Potec Data Set Analysis

MVA project

Diego Garcia-Olano, Elsa Mullor
27/06/2014
## Content:

I – Problem Dataset and Description ...............................................3  
II- Pre-processing.................................................................................3  
III- MCA .............................................................................................5  
IV - Clustering ................................................................................... 12  
V – Prediction ..................................................................................... 15  
VI – Conclusion .................................................................................. 19  
VII – Appendix .................................................................................. 20  

1. Catdes results .................................................................................... 20  
2. Prediction errors with different processing of outliers ......................... 23  
2. R code ............................................................................................. 24
I – Problem Dataset and Description

For our problem we decided on the POTEC dataset, which is based on the Adult dataset that can be found at [https://archive.ics.uci.edu/ml/datasets/Adult](https://archive.ics.uci.edu/ml/datasets/Adult) in the UCI Machine Learning Repository. Also known as the "Census Income" data set, the data set contains 32561 individuals information along 15 variables taken from the 1994 US Census data.

The task at hand then is to predict whether an individual’s income exceeds $50,000 dollars per year. The binary target variable “target” contains values of either “<=50K” to denote the individual makes less than or equal to 50,000 dollars a year or “>50K” denoting they make more than that amount. The target is fairly imbalanced as only 24% of the population makes more than 50 thousand dollars a year.

The dataset variables consists of the following variables and values for each:

- **fnlwgt**: continuous (a weight originally set by initial data handlers)
- **education**: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
- **education-num**: continuous (number of years of schooling)
- **marital-status**: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, etc.
- **relationship**: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
- **race**: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
- **sex**: Male, Female
- **capital-gain**: continuous (per year)
- **capital-loss**: continuous (per year)
- **hours-per-week**: continuous (per week)
- **native-country**: United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Indonesia, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, Trinidad&Tobago, Peru, Hong, Holand-Netherlands.

II- Pre-processing

We take an initial look at our mostly categorical data and notice that three of our variables ("workclass", "occupation" & "native.country") contain some values of “?”, so we set a new level of “level_NA” for each and map those values to that.

After that we run an algorithm for outlier detection using initial mahalanobis distances between individuals as compared with a robust derived mahalanobis distance calculation between points, and obtain the following plot. This outlier detection is made using only the continuous variables.
We determine that there are thirty-eight outliers, located in the upper right portion of the prior plot. As the Mahalanobis distance should follow a Chi-Square distribution (with 5 degrees of freedom according to the number of continuous variables), we set as outliers the points for which the robust mahalanobis distance is higher than the 97.5%-quantile of this distribution. The red lines show this value.

These outlier individuals have in common that they all make over 50k a year, but more importantly that they all answered that there “capital gains” per year was 99999 which is irregular for that variable. From here on out, we will assign these individuals with a very low weight so that their capital gains responses don’t unduly influence the analysis.

We next discretize (“fnlwgt”, “education.num”, “capital.gain”, “capital.loss” “hours.per.week”) into quartiles. We also discretize the “age” variable into 5 groups (under 25, 26 to 34, 35 to 49, 50 to 62, 63 and up).

In the end we are left we with a dataset as follows:

```r
> summary(potec)

age  workclass  fnlwgt  education  education.num
age:25 and under: 6411  Private  :22696  HS-grad :10501  education.num [1,9]: 14734
age:26 to 35: 8514  self-emp-not-inc: 2541  some-college: 7291  education.num [9,10]: 7921
age:36 to 49: 10374  Local-gov: 2093  Bachelors : 5355  education.num [10,12]: 2449
age:50 to 64: 5726  Level-NA: 1826  Masters : 1723  education.num [12,16]: 8067
age:65 and up: 1336  self-emp-inc: 1116

marital.status  occupation  relationship  race  sex
Divorced: 4443  Prof-specialty: 4140  Husband: 13193  Amer-Indian Eskimo: 311  Female: 10771
Married-spouse-absent: 418  Adm-clerical: 3770  Own-child: 5068  Other: 271
Never-married: 10683  Sales: 3650  Unmarried: 3446  White: 27816
Separated: 1025  Other-service: 3295  Wife: 1568
Widowed: 983  (Other): 9541

capital.gain  capital.loss  hours.per.week  native.country  target
capital.gain 0: 29849  capital.loss 0: 31042  hours.per.week [1,40]: 7763  United-States: 29170  <=50K: 24720
capital.gain (0,1]: 2712  capital.loss [0,4,36]: 1519
hours.per.week 40: 15217  Mexico: 643  >50K: 7841
hours.per.week (40,45]: 2442  Level-NA: 583
hours.per.week (45,99]: 7139  Philippines: 108
Germany: 137
Canada: 121
(Other): 1709
```
From our data summary, we can see the most occurring individuals are between 36 and 49, work for private employers, have at least a high school education, are husbands, white, work 40 hours a week, consider the US their native country and make less than 50 thousand dollars a year.

