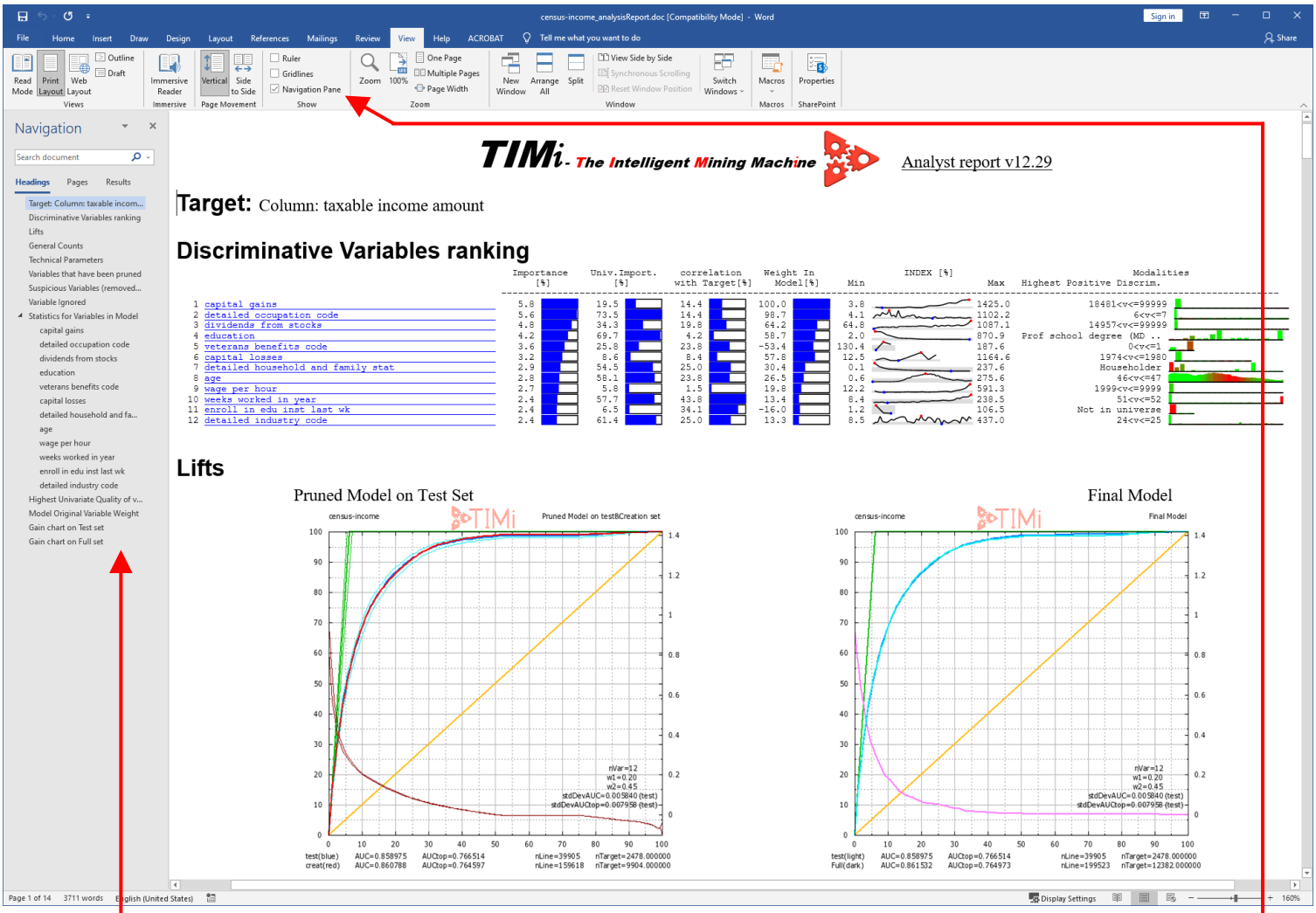


And now a small explanation about what just happened: We just asked TIMi Modeler to build a predictive model using all the lines and all the columns of our dataset. This newly created predictive model will be used to compute, for each individual in the dataset, the probability to be “inside the target” (to have an income level above $50K). The results of all these computations are stored inside the directory “demoIncome_000”:



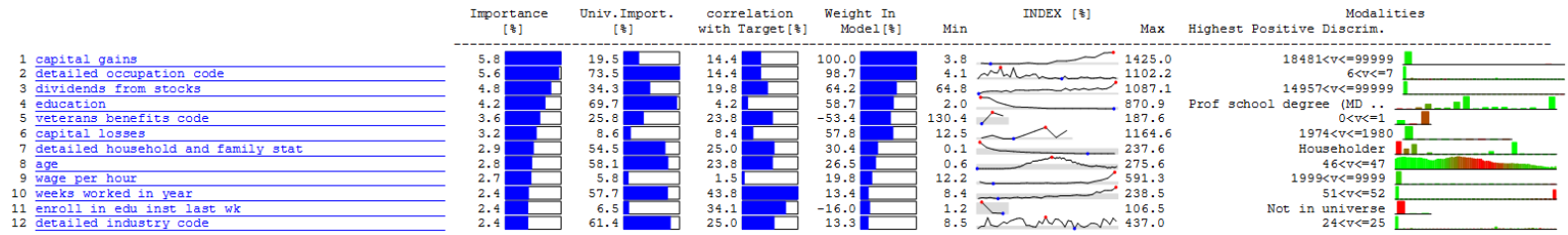
The “census-income_analystReport.doc” and “census-income_analystReport.xml” files have the same content. The only difference between these files is: « *Most of the time, the .doc file is the one you are interested in. The “.xml” is only useful for the automatic generation of commercial reports*. ».

Let us have a look at the “census-income_analystReport.doc”:



To see the left panel inside Microsoft Word here , click in the toolbar on the category menu “View” and then check the checkbox named “Navigation pane”:

Let us see what are the best columns used to predict accurately the target. During a predictive modeling exercise, the columns of the dataset are sometimes referred as “variables”. Our predictive model does not use all the “variables” or “column” of our dataset to compute its prediction: i.e. It only uses 12 columns out of 42. Theses 12 columns have each a different importance: Some columns are *very* useful to predict the target, some other are nearly useless. Click in the “Document Explorer” panel on “Discriminative Variables ranking”. See illustration:

