**Life Insurance Promotion** *

Here we have an Excel-based dataset containing information about credit card holders who have accepted or rejected various promotional offerings. We are trying to infer relations about the likelihood of different card holders in accepting or rejecting life insurance. Information about each customer’s age, income, gender and whether the customers took advantage of the life insurance promotion or not are included in the dataset as shown in Table 1. In order to produce the decision tree, we are using the RapidMiner software. RapidMiner supports many different data mining techniques, but we will focus only on decision trees here.

<table>
<thead>
<tr>
<th>Income Range</th>
<th>Life Insurance Promo</th>
<th>Credit Card Insurance</th>
<th>Sex</th>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>40-50,000</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>45</td>
</tr>
<tr>
<td>30-40,000</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>40</td>
</tr>
<tr>
<td>40-50,000</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>42</td>
</tr>
<tr>
<td>30-40,000</td>
<td>Yes</td>
<td>Yes</td>
<td>Male</td>
<td>43</td>
</tr>
<tr>
<td>50-60,000</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>38</td>
</tr>
<tr>
<td>20-30,000</td>
<td>No</td>
<td>No</td>
<td>Female</td>
<td>55</td>
</tr>
<tr>
<td>30-40,000</td>
<td>Yes</td>
<td>Yes</td>
<td>Male</td>
<td>35</td>
</tr>
<tr>
<td>20-30,000</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>27</td>
</tr>
<tr>
<td>30-40,000</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>43</td>
</tr>
<tr>
<td>30-40,000</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>41</td>
</tr>
<tr>
<td>40-50,000</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>43</td>
</tr>
<tr>
<td>20-30,000</td>
<td>Yes</td>
<td>No</td>
<td>Male</td>
<td>29</td>
</tr>
<tr>
<td>50-60,000</td>
<td>Yes</td>
<td>No</td>
<td>Female</td>
<td>39</td>
</tr>
<tr>
<td>40-50,000</td>
<td>No</td>
<td>No</td>
<td>Male</td>
<td>55</td>
</tr>
<tr>
<td>20-30,000</td>
<td>Yes</td>
<td>Yes</td>
<td>Female</td>
<td>19</td>
</tr>
</tbody>
</table>

Table 1  A data set for Life Insurance Promotion

Here are the steps needed to be completed.

Data Files needed: Life_Insurance_Promotion.xls.

Complete the following steps:

Start the RapidMiner software. After it loads, the welcome page will open as in Fig 1, choose **New Process** to start a new project.

---

*Acknowledgement: The tutorial was created with the help of my graduate assistant, Nezar Hussain.*
1. Fig 2 shows the new process page that should open. Below is a brief description of important areas/buttons in RapidMiner. (if the default view disappeared, select “View” the click “Restore Default Perspective”)
2. First we need to specify the source of the data that we want to use for our decision tree. In our case the data is in an Excel sheet, so we need to choose the operator that imports from Excel files. There are several ways to find the operator we are looking for:
   - Know which group the operator belongs to, and just expand the list until we find it.
   - Start typing what we think the operator might be called into the filter field under the Operators tab, so that the software will start showing suggestion as you type (i.e. there is no need to hit enter for the search string to register, similar to how Google search works now).

The operator we need is called Read Excel and can be found under the Import group, under the Data subgroup. Once the operator is located, we can either double click it and it will be added automatically into the process area, or we can drag the operator and drop it where we need. Both approaches would produce the same results in this situation because it is our first operator to add. Fig 3 shows what the operator looks like after it is placed in the process area. (Note that if the little circle in the Read Excel Operator is red, it will not turn into green until you complete the input process [see Step 8]).

![Fig 3](image.png)

3. For the operator to extract the data correctly we need to specify certain parameters, such as where the Excel file exists, which sheet(s) to extract from, columns, cell range etc. We can enter those parameters manually, however since we are still learning how to use the software, it’s a better idea to use the Import Configuration Wizard button to specify our parameters in 4 steps.

4. Fig 4 shows the first step which is to locate where the Excel file is located on the hard drive. (Note: You can specify bookmarks where you save your data so that you can go straight to that specific folder, rather than search for it every time you use an import operator). Once the required Excel file is located, Life_Insurance_Promotion.xls, click on Next.

![Fig 4](image.png)
5. Step 2 is where we choose the appropriate sheet and cell range. Our data resides in the Life Insurance Promotion sheet, and columns A through E are needed. The cells should be highlighted as shown in Fig 5. Click Next.

6. For step 3, simply make sure is there under the Annotation column, in the column titles row (Income Range, Life Insurance Promotion, etc.) as shown in Fig 6. Click Next.
7. In step 4, is where we choose which attribute (column) we are trying to forecast based on the data we have. In our case that is “Life Insurance Promotion”. To do so, we go to the Life Insurance Promotion column, to the cell that says attribute, which is above our actual data, and from the drop-down menu change it from attribute to label. Fig 6a) & b show how the wizard window should look like before we click on Finish to return to our process view. (Note: Step 4 also allows us to choose how many values the data takes, such as binomial, polynomial, numeric, etc. This is done automatically for you if your data set contains less than or equal 100 rows. If your data is larger, it is recommended you uncheck the box that says Preview uses only the first 100 rows, and click on the Reload data button, then click on Guess value types button. You can also uncheck the boxes above a certain column if you do not want to include it in your process/calculations. This is beyond our example’s need but can become helpful later to make sure to specify the data type to better specify how the software processes the data.)
8. Now that the “Read Excel” preferences are set up, we need to select the Decision Tree Operator to actually produce the decision tree we desire. First you need to choose “Modeling” Operator (move away from Import) then look for Decision Tree Operator. This operator is found under the Classification and Regression group, under the Tree Induction subgroup. Drag the operator of “Decision Tree” onto the “Process Area” (see Fig. 7) so that it follows the Read Excel and the result as shown in Fig 8. Here we don’t need to change any of the Decision Tree Operator parameters. (Note: On each operator there is a little circle like an LED that tells you if the operator is receiving the proper or necessary inputs it needs or not. Usually it should be either “green”, meaning it already ran and is working well, or “yellow”, meaning that the operator hasn’t run yet but seems to functioning properly. If you have a red triangle with an exclamation mark, that usually means something is wrong with your setup and needs to be rectified before proceeding further, otherwise your model will probably not run.) Note that if the new Decision Tree operator is with red, you need to re-connect them on their “out” and “tra” (“training”) until the red turns into yellow.