III- MCA

Now all of our variables are factors, and so is the target. It is a natural idea to perform an MCA on this dataset. Besides, this will allow us to detect non-linear relationship between variable. As we have a lot of individuals and quite many modalities too, it would be a more adapted method than applying a PCA.

As we only have 14 variables and we assume that they all play a role to predict the target, we will only set the target variable as illustrative. So we have 14 active variables. We will also set the weights of the MCA as 1 for all individuals except the outliers whom will be assigned a weight of 0.00001.

The followings are some plots of the MCA results.

The first plot represents all individuals and all variables (active and illustrative), it is very loaded and we cannot really extract information from it except from the global distribution and some outlier modalities.
The second plot below is easier to read. It represents the variables according to their correlations (cos square) to the dimensions. We can see that the most relevant modalities are “education”, “education.num” and “occupation” (further from the center). Those three variables are also the most correlated with the second dimension. On the other hand “relationship” and “marital.status” are highly correlated with the first dimension. The “target” variable is also very close to the axis of the first dimension. Anyway to interpret the distribution of individuals, the latent concepts and the clustering, we are more interested in the distribution of modalities.

The next plot representing all active variables is quite difficult to read. We can easily detect outlier modalities but the more central ones are overlapping. To have a better understanding we will plot only the variables that contributed most to the dimensions.
The following plot shows the 20 variables that contributed most to the dimensions. We can already notice some pattern in the distribution of modalities. Like the opposition of “female” and “male” or the curve described by the number of studying years. Also the worked hours per week seem to follow a straight line along the first dimension. Very young people would be in the middle right of the plot, whereas highly educated ones will be at the top left.

![MCA factor map](image)

We can also plot the 20 variables that are most correlated to the dimensions. The two modalities of the target variable appear on this plot. They are distributed over the first dimension axis and we can see that the “left” part of the plot would be the one containing people earning more than 50K a year while the “right” part would be people earning less than 50K a year. The first dimension could also be the dimension of wealth.

Apart from the target the other variables are approximately the same as the ones which contributed most to the dimensions. We can keep reading information from this plot. For instance, we can see that the female modality is quite high on the second dimension but in the right part of the plot, while the male modality is on the contrary quite low on the second dimension but in the left part of the plot. So comparing those positions with the “education.num” evolution on the plot, and the target distribution, we can conclude that women are globally more educated but will make less money, while men are less educated but will globally make more money. We can also notice that people under 25 are very unlikely to make more than 50K a year.
The last plot only shows illustrative variables (here the target). We can notice that the modality "<=50K" is quite central (even if a little bit on the right) so it will be a common modality.
To obtain a more formal description of the dimensions we use the ‘dimdesc’ function. Here are the results we obtained (keeping only the p-values equal to zero). The firsts tables show the ‘$quali’ result, that is the relevant variables, whereas the second ones will show the ‘$category’ that is the most relevant modalities. We are more interested in the second one, although the variables can also give interesting information.

The first dimension is very discriminative (even if the percentage of variation explained is only 3.14%). First this dimension separates the target modalities. So the first dimension is the dimension of “wealth”. We can point out that a positive capital gain will also be negatively correlated with the first dimension (on the left) and that makes sense as we can assume that only wealthy people will make any kind of ‘capital gain’. The first dimension also separates the working hours modalities: on the right there will be people working less (between 1 and 40 hours a week) certainly including people who do not have a job. On the left there will be people working a lot (between 45 and 99 hours a week). Furthermore we can see that education is also distributed on this dimension, with two lower modalities on the right and higher education in the left part of the plot. This kind of information is directly related to the professional situation and what one is earning a year. But the first dimension also bears some ‘social’ information. First the age categories: in the right there will be younger people, while in the left there are middle age ones (36 to 64; so not retired people). This is coherent with the professional and financial discrimination (as young people often do not work, and we can expect that the salary of someone will reach a maximum when he is between 36 and 50). Finally we can notice that people who did not give their profession (occupation_level_NA) are in the right part of the plot too, as is the ‘female’ modality whereas high professions (managerial…) will be on the left part (negative correlation).