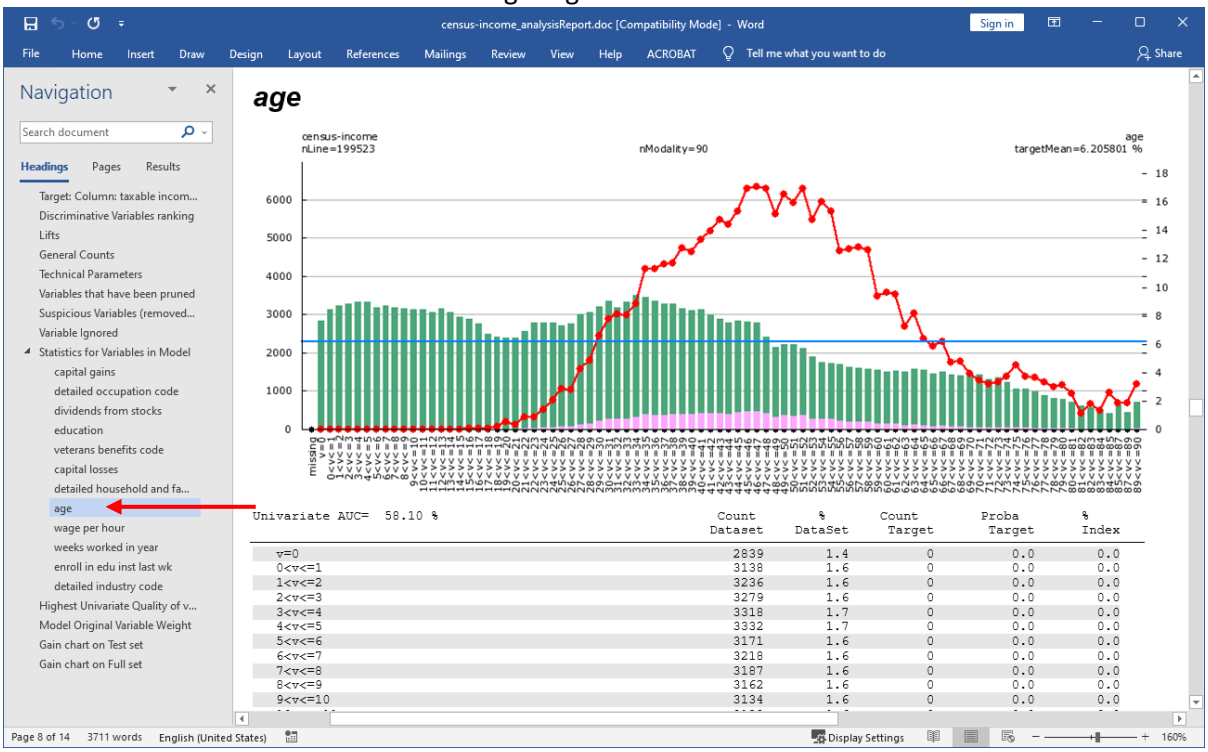


		Importance [%]	Univ.Import. [%]	correlation with Target[%]	Weight In Model[%]	Min	INDEX [%]	Max	Modalities Highest Positive Discrim.
1	capital gains	5.8	19.5	14.4	100.0	3.8		1425.0	18481<v<=99999
2	detailed occupation code	5.6	73.5	14.4	98.7	4.1		1102.2	6<v<=7
3	dividends from stocks	4.8	34.3	19.8	64.2	64.8		1087.1	14957<v<=99999
4	education	4.2	69.7	4.2	58.7	2.0		870.9	Prof school degree (MD ..
5	veterans benefits code	3.6	25.8	23.8	-53.4	130.4		187.6	0<v<=1
6	capital losses	3.2	8.6	8.4	57.8	12.5		1164.6	1974<v<=1980
7	detailed household and family stat	2.9	54.5	25.0	30.4	0.1		237.6	Householder
8	age	2.8	58.1	23.8	26.5	0.6		275.6	46<v<=47
9	wage per hour	2.7	5.8	1.5	19.8	12.2		591.3	1999<v<=9999
10	weeks worked in year	2.4	57.7	43.8	13.4	8.4		238.5	51<v<=52
11	enroll in edu inst last wk	2.4	6.5	34.1	-16.0	1.2		106.5	Not in universe
12	detailed industry code	2.4	61.4	25.0	13.3	8.5		437.0	24<v<=25

We can see that one of the best column to predict the target is “capital gains”. That’s not a surprise!

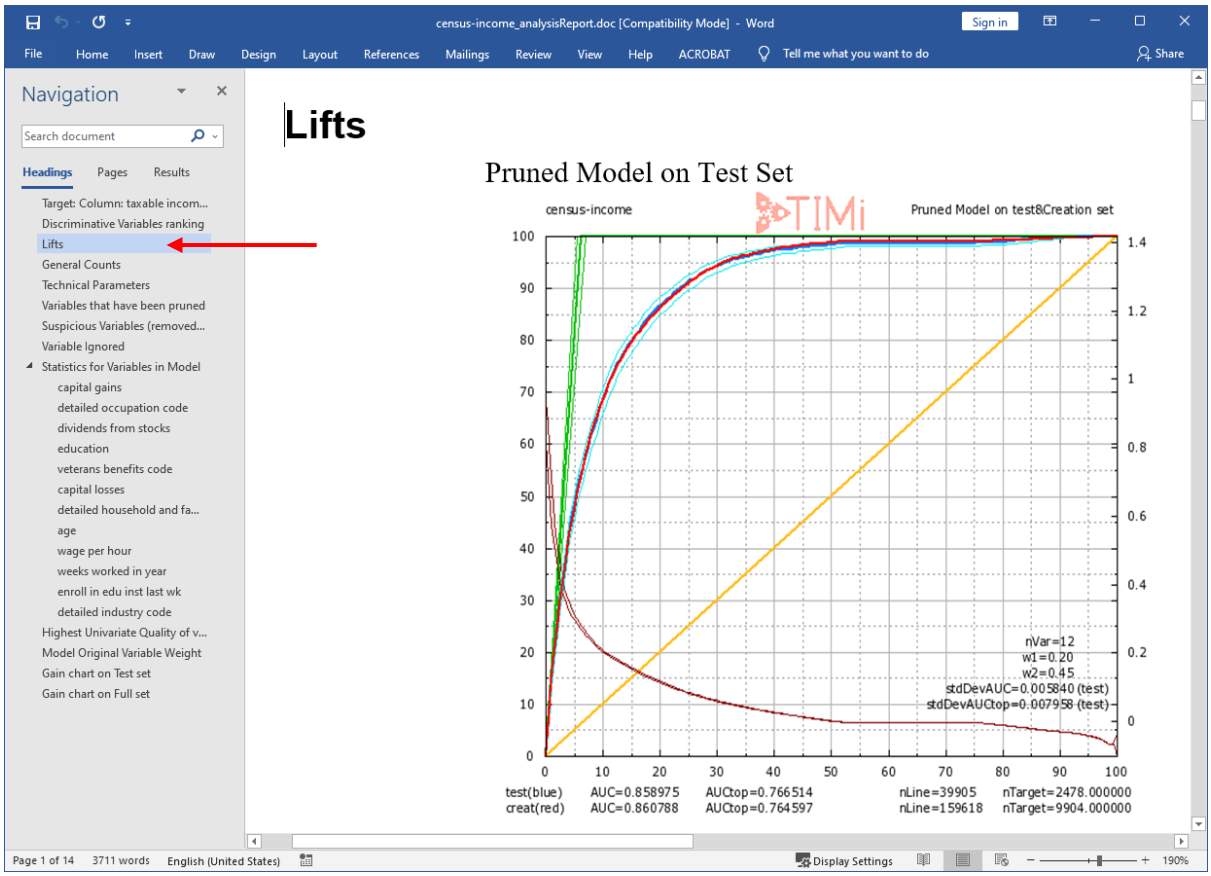
The variable “age” has a non-negligible importance. One possible interpretation of this high importance is that you have to wait to have the right age to be wealthy. Young people have very little chance to be wealthy.

Let us have a closer look at the variable “age” again:



Note that TIMi Modeler has now decided that it needed to cut/discretize the variable “age” in 90 “bins” or “modalities” to obtain a good prediction accuracy (compare this with the chart of the “age” generated for the AUDIT report: we had only 15 modalities/bins).

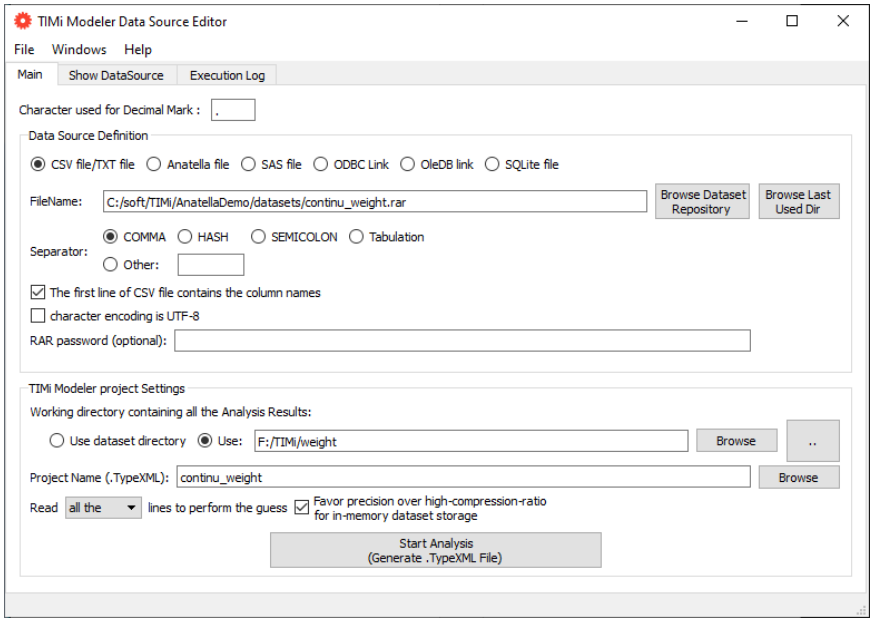
Let us have a look at the quality of prediction of the model produced by Modeler! On the “Document Map”, click on the “Lifts” section. Look at the dark blue line.