![Drag “Decision Tree” onto “Process Area”](image)

![Fig 7](image)

![Fig 8](image)
9. If you wish, you may click on “Process” and select “Decision Tree” (Fig. 9-a) and the result is shown in Fig. 9-b.

Fig 9-a)

Fig 9-b)
10. You should save your work constantly, but if you haven’t already now it is important to do so by clicking on the Save button. You should organize your processes under folders, usually under My Repository and with meaningful names (e.g., Life Insurance Promotion) you can find later. Once you do so, click on the Play button, also known as the Run or resume process (Fig. 9-b), then you will get a series of prompt windows to save your process, view results, close old results if they existed, etc. and the “yellow” light turns into “green”. Answer them appropriately (Fig. 10-a and 10-b) until you get to the Results workspace as shown in Fig 10-c.

![Fig 10-a)](image.png)

![Fig 10-b)](image.png)
11. On the **Result Overview** tab, shown in Fig 10-c, you can click on the result process that just ran to view a small preview of the results. To view the results in more detail (graphical mode), click on the **Tree** (Decision Tree) tab. **Graph View** shows the decision tree produced in its graphical form as the view name suggests. When a node has two or more colors that means that the data couldn’t be separated into distinct nodes in that instance, but the majority of the data led to that result. An example is the No node of the Credit Card Ins. node. The Graph view is shown in Fig 11.

Note: In Fig 11, we can see what are referred to as **nodes** and **leaves**. **The nodes are the gray oval shapes.** They are attributes which serve as good predictors for our label attribute. The leaves are the multicolored end points that show us the distribution of categories from our label attribute that follow the branch of the tree to the point of that leaf. We can see in this tree that **Age** is our best predictor of whether or not a customer is going to accept the Life Insurance promotion. If the Age is greater than 44, we see that their Life Insurance likely to be denied. If however, the Age is less than or equal to 44, then **Sex** becomes the next best predictor of the Life Insurance promotion category. If the Age is less than or equal to 44, **Credit Card Insurance** is another predictive attribute which forms a node, with Female adopting sooner than Male. This is seen on the branches for the two leaves coming from the Age node in Figure 11.
12. The Text View in Fig 12 shows the tree in a *textual* form, explicitly stating how the data branched into the Yes and No nodes. As mentioned earlier the No node of the Credit Card Ins. node weren’t all No’s in our data, we had 3 No’s and 1 Yes. The Annotations page is beyond the scope of this exercise.

**Tree**

```
Age > 44: No {No=3, Yes=0}
Age ≤ 44
    | Sex = Female: Yes {No=0, Yes=6}
    | Sex = Male
    |    | Credit Card Ins. = No: No {No=3, Yes=1}
    |    | Credit Card Ins. = Yes: Yes {No=0, Yes=2}
```
Extra Features and details

- The parameters of the Read Excel Operator shown in Fig 13 below can be edited without the need of re-running the import configuration wizard if need be.

1. **Excel file**: Defines where the Excel source file is located. If this needs to be changed, you can click on the folder button next to the excel file field and locate the required file.
2. **Sheet number**: Defines which sheet in the Excel file contains the data. Note that even if the sheet has a name, Rapid Miner numbers them according to their sequence.
3. **Imported cell range**: As the field name suggests defines the cell range that contains the data.
4. **First row as names**: Check this if the first row of your data contains the column headers.
5. **Annotations**: If you click on the Edit List (…) button, a window will open that allows you to define certain rows as either Name, aka Header, Comment, if you have row(s) that you don’t want to include in your data, or Unit, where certain row(s) contain the units for your data.
6. **Data format**: Defines the date format for your data, leave as is if you want the system default.
7. **Time zone**: Defines the time zone of the date fields, again leave as on default (SYSTEM) unless your data comes from another time zone and you want that to be reflected in your data.
8. **Locale**: Defines the language script that will be used for the data in the Excel file.
9. **Data set meta data information**: If you click on the Edit List (…) button, a window as shown in Fig 13b will open. This allows you to define the Meta data requirements for your data.
   - **Column index**: Contains the column number
   - **Attribute meta data information includes**:
     - First column is the column names (headers)
     - Second column allows you to choose whether to include that column as part of your data or not, by checking/ unchecking the tick box.
     - Third column defines the column data type.
     - Fourth column defines the column role.
10. **Read not matching values as missing**: Is checked by default and causes Rapid Miner to skip data rows that contain data that doesn’t match the specified Meta data.
11. **Data management**: Explanation to be added?
12. **Compatibility level**: Explanation to be added?