<table>
<thead>
<tr>
<th>Dim1 Qualitative</th>
<th>R2</th>
<th>p.value</th>
<th>Dim1 Categorical</th>
<th>Estimate</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>0.35</td>
<td>0.00</td>
<td>&lt;=50K</td>
<td>0.31</td>
<td>0.00</td>
</tr>
<tr>
<td>workclass</td>
<td>0.18</td>
<td>0.00</td>
<td>hours.per.week [1,40]</td>
<td>0.44</td>
<td>0.00</td>
</tr>
<tr>
<td>education</td>
<td>0.37</td>
<td>0.00</td>
<td>capital.gain 0</td>
<td>0.21</td>
<td>0.00</td>
</tr>
<tr>
<td>education.num</td>
<td>0.32</td>
<td>0.00</td>
<td>Female</td>
<td>0.25</td>
<td>0.00</td>
</tr>
<tr>
<td>marital.status</td>
<td>0.52</td>
<td>0.00</td>
<td>Own-child</td>
<td>0.42</td>
<td>0.00</td>
</tr>
<tr>
<td>occupation</td>
<td>0.41</td>
<td>0.00</td>
<td>occupation_level_NA</td>
<td>0.49</td>
<td>0.00</td>
</tr>
<tr>
<td>relationship</td>
<td>0.60</td>
<td>0.00</td>
<td>education.num (9,10)</td>
<td>0.28</td>
<td>0.00</td>
</tr>
<tr>
<td>race</td>
<td>0.05</td>
<td>0.00</td>
<td>education.num [1,9]</td>
<td>0.19</td>
<td>0.00</td>
</tr>
<tr>
<td>sex</td>
<td>0.24</td>
<td>0.00</td>
<td>age:25 and under</td>
<td>0.54</td>
<td>0.00</td>
</tr>
<tr>
<td>capital.gain</td>
<td>0.06</td>
<td>0.00</td>
<td>&gt;50K</td>
<td>-0.31</td>
<td>0.00</td>
</tr>
<tr>
<td>hours.per.week</td>
<td>0.28</td>
<td>0.00</td>
<td>hours.per.week (45,99)</td>
<td>-0.31</td>
<td>0.00</td>
</tr>
<tr>
<td>target</td>
<td>0.29</td>
<td>0.00</td>
<td>capital.gain (0,1)</td>
<td>-0.21</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Male</td>
<td>-0.25</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Husband</td>
<td>-0.59</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>occupation_Prof-specialty</td>
<td>-0.50</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>occupation_Exec-managerial</td>
<td>-0.44</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Married-civ-spouse</td>
<td>-0.53</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>education.num (12,16)</td>
<td>-0.43</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Prof-school</td>
<td>-0.77</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Masters</td>
<td>-0.56</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Doctorate</td>
<td>-0.73</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Bachelors</td>
<td>-0.36</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>age:50 to 64</td>
<td>-0.24</td>
<td>0.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>age:36 to 49</td>
<td>-0.25</td>
<td>0.00</td>
</tr>
</tbody>
</table>
The second dimension is clearly the dimension of education, with highest education on top and lower ones at the bottom. It also differentiates women and men. Some professions also appear as relevant, mostly because they are professions requiring high/low (if there is a positive/negative correlation) level of education.

<table>
<thead>
<tr>
<th>Dim2 qualitative</th>
<th>R2</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>workclass</td>
<td>0.08</td>
<td>0.00</td>
</tr>
<tr>
<td>education</td>
<td>0.64</td>
<td>0.00</td>
</tr>
<tr>
<td>education.num</td>
<td>0.62</td>
<td>0.00</td>
</tr>
<tr>
<td>marital.status</td>
<td>0.14</td>
<td>0.00</td>
</tr>
<tr>
<td>occupation</td>
<td>0.50</td>
<td>0.00</td>
</tr>
<tr>
<td>relationship</td>
<td>0.19</td>
<td>0.00</td>
</tr>
<tr>
<td>sex</td>
<td>0.18</td>
<td>0.00</td>
</tr>
<tr>
<td>native.country</td>
<td>0.09</td>
<td>0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dim2 category</th>
<th>Estimate</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>0.19</td>
<td>0.00</td>
</tr>
<tr>
<td>occupation_Prof-specialty</td>
<td>0.56</td>
<td>0.00</td>
</tr>
<tr>
<td>education.num (12,16]</td>
<td>0.34</td>
<td>0.00</td>
</tr>
<tr>
<td>Some-college</td>
<td>0.28</td>
<td>0.00</td>
</tr>
<tr>
<td>Prof-school</td>
<td>0.55</td>
<td>0.00</td>
</tr>
<tr>
<td>Masters</td>
<td>0.61</td>
<td>0.00</td>
</tr>
<tr>
<td>Doctorate</td>
<td>0.62</td>
<td>0.00</td>
</tr>
<tr>
<td>Bachelors</td>
<td>0.53</td>
<td>0.00</td>
</tr>
<tr>
<td>Male</td>
<td>-0.19</td>
<td>0.00</td>
</tr>
<tr>
<td>Husband</td>
<td>-0.26</td>
<td>0.00</td>
</tr>
<tr>
<td>occupation_Craft-repair</td>
<td>-0.34</td>
<td>0.00</td>
</tr>
<tr>
<td>education.num [1,9]</td>
<td>-0.45</td>
<td>0.00</td>
</tr>
<tr>
<td>HS-grad</td>
<td>-0.22</td>
<td>0.00</td>
</tr>
<tr>
<td>7th-8th</td>
<td>-0.44</td>
<td>0.00</td>
</tr>
<tr>
<td>5th-6th</td>
<td>-0.59</td>
<td>0.00</td>
</tr>
</tbody>
</table>