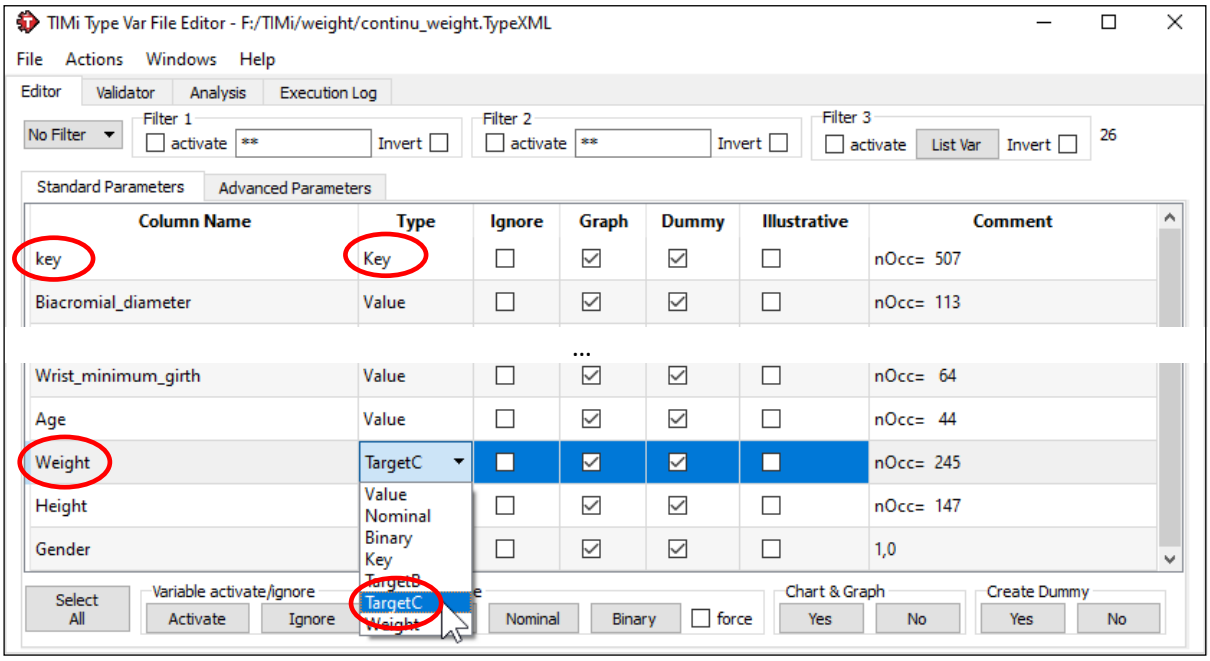
The model produced by TIMi Modeler has a AUC= 85.8% (on the test set). This is quite good. You can now close the “Microsoft Word” application (the file “demoIncome_analystReport.doc” was visible inside Microsoft Word).

7. An example for a Continuous target

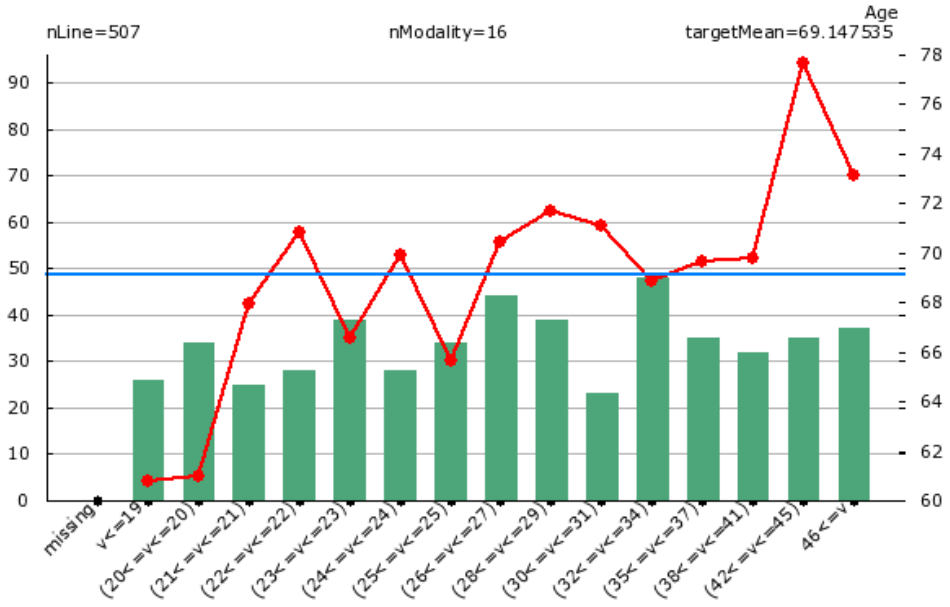
We will now build another kind of predictive model: a predictive model that predicts a continuous value, in opposition to the other sections above where the target was binary (0/1). We will study the dataset named “continu_weight”. This dataset contains on each line a different person. For each person (on each line of the dataset), we measured various body circumference lengths. Based on these lengths, we want to predict the weight of each individual. Let us start by defining where is located our dataset! Run the Datasource Editor and enter the following:



Press the button to generate a new .TypeXML and edit the newly generated .TypeXML file. Enter the following:



We just selected the column “Weight” as our “Continuous Target” and the column “key” as our primary key. Run the “Univariate Analysis”. You obtain a “weight_AUDIT.doc” file. Open this file and look at the variable “age”:

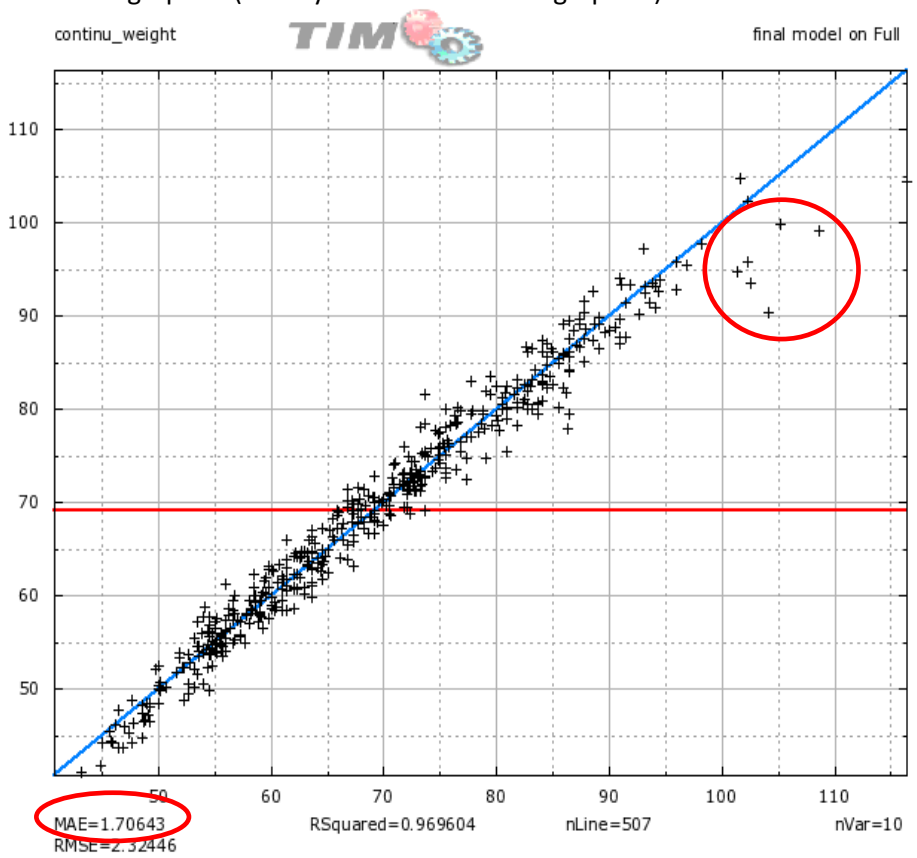


What do we see on the chart above? When we look at the green bar, we see that there are inside our dataset 50 people that are between 32 and 34 years old. Looking at the red line, we see that these same persons have a weight of 68.9 Kg in average.

Open the newly generated “.CfgXML” file editor and run the multivariate analysis: click on the “Run TIMi Modeler” button. Open the final Analyst report inside MS Word. You should get something like this:

Interestingly, the model does NOT use the variable “gender”. Some physicians claim that Males have bigger and heavier bones than Women and thus are inherently heavier. The results presented here contradict this hypothesis since the variable “gender” is not used by the predictive model to do the prediction (Note that this effect could be the result of the relatively small size of our sample: i.e. our sample could be slightly biased).

Let us look at the first graphics (usually named “dot cloud graphic”):



Each point in the “dot cloud graphic” represents a prediction for one individual. The X-coordinate of a point represents the real value to predict. The Y-coordinate of a point is the prediction value. If all predictions are perfects, all the points are aligned on the blue diagonal. We see on the graphic above (look at the red circle) the following: For the people that have very high weight (above 95 Kg), the predictions are too low: the prediction under-evaluate the real weight.

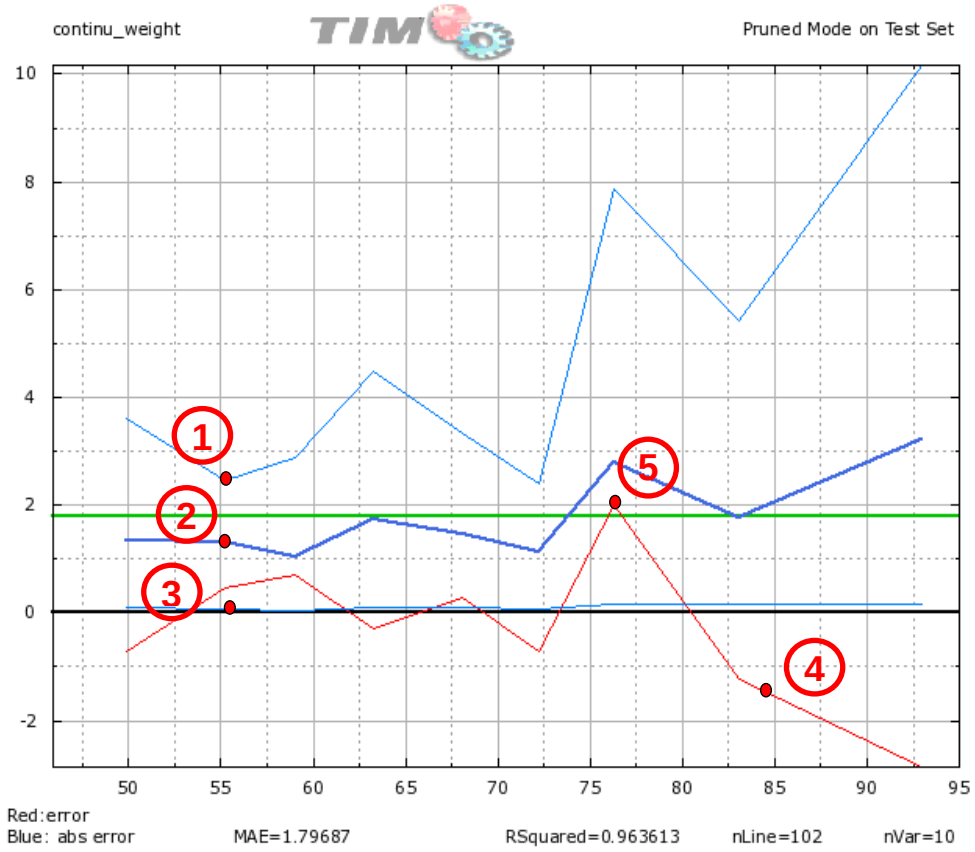
When we apply the model on the whole (full) dataset, the “Mean Absolute prediction Error” (MAE) is 1.70 Kg. It means that, in average, the predictions are wrong by an offset of 1.70 Kg.

On the graphic, you also see a red horizontal line centered at 69 Kg. This is the average weight of the people inside the dataset. On some difficult problems (not this one), it is impossible to predict anything. In such difficult situation, TIMi Modeler delivers a predictive model that always predicts the same constant value: i.e. The average of the target (and then, all the points will be aligned on the red horizontal line).

To summarize:

- when the predictions are **easy** to do: All the points are aligned on the **blue diagonal** line.
- when the predictions are **difficult** to do: All the points are aligned on the **red horizontal** line.

Let us look at the second graphic:



What do we see on the graphic above?

- For the people around 55 Kg (on the X-axis), the absolute prediction error is (95% of the time) between 2.6 Kg (point **1**) and .04 Kg (point **3**). For these persons, in average, the absolute prediction error is 1.3 Kg (point **2**).
- For the people above 80 Kg, the red line is clearly below the zero axis (point **4**). It means that the predictions for these persons have strong negative errors: the predictive model under-evaluate the real weight of the persons (this is confirmed by the analysis we did on the previous "dot-cloud" graphic).
- For the people around 77 Kg, the red line is clearly above the zero axis (point **5**). It means that the predictions for these persons have strong positive errors: the predictive model over-evaluate the real weight of the persons.
- The dark blue line is higher on the right than on the left: it means that the errors of prediction are higher for heavy persons.

8. Optional steps for binary targets

8.1. Deployment of a predictive model on unknown datasets

Usually, predictive models for binary target are used for applications like customer acquisition, cross-selling, up-selling, churn prevention, etc.

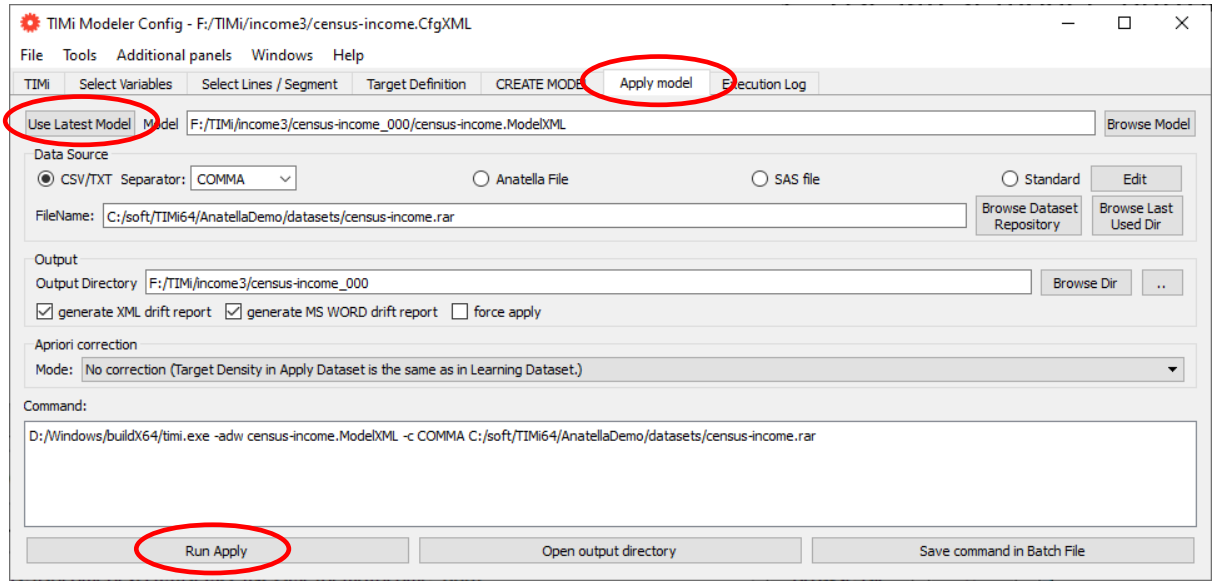
The predictive model that we built in section 7 of this document is typically used for fraud prevention. Using this model, we want to detect people that cheated when they answered to the question "*Do you earn more than $50K per year?*" on their IRS form. The taxation is higher for people that earn

more than 50.000 dollars per year and thus people are always tempted to cheat and say: "*I earn **less** than 50.000 dollars per year*".

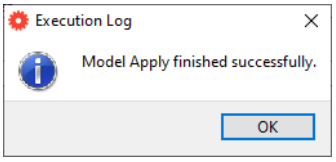
Using our predictive model, we can compute the probability to be inside the target (i.e. "*to earn more than 50.000 dollars per year*") for all the persons in a given dataset. The persons that are not inside the target (i.e. the persons that said: "*I earn **less** than 50.000 dollars per year*") and for which the predictive model still says that they have a very high probability to be inside the target (i.e. "*to earn more than 50.000 dollars per year*") are suspicious (Don't worry! This will be re-explained in more detail on an example later).

Let us apply the model on a dataset and find out the persons that are suspicious!