The third dimension will give a finer separation of age modalities. It opposes retired people (over 65 years old) to more middle age ones (26 to 49). It also opposes some education levels but in a different way as before. The ‘extremes’ level of education will be positively correlated while the ‘middle’ one (10 to 12 years of education) will be negatively correlated.

<table>
<thead>
<tr>
<th>Dim3 qualitative</th>
<th>R2</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>age</td>
<td>0.16</td>
<td>0.00</td>
</tr>
<tr>
<td>workclass</td>
<td>0.44</td>
<td>0.00</td>
</tr>
<tr>
<td>education</td>
<td>0.31</td>
<td>0.00</td>
</tr>
<tr>
<td>education.num</td>
<td>0.28</td>
<td>0.00</td>
</tr>
<tr>
<td>marital.status</td>
<td>0.16</td>
<td>0.00</td>
</tr>
<tr>
<td>occupation</td>
<td>0.47</td>
<td>0.00</td>
</tr>
<tr>
<td>relationship</td>
<td>0.20</td>
<td>0.00</td>
</tr>
<tr>
<td>sex</td>
<td>0.06</td>
<td>0.00</td>
</tr>
<tr>
<td>hours.per.week</td>
<td>0.07</td>
<td>0.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dim3 category</th>
<th>Estimate</th>
<th>p.value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>0.10</td>
<td>0.00</td>
</tr>
<tr>
<td>Own-child</td>
<td>0.24</td>
<td>0.00</td>
</tr>
<tr>
<td>Husband</td>
<td>0.15</td>
<td>0.00</td>
</tr>
<tr>
<td>occupation_level_NA</td>
<td>1.02</td>
<td>0.00</td>
</tr>
<tr>
<td>education.num (12,16]</td>
<td>0.25</td>
<td>0.00</td>
</tr>
<tr>
<td>education.num (9,10]</td>
<td>0.24</td>
<td>0.00</td>
</tr>
<tr>
<td>workclass_level_NA</td>
<td>0.76</td>
<td>0.00</td>
</tr>
<tr>
<td>age:65 and up</td>
<td>0.41</td>
<td>0.00</td>
</tr>
<tr>
<td>Female</td>
<td>-0.10</td>
<td>0.00</td>
</tr>
<tr>
<td>Unmarried</td>
<td>-0.35</td>
<td>0.00</td>
</tr>
<tr>
<td>education.num (10,12]</td>
<td>-0.58</td>
<td>0.00</td>
</tr>
<tr>
<td>Assoc-voc</td>
<td>-0.70</td>
<td>0.00</td>
</tr>
<tr>
<td>Assoc-acdm</td>
<td>-0.72</td>
<td>0.00</td>
</tr>
<tr>
<td>age:36 to 49</td>
<td>-0.22</td>
<td>0.00</td>
</tr>
<tr>
<td>age:26 to 35</td>
<td>-0.20</td>
<td>0.00</td>
</tr>
</tbody>
</table>

We can make a plot to summarize these ideas and to give a ‘visual’ of latent concepts, but the plot is a little bit loaded... We can point out that in a society where there will be sex-equality, the red line and the grey line would be perfectly perpendicular, (they should be mediatrices of each other) so the angle between them is an inequality measure (or its sine)! We can also point out that even if the ‘male’ modality is low on the second dimension while
the ‘female’ one is higher, it is not so clear that men are less educated than women because of the curved line of education levels.

Now we want to apply clustering on our data to create groups of people. For that we need to decide how many dimensions we want to keep from the MCA analysis. First we can plot the eigenvalues according to the dimensions. It total we have 109 dimensions. There are several rules we could use to select the dimensions. We can keep the dimensions for which the eigenvalue is higher than the mean. We chose to keep the dimensions for which the eigenvalue is higher than one over the number of active variables (in this case 14). With this rule we keep 51 dimensions.