Re-open the "Config File editor", go to the "Apply Model" panel, click the "Use latest model" button and click the "Run Apply" button: See the illustration below:



After a short time, you should see:

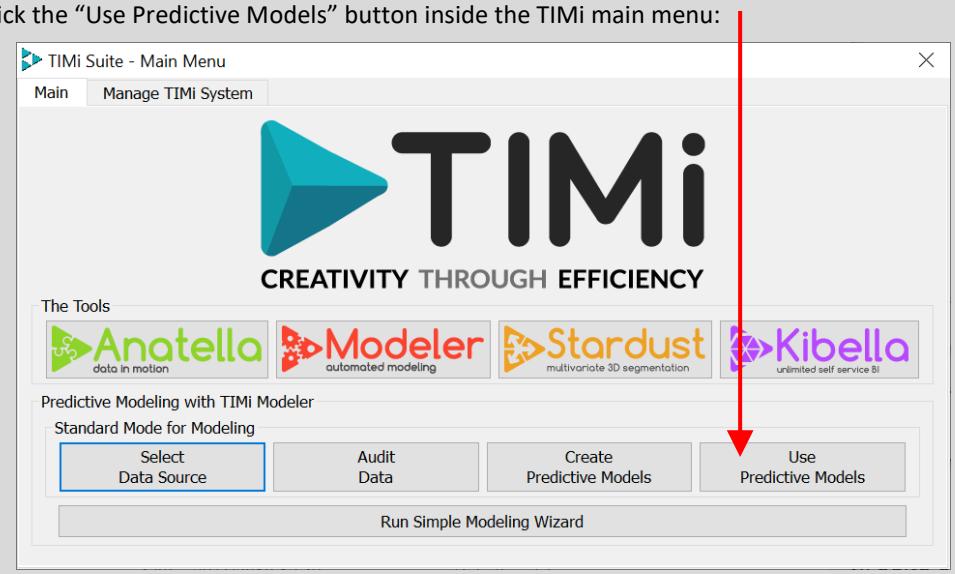


By default, the predictive model will be applied on the same dataset that was used to create the model (here: "census-income.rar"). Most of the time, you will have to change this default settings to apply your model on the latest version of your customer database. The datasets on which we apply our models are commonly named the "apply datasets" or "scoring datasets". Click the "Browse Dataset Repository" button to change it.

If you **only** have the predictive model file (the ".ModelXML" file), you can still apply directly your predictive model using the "Use Model" window.

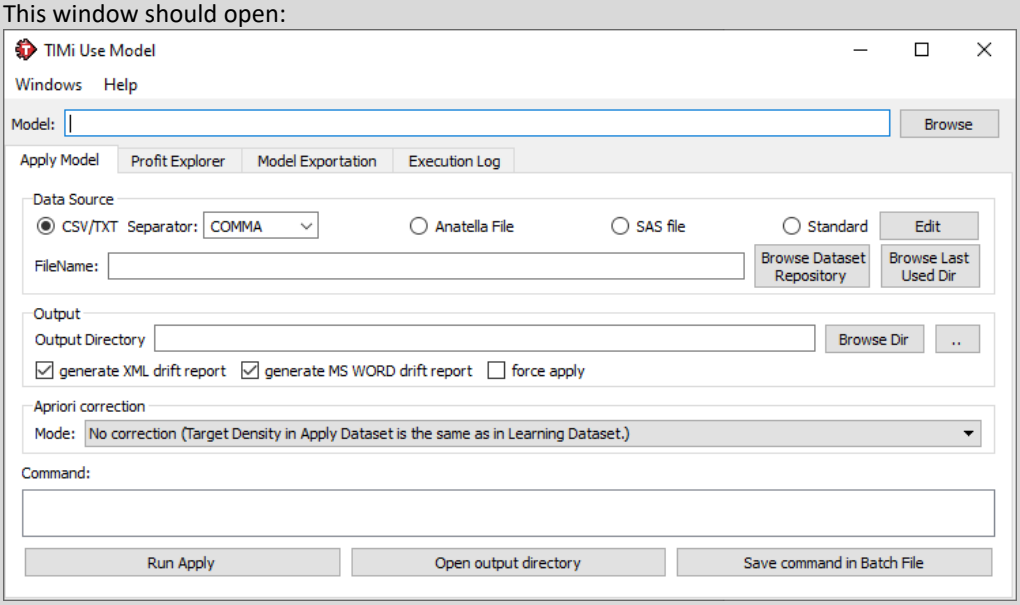
There are two ways to open the "Use Model" window

1. Click the "Use Predictive Models" button inside the TIMi main menu:

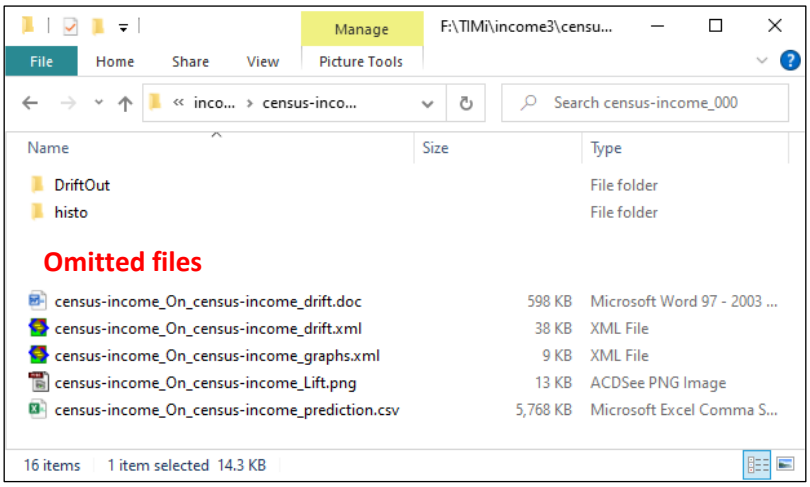


2. Double-click on your ".ModelXML" file inside the MS-File Explorer.

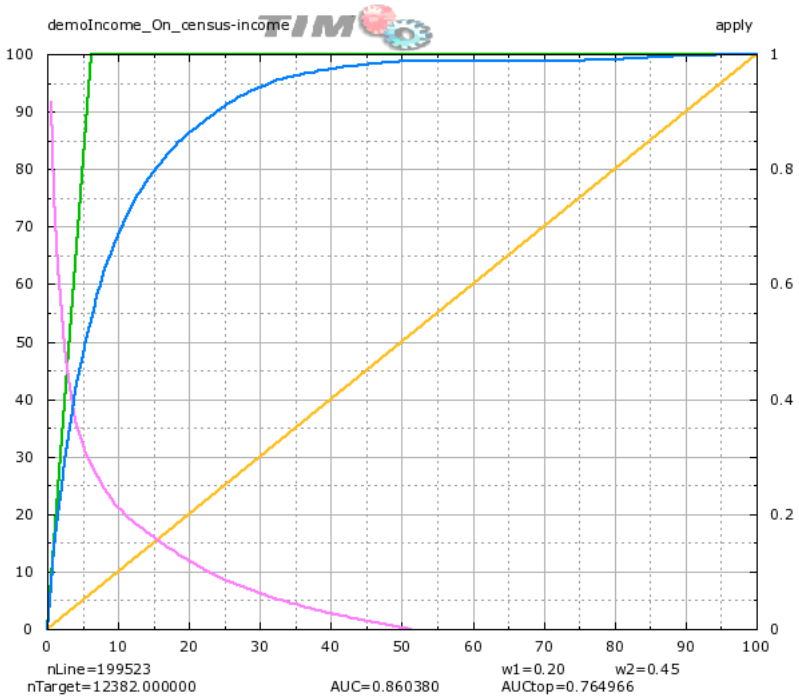
This window should open:



After “applying” your model on your (scoring) dataset, there are now five more files inside the output directory “demoIncome_000”:

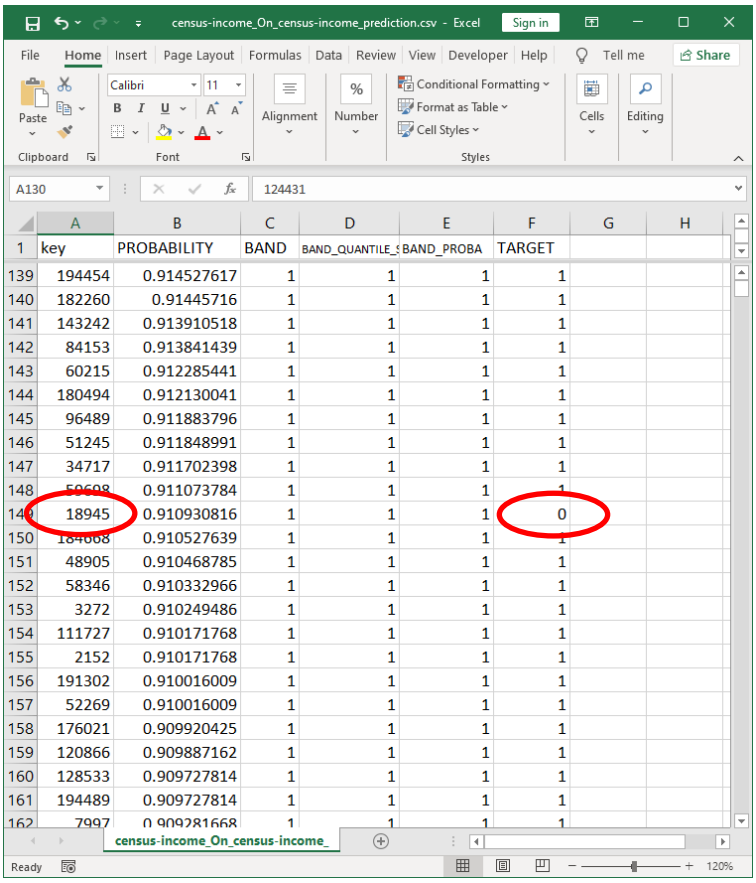


The “demoIncome_appliedOn_census-income_lift_apply.png” file is a graphics of the lift that represents the performance of the model on the specified dataset. Double-click the ‘.png’ file:



The “*_graph.xml” file can be deleted (it is an intermediate file to produce the “.png” file). Double click the “demoIncome_appliedOn_census-income_customerListApply.csv” file! (Microsoft Excel might complain that the file is too big but it doesn’t matter).

You should now see inside Microsoft Excel the following:



	A	B	C	D	E	F	G	H
1	key	PROBABILITY	BAND	BAND_QUANTILE_	BAND_PROBA	TARGET		
139	194454	0.914527617	1	1	1	1		
140	182260	0.91445716	1	1	1	1		
141	143242	0.913910518	1	1	1	1		
142	84153	0.913841439	1	1	1	1		
143	60215	0.912285441	1	1	1	1		
144	180494	0.912130041	1	1	1	1		
145	96489	0.911883796	1	1	1	1		
146	51245	0.911848991	1	1	1	1		
147	34717	0.911702398	1	1	1	1		
148	50698	0.911073784	1	1	1	1		
149	18945	0.910930816	1	1	1	0		
150	184668	0.910527639	1	1	1	1		
151	48905	0.910468785	1	1	1	1		
152	58346	0.910332966	1	1	1	1		
153	3272	0.910249486	1	1	1	1		
154	111727	0.910171768	1	1	1	1		
155	2152	0.910171768	1	1	1	1		
156	191302	0.910016009	1	1	1	1		
157	52269	0.910016009	1	1	1	1		
158	176021	0.909920425	1	1	1	1		
159	120866	0.909887162	1	1	1	1		
160	128533	0.909727814	1	1	1	1		
161	194489	0.909727814	1	1	1	1		
162	7997	0.909281668	1	1	1	1		

Each line of the Customer-List file represents a line of our “apply dataset”. Thus, in our case each line represents a person.

Look at the line 139: it means that the person with the primary key “194454” has 91.45% chances to be a target (i.e. has 91.45% chances to have an income level above $50K). You also see that the person “194454” was a target inside our dataset (i.e. the cell F139 is “1”).

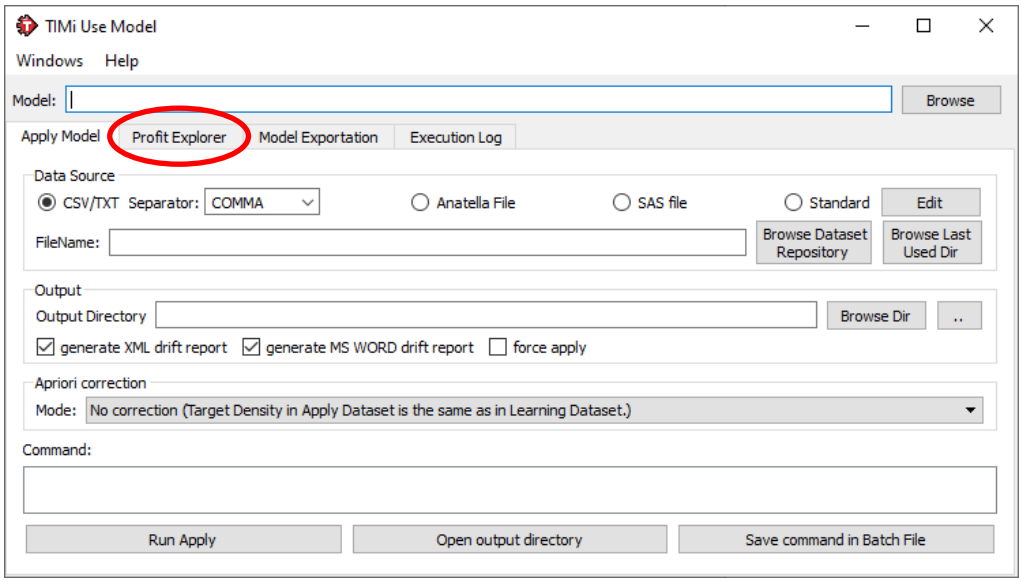
The person with a primary key “18945” on line 149 is **NOT** marked as a target inside the “apply dataset” (see the cell F149). Strangely, the TIMi Modeler predictive model has estimated that this same person has a probability of 91.09% to be a target. This is very suspicious. My guess is that the person “18945” declared a low income (below $50K) to avoid paying heavy taxes. He is cheating.

8.2. Profit Explorer

Most of the time, Predictive modeling techniques are used to acquire new clients during a marketing campaign. Let us now assume that we want to sell a product to a list of potential clients. The target to predict is “*Will this client buy my product?*”

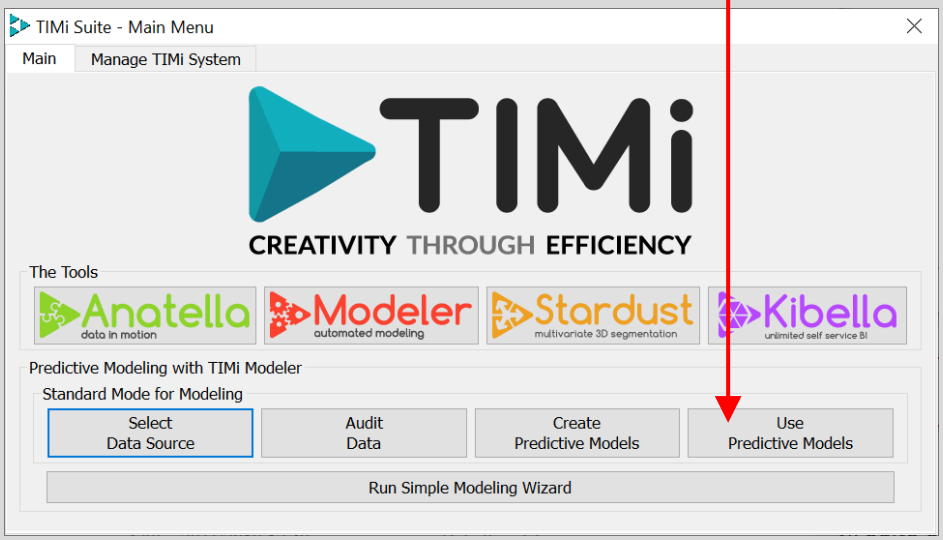
Inside this context (and referring to the Microsoft Excel view above), the prospect “18945” does not have my product (yet), because the cell F149 is 0, but he has 91.09% chances to buy my product if I propose my product to him. He is a very good potential customer. We should contact him! We should also contact all the other persons that have a “high” purchase probability. But what is a “high” purchase probability? Is it 90% or 70%? Obviously, there is a choice to be made. The “Profit Explorer”

application will help you to make this choice. Open now the “Profit Explorer”. Simply click on the “Profit Explorer” tab inside the “TIMi Use Model” Window:



There are two ways to open the “TIMi Use Model” window

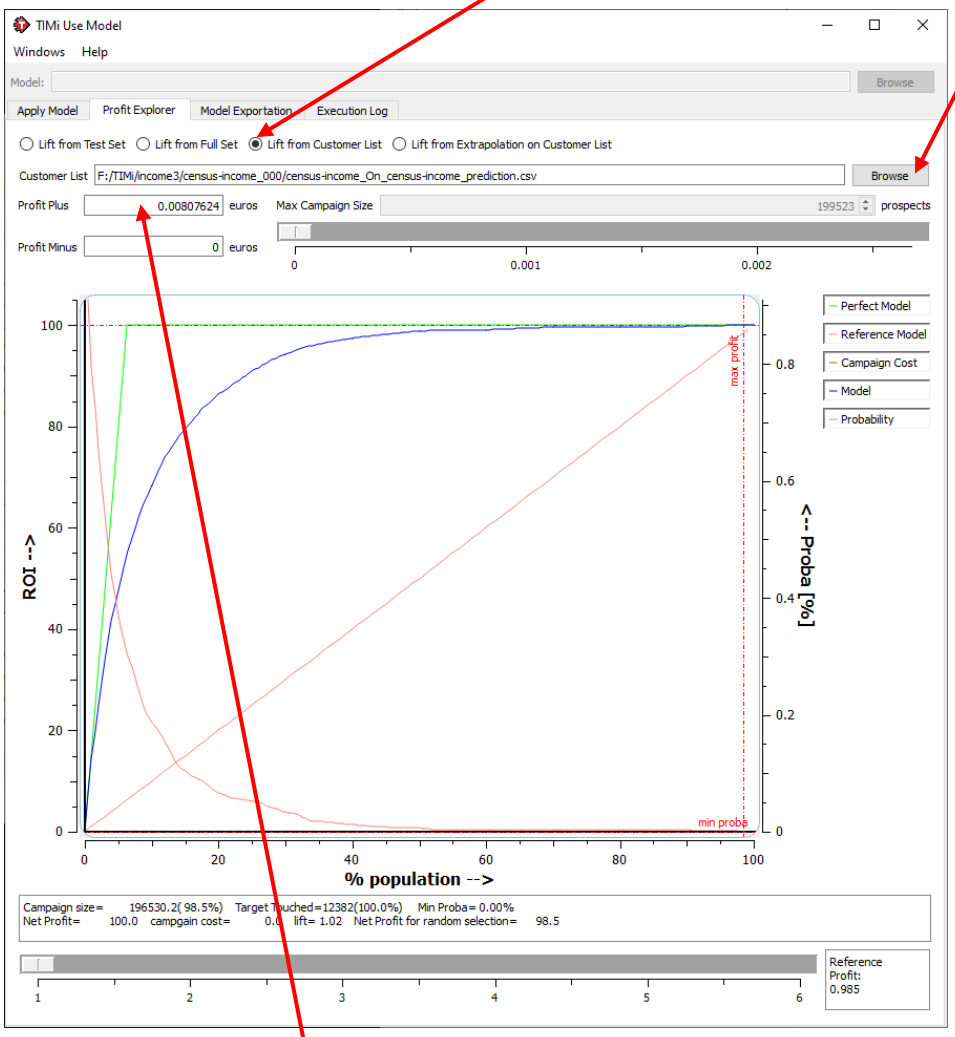
1. Click the “Use Predictive Models” button inside the TIMi main menu:



2. Double-click on your “.ModelXML” file inside the MS-File Explorer.

Let’s now assume that you have already applied your predictive model on a dataset that contains all the persons susceptible to be mailed for your marketing campaign. You just obtained a Customer-List file. The first lines of this file contain the “good” prospects that will be contacted (see the previous section on how to generate the required Customer-List file).

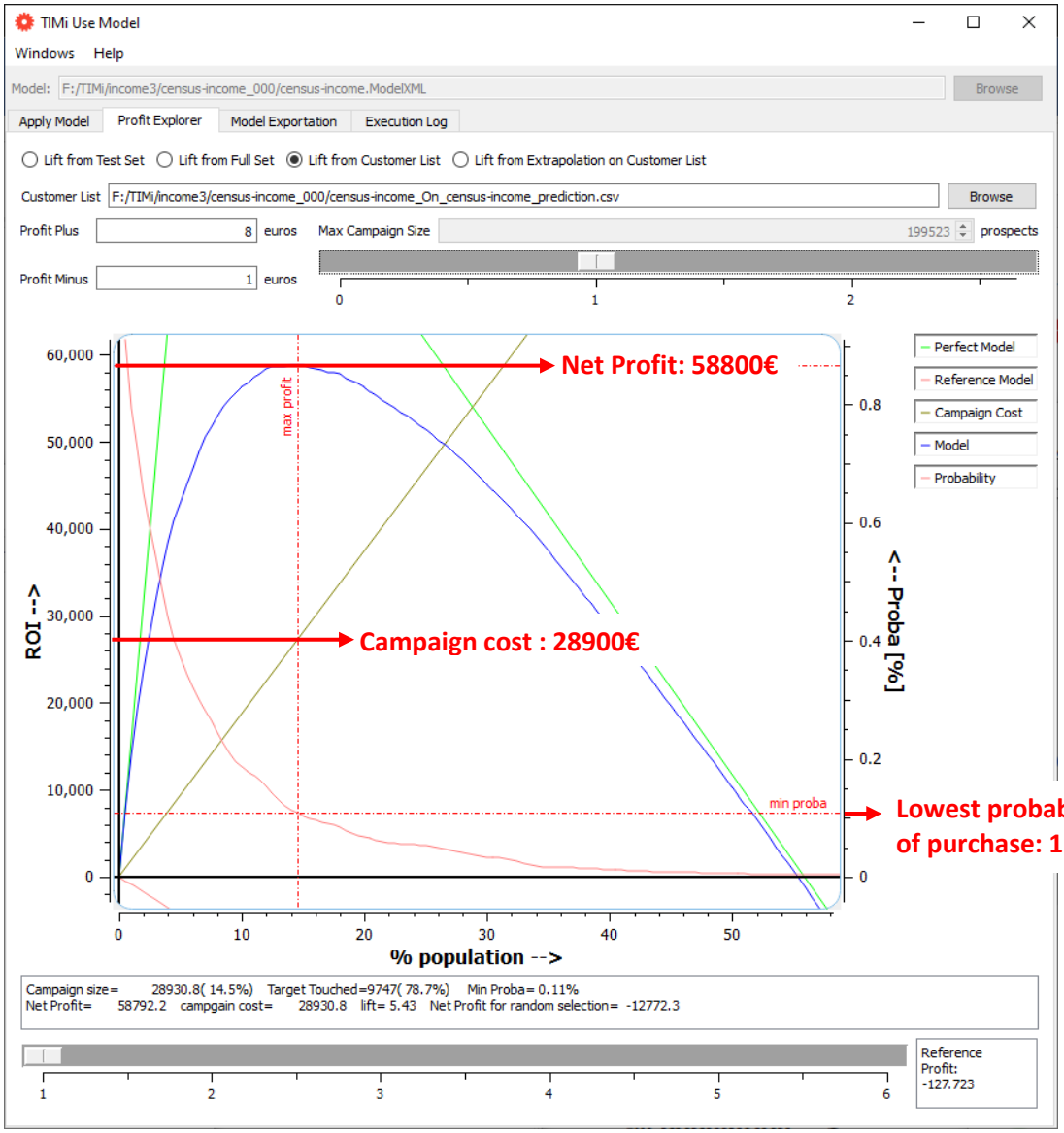
Let us analyze our Customer-List file to discover how many prospects we should contact! As an example, we will use the Customer-List generated at the previous step (on the "census-income" dataset). Click on the radio-button "Lift from Customer List": and thereafter on the "Browse" button: and select our Customer-List file. You should see:



There are now two parameters to adjust:

1. The "ProfitPlus" parameter (ROI): This parameter is the amount (in euros, dollars, pounds,…) of money that you will receive if the prospect accept to purchase your product.
2. The "ProfitMinus" parameter (Contact Cost): This parameter is the contact cost. It's the amount (in euros, dollars, pounds,…) of money that you have to spend to contact a prospect.