In the following plot, the cut is indicated by the vertical gray line.
IV- Clustering

After deciding to keep the first 51 dimensions, we then re-run MCA (fixing the number of dimension kept to 51) and store only the significant dimensions into Psi. Because our dataset is relatively large, we progress with the following strategy to handle clustering:

We decide to first perform two separate runs of the k-means algorithm on Psi giving each the same arbitrarily large number of clusters (12 in our instance) to look for. We then do a hierarchical clustering upon the centroids of crossing these 2 kmeans partitions using the ward distance criterion.

Upon looking at the results, we do a barplot of the heights of the jumps between different clusterings, and decide that taking 5 clusters seems reasonable.
We then prune our results at a depth of 5, recalculate the centers of gravity given 5 clusters, and plot our findings which shows the clustering of individuals into 5 clusters.

This plot is not very well separated so we decided to run kmeans again (consolidation of the clustering) on our data this time giving it an input of 5 clusters to find beginning from the centroids obtained in the prior step.
This time the clusters are much better separated. We then run the function catdes on our clusters to get a description of what characterizes them (see Appendix for full listing).

cluster 1: **White Working Class Men (25%)**
- Male, White, Married, self employed, age 36 to 50
- Blue collar occupations: craft-repair, transport-moving, farming-fishing, etc
- High school, Some-college
- Works 45 to 99 hours a week

cluster 2: **Vocational School Tech-support (7%)**
- Education = Assoc-voc, assoc-admin

cluster 3: **The Haves (24.3%)**
- Occupation: Prof-speciality, Exec-managerial
- Education: Masters, Bachelors, Doctorate
- Works 45 to 99 hours a week
- Husbands, age 36 to 49
- Workclass: Local gov, State gov, self employed
- Capital gains and capital losses

cluster 4: **The Diverse Workers (25%)**
- Female, under 25, Black, Mexican,
- High School
- Working class occupation,
- Works under 40 hours a week

cluster 5: **The Unemployed (17.7%)**
- Female, occupation: NA, workingclass: NA
- Under 25, unmarried / divorced, Some college
V - Prediction

We chose to use prediction trees. Our first choice was to use C4.5 (which is a multi-way tree using theory of information: the splits are chosen according to entropy). Mostly because the dataset information included error rates obtained using standard algorithms and C4.5 had pretty good results. Additionally, we studied prediction trees in class. As a comparison of this model we also implemented a CART tree (Classification and Regression Tree). The two libraries used in R are RWeka and rpart respectively.

The validation protocol is the following:

First we will split our data in two parts (a training part that will contain 2/3 of the data and a testing part with 1/3). The training data will be used to optimize the parameters and to build the final model. The testing data are only used to compute the final validation error.

Then, for each model (C4.5 and rpart) we will optimize the parameters on the training data. To do so, we use 10 fold cross validation. Setting a range for our parameters, we evaluate the 10 fold CV error (the training and model are built with 9/10 of the data and the testing error is evaluated with 1/10) for every one of those and we will select the parameter that gives the lowest error.

The train function in R helps doing this loop.

The parameters we want to optimize are:

- For C4.5: C the Confidence threshold for pruning the tree
- For CART: cp the complexity parameter

We will optimize ‘C’ from 0 to 0.5 by 0.1, and ‘cp’ from 0.001 to 0.01 by 0.0005.

The following screenshot of R results describe the model and the accuracy obtained for the different parameters. In the C4.5 tree we are testing 10 parameters.

<table>
<thead>
<tr>
<th>C</th>
<th>Accuracy</th>
<th>Kappa</th>
<th>Accuracy SD</th>
<th>Kappa SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05</td>
<td>0.832</td>
<td>0.502</td>
<td>0.0126</td>
<td>0.0406</td>
</tr>
<tr>
<td>0.1</td>
<td>0.841</td>
<td>0.543</td>
<td>0.00953</td>
<td>0.0214</td>
</tr>
<tr>
<td>0.15</td>
<td>0.844</td>
<td>0.55</td>
<td>0.00965</td>
<td>0.0227</td>
</tr>
<tr>
<td>0.2</td>
<td>0.847</td>
<td>0.56</td>
<td>0.00995</td>
<td>0.0218</td>
</tr>
<tr>
<td>0.25</td>
<td>0.849</td>
<td>0.565</td>
<td>0.00987</td>
<td>0.0234</td>
</tr>
<tr>
<td>0.3</td>
<td>0.856</td>
<td>0.586</td>
<td>0.00972</td>
<td>0.0202</td>
</tr>
<tr>
<td>0.35</td>
<td>0.862</td>
<td>0.606</td>
<td>0.00892</td>
<td>0.0181</td>
</tr>
<tr>
<td>0.4</td>
<td>0.866</td>
<td>0.618</td>
<td>0.00755</td>
<td>0.0143</td>
</tr>
<tr>
<td>0.45</td>
<td>0.871</td>
<td>0.632</td>
<td>0.00777</td>
<td>0.0165</td>
</tr>
<tr>
<td>0.5</td>
<td>0.876</td>
<td>0.646</td>
<td>0.00711</td>
<td>0.0141</td>
</tr>
</tbody>
</table>

Accuracy was used to select the optimal model using the largest value. The final value used for the model was C = 0.5.

This is the accuracy curve for C4.5 model, when optimizing the parameter. We can see that the accuracy keeps increasing. The best parameter will be C=0.5.
Once we have selected the best parameter, we can build the model with the whole training data (no more 10 fold CV), and the best parameter.

From this model we are then able to compute the training and testing error.

```r
model.tree<-J48(target~., data=potec, subset=learn, control=Weka_control(C=0.5), na.action=NULL)
Train
> pred.learn<-predict(model.tree, data=potec[learn])
> tab<-table(pred.learn,potec$target[learn])
> (error.learn<-100*(1-sum(diag(tab))/nlearn))
> tab
   pred.learn <=50K >50K
     <=50K  15473  1659
     >50K     955  3620
The Training error is 12%

Test
> pred.test<-predict(model.tree, newdata=potec[-learn])#subset=-learn
> tab<-table(pred.test,potec$target[-learn])
> (error.test<-100*(1-sum(diag(tab))/ntest))
> tab
   pred.test <=50K >50K
     <=50K  7513  1070
     >50K    779  1492
The Test error is 17%. So there is an over-fitting of the model, the training error is a little bit optimistic.
Then we do the same process with CART tree. Here we are testing 19 parameters.

```
> rpart.fitted
CART

21707 samples
14 predictors
2 classes: ' <=50K', ' >50K'

No pre-processing
Resampling: Cross-Validated (10 fold)

Summary of sample sizes: 19536, 19536, 19536, 19536, 19536, 19536, ...

Resampling results across tuning parameters:

<table>
<thead>
<tr>
<th>cp</th>
<th>Accuracy</th>
<th>Kappa</th>
<th>Accuracy SD</th>
<th>Kappa SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.001</td>
<td>0.853</td>
<td>0.576</td>
<td>0.0104</td>
<td>0.0286</td>
</tr>
<tr>
<td>0.0015</td>
<td>0.85</td>
<td>0.566</td>
<td>0.011</td>
<td>0.0264</td>
</tr>
<tr>
<td>0.002</td>
<td>0.845</td>
<td>0.56</td>
<td>0.0121</td>
<td>0.0261</td>
</tr>
<tr>
<td>0.0025</td>
<td>0.844</td>
<td>0.558</td>
<td>0.0127</td>
<td>0.027</td>
</tr>
<tr>
<td>0.003</td>
<td>0.842</td>
<td>0.552</td>
<td>0.011</td>
<td>0.0218</td>
</tr>
<tr>
<td>0.0035</td>
<td>0.841</td>
<td>0.551</td>
<td>0.0115</td>
<td>0.0256</td>
</tr>
<tr>
<td>0.004</td>
<td>0.841</td>
<td>0.551</td>
<td>0.0115</td>
<td>0.0256</td>
</tr>
<tr>
<td>0.0045</td>
<td>0.84</td>
<td>0.545</td>
<td>0.00995</td>
<td>0.0218</td>
</tr>
<tr>
<td>0.005</td>
<td>0.839</td>
<td>0.542</td>
<td>0.0101</td>
<td>0.0238</td>
</tr>
<tr>
<td>0.0055</td>
<td>0.838</td>
<td>0.54</td>
<td>0.00917</td>
<td>0.0219</td>
</tr>
<tr>
<td>0.006</td>
<td>0.838</td>
<td>0.538</td>
<td>0.00887</td>
<td>0.0207</td>
</tr>
<tr>
<td>0.0065</td>
<td>0.836</td>
<td>0.535</td>
<td>0.0102</td>
<td>0.0205</td>
</tr>
<tr>
<td>0.007</td>
<td>0.832</td>
<td>0.522</td>
<td>0.0108</td>
<td>0.0214</td>
</tr>
<tr>
<td>0.0075</td>
<td>0.83</td>
<td>0.521</td>
<td>0.00965</td>
<td>0.0221</td>
</tr>
<tr>
<td>0.008</td>
<td>0.829</td>
<td>0.52</td>
<td>0.00972</td>
<td>0.0218</td>
</tr>
<tr>
<td>0.0085</td>
<td>0.83</td>
<td>0.521</td>
<td>0.00817</td>
<td>0.0199</td>
</tr>
<tr>
<td>0.009</td>
<td>0.83</td>
<td>0.519</td>
<td>0.00797</td>
<td>0.0181</td>
</tr>
<tr>
<td>0.0095</td>
<td>0.83</td>
<td>0.519</td>
<td>0.00797</td>
<td>0.0181</td>
</tr>
<tr>
<td>0.01</td>
<td>0.83</td>
<td>0.517</td>
<td>0.00785</td>
<td>0.018</td>
</tr>
</tbody>
</table>
```

Accuracy was used to select the optimal model using the largest value. The final value used for the model was `cp = 0.001`.

As the parameters plays an opposite role (the confidence threshold as opposed to the complexity parameter), in this case the accuracy curve has a decreasing behavior. The best parameter is 0.001.