Let us assume that ProfitPlus="8 euros" and ProfitMinus="1 euros". Enter these two values inside the "Profit Explorer". You should now see:



You can adjust precisely, in real-time, the ProfitPlus & ProfitMinus parameters (use the slider!) and the campaign size to match your desire. The statistics in the lower part of the window are linked to the position of the optimal campaign size. It is amusing to see that, in the example above, if you don't use any predictive technique at all and simply contact randomly 28930 persons, you will actually **loose** around 12172 euros instead of **gaining** 58800 euros!

9. Conclusions

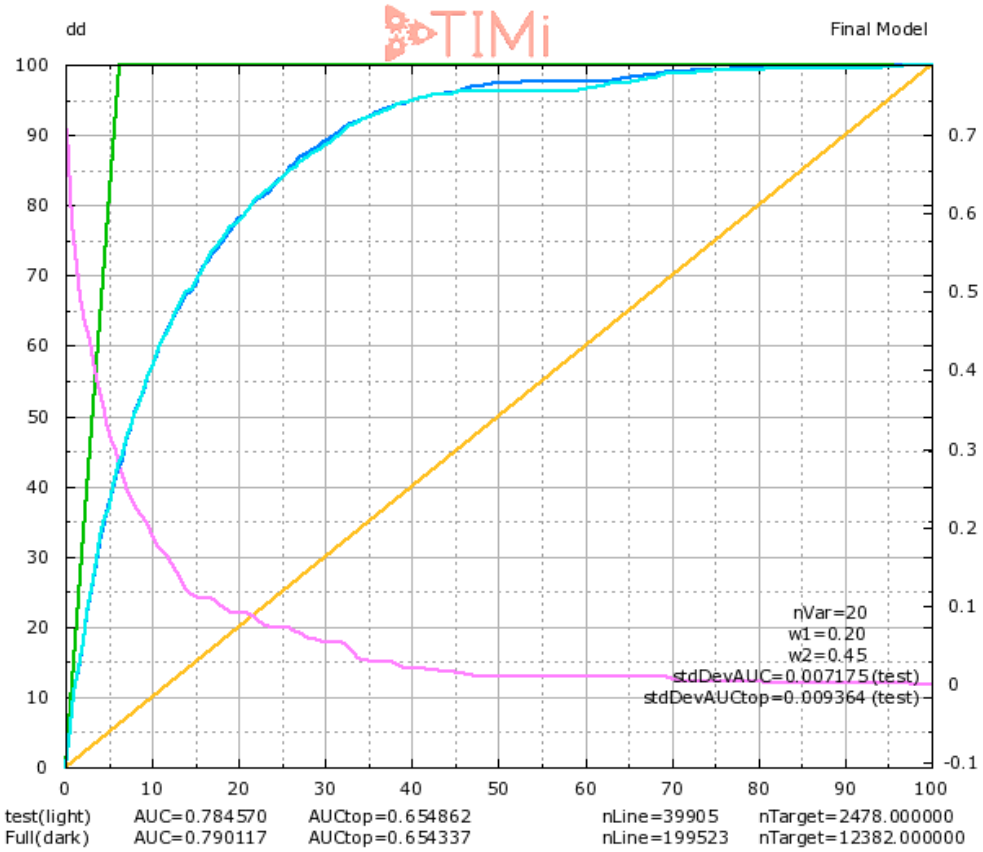
This concludes our (very) brief introduction to TIMi Modeler.

I hope you enjoyed the ride!

I invite you to play a little more with TIMi Modeler to answer the following question: « *Can we predict if a person will be wealthy at adult age based only on information available when he leaves school?* »
Open the "Config file editor", ignore all variables except these 4 variables:
education
major industry code
marital stat

sex
…, switch to “performance mode”, and then run TIMi Modeler.

We obtain the following lift:



The performance of this last model should normally be low because we only used columns that contain very “generic” information. But since the target is very easy to predict, we still get a decent lift. I can even say that we get a pretty good lift compared to the optimal lift generated previously! Does this means that our whole (financial) life is decided for us once we leave school? Is it a possible interpretation of these results? I hope not! What’s your opinion? I now leave you thinking about these serious reflections about the *Life, the Universe and Everything* (don’t forget: the answer is ‘42’![1]).
See you!

Frank Vanden Berghen, CEO and Founder of TIMi.

[1] See “The Hitchhiker's Guide to the Galaxy” by Douglas Adams - September 27, 1995.

Appendix A: The compressed-CSV-file format

The compressed-CSV-file format is one of the most efficient way to store a dataset that will be used inside TIMi modeler. The abbreviation CSV means “comma-separated value”. TIMi Modeler reads natively datasets compressed in RAR, ZIP, and GZ. The preferred compression format is RAR.

An example: The following table

key	Is taxable income amount above 50K ?	age	education	marital stat	race	sex	country of birth	weeks worked in year
1	0	73	High school graduate	Widowed	White	F	USA	0
2	0	58	Some college but no degree	Divorced	White	M	USA	52
3	0	18	10th grade	Never married	Asian	F	Vietnam	0
4	0	9	Children	Never married	White	F	USA	0
5	0	10	Children	Never married	White	F	USA	0
6	0	48	Some college but no degree	Married-civilian	Indian	F	USA	52
7	0	42	Bachelors degree(BA AB BS)	Married-civilian	White	M	USA	52
8	1	28	High school graduate	Never married	White	F	USA	30
9	0	47	Some college but no degree	Married-civilian	White	F	USA	52
10	0	34	Some college but no degree	Married-civilian	White	M	USA	52
11	0	8	Children	Never married	White	F	USA	0
13	0	51	Some college but no degree	Married-civilian	White	M	USA	52
14	1	46	High school graduate	Divorced	White	F	Columbia	52
15	0	26	Bachelors degree(BA AB BS)	Never married	White	F	USA	52
16	0	13	Children	Never married	Black	F	USA	0
17	0	47	Bachelors degree(BA AB BS)	Never married	White	F	USA	52
18	0	39	10th grade	Married-civilian	White	F	Mexico	0
19	0	16	10th grade	Never married	White	F	USA	0
20	0	35	High school graduate	Married-civilian	White	M	USA	49

… is equal to a “.csv” file containing:

```
key#taxable income amount#age#education#marital stat#race#sex#country of birth#weeks worked in year
1#0#73#High school graduate#Widowed#White#F#USA#0
2#0#58#Some college but no degree#Divorced#White#M#USA#52
3#0#18#10th grade#Never married#Asian#F#Vietnam#0
4#0#9#Children#Never married#White#F#USA#0
5#0#10#Children#Never married#White#F#USA#0
6#0#48#Some college but no degree#Married-civilian #Indian#F#USA#52
7#0#42#Bachelors degree(BA AB BS)#Married-civilian #White#M#USA#52
8#1#28#High school graduate#Never married#White#F#USA#30
9#0#47#Some college but no degree#Married-civilian #White#F#USA#52
10#0#34#Some college but no degree#Married-civilian #White#M#USA#52
11#0#8#Children#Never married#White#F#USA#0
13#0#51#Some college but no degree#Married-civilian #White#M#USA#52
14#1#46#High school graduate#Divorced#White#F#Columbia#52
15#0#26#Bachelors degree(BA AB BS)#Never married#White#F#USA#52
16#0#13#Children#Never married#Black#F#USA#0
17#0#47#Bachelors degree(BA AB BS)#Never married#White#F#USA#52
18#0#39#10th grade#Married-civilian #White#F#Mexico#0
19#0#16#10th grade#Never married#White#F#USA#0
20#0#35#High school graduate#Married-civilian #White#M#USA#49
```

The first line contains the name of the columns. Each field (inside a line) is separated from the next field using a ***separator character***. In the example above, the ***separator character*** is ‘#’. In a classical “.csv” file the ***separator character*** is the ‘,’.

After RAR compression, the size of the file “census-income.csv” is reduced to 4.1 MB. It’s a compression ratio of more than 95% (Usually dataset files are very easy to compress). TIMi Modeler is able to work directly with datasets in their compressed form (in other word, you never have to decompress yourself your datasets). The economy in hard drive space is substantial. If your dataset files are stored on a remote network drive, the compression mechanism of TIMi Modeler allows to reduce substantially the network bandwidth used when manipulating your datasets.

See the document “DataPreparation_churn.doc” to have more information about the construction of a good Dataset.