```
> rpart.fitted$bestTune
1 0.001
```

With this parameter and the whole training set, we build the final model, and compute error rates.

```
> p1 <- rpart(target ~ ., data=learndata, control=rpart.control(cp=0.001), weights=w[learn])
```

```
Train
> pred.learn<-predict(p1, data=learndata, type="class")
> tab<-table(pred.learn,learndata$target)
> (error.learn<-100*(1-sum(diag(tab))/nlearn))
[1] 14.65426
```
The training error is 14.7%, which is slightly worst than with the C4.5 model.

Test

```r
> pred.test <- predict(p1, newdata=potec[-learn,], type="class") #subset=-learn
> tab.test <- table(pred.test, potec$target[-learn])
> (error.test <- 100*(1-sum(diag(tab.test))/n.test))
[1] 15.93882
```

The testing error is 16%. This time, this is less than the previous model. We computed the validation error on both model because as we are using only two models this isn’t costly. Nonetheless we should select our best final model according to the training error obtained. Therefore we would have chosen the C4.5 tree (but in this case we know that because of over-fitting this model is actually worst).

Here follow a plot of the tree obtained with CART algorithm (as we had issues plotting the C4.5 tree from RWeka library) and a complexity parameter non-optimal (cp=0.05) because otherwise the tree was too loaded. This plot is just here to ease the interpretation and give a visual.

One of the main advantages of prediction tree is their interpretability. Indeed here we can understand very quickly what is happening to make a decision. The cuts are done on ‘relationship’, ‘occupation’, ‘age’, ‘capital.gain’ and ‘workclass’. The dark grey represents the modality '>50K' while the light grey represents ‘<=50K’. For instance, following branches we can see that someone who is married and highly educated will have a high probability to make more than 50K a year (leaf 19, bottom right). On the other hand, someone unmarried or not in family will be very likely to make less than 50K a year (leaf 2, bottom left).

Finally, we also have to mention that we had issues to modify the weights with C4.5 method, while it was possible with the CART algorithm. In our studies of the impact the outliers can have (they represent only a 0.12% of the individuals which is quite negligible); we tried to
apply the algorithm building the model without the outliers (just suppressing the corresponding rows) and to test them whether on the testing data also suppressing outliers or on the whole remaining data (test data and outliers). In both cases we got a slightly better training error (which makes sense as we get rid of difficult cases) and a slightly worst validation error. What was strange was that the validation error obtained with the outliers was smaller than the one without outliers. As the differences were quite small (less than 0.3%) and it impacted neither the best parameter optimization nor the comparison of the best model, we chose to keep our results as they were. Another solution would have been to set the corresponding value of ‘Gain’ (99999) to NA and to impute it using Mice method, transforming therefore outliers into more ‘normal’ individuals. The different results obtained are in the appendix.

VI - Conclusion
Going over all the task of this work we can say that they are complementary and necessary. First the pre-processing of the data gives us a first overview and understanding of the topic. Besides before going into further processing we need to clean the data and put it in the desired format. Then MCA will give us a deeper understanding of the data and most importantly, of how variables are related to each other, and to individuals. This step is very important and also very relevant to the clustering. Indeed the clustering allows us to group individuals but we have to link those groups to the modalities to understand who are the individuals in each group.

Then the prediction is a different task, once we have understood the data, we can try to build a model to predict the target. Though we can see that some of the global behaviors we noticed observing the data (with MCA or with clustering) are retrieved in the prediction trees. The final predicting models are not so bad, as we have validation errors around 17% for the selected model (and 14% for the CART tree). If we consider that we are predicting the year income of individuals with only few information (14 categorical variables), we couldn’t expect very low errors. Indeed, explaining income with those few variables has led us to have very stereotyped results. We can remark as a final conclusion that our results, on the interpretation point of view, have to be considered taking some step back and keeping in mind that they only describe tendencies.
### 1. Catted results

**CatDes**<sup>-</sup>catedes(pot.num.var=15)

<table>
<thead>
<tr>
<th>$'1'$</th>
<th>Cls/Mod</th>
<th>Mod/Clas</th>
<th>Global</th>
<th>p.value</th>
<th>v.test</th>
</tr>
</thead>
<tbody>
<tr>
<td>sex= Male</td>
<td>37.46772786</td>
<td>98.84873968</td>
<td>66.92086912</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>relationship= Husband</td>
<td>60.72917467</td>
<td>97.00932316</td>
<td>40.61779736</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>occupation= Craft-repair</td>
<td>53.52650060</td>
<td>26.56496944</td>
<td>12.58867971</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>marital.status= Married-civ-spouse</td>
<td>54.30691103</td>
<td>98.47439157</td>
<td>45.99367341</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>education_num=education_num [1,9]</td>
<td>40.46882627</td>
<td>72.27627224</td>
<td>45.31876717</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>education= HS-grad</td>
<td>42.28169746</td>
<td>53.76935056</td>
<td>32.25029301</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
</tbody>
</table>

| age=50 to 64 | 39.15473280 | 27.14614360 | 17.58641499 | 0.05349043 | 13.2848269 |
| workclass= Self-emp-not-inc | 44.36581708 | 13.64571092 | 7.80831483 | 0.00176460 | 21.697386 |
| hours.per.week=hours.per.week (45,99) | 33.43695854 | 29.90180409 | 21.92905230 | 0.14735028 | 17.367645 |
| race= White | 26.96664410 | 90.82213433 | 85.42731744 | 0.04758183 | 16.778363 |
| occupation= Farming-fishing | 48.99336578 | 5.89656785 | 3.52731820 | 0.19132504 | 16.264988 |
| education= 7th-8th | 58.10835913 | 4.31044921 | 1.93895686 | 0.19470354 | 16.233130 |
| occupation= Machine-op-inspect | 40.05994006 | 9.71061872 | 6.14849581 | 0.34508050 | 14.898512 |
| education_num=education_num (9,10) | 31.31257715 | 77.64267174 | 22.39181843 | 7.42771409 | 13.032811 |
| education= Some-college | 31.31257715 | 77.64267174 | 22.39181843 | 7.42771409 | 13.032811 |
| age=36 to 49 | 29.08965390 | 38.16442469 | 32.47443260 | 0.85877037 | 12.670762 |
| hours.per.week=hours.per.week 40 | 28.38930144 | 52.30657465 | 46.73392267 | 0.81957683 | 11.737927 |

<table>
<thead>
<tr>
<th>$'2'$</th>
<th>Cls/Mod</th>
<th>Mod/Clas</th>
<th>Global</th>
<th>p.value</th>
<th>v.test</th>
</tr>
</thead>
<tbody>
<tr>
<td>education_num=education_num (10,12)</td>
<td>100.000000</td>
<td>100.000000</td>
<td>7.52126780</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>education= Assoc-voc</td>
<td>56.41319641</td>
<td>56.41319641</td>
<td>4.24434140</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>education= Assoc-acdm</td>
<td>100.000000</td>
<td>43.56880359</td>
<td>3.27692644</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>occupation= Tech-support</td>
<td>21.44396667</td>
<td>8.12367865</td>
<td>2.8503553</td>
<td>0.16999972</td>
<td>13.666841</td>
</tr>
<tr>
<td>education= 10th</td>
<td>0.000000</td>
<td>0.000000</td>
<td>2.8653010</td>
<td>6.85263433</td>
<td>-11.945515</td>
</tr>
<tr>
<td>education= 4th</td>
<td>0.000000</td>
<td>0.000000</td>
<td>3.68681155</td>
<td>2.14653924</td>
<td>-11.476643</td>
</tr>
<tr>
<td>education= Masters</td>
<td>0.000000</td>
<td>0.000000</td>
<td>5.29160665</td>
<td>6.95888261</td>
<td>-11.464543</td>
</tr>
<tr>
<td>education= Bachelors</td>
<td>0.000000</td>
<td>0.000000</td>
<td>16.4450851</td>
<td>3.67558680</td>
<td>-20.385462</td>
</tr>
<tr>
<td>education_num=education_num (9,10)</td>
<td>0.000000</td>
<td>0.000000</td>
<td>22.39181843</td>
<td>1.38055534</td>
<td>-29.922099</td>
</tr>
<tr>
<td>education= S-nursing</td>
<td>0.000000</td>
<td>0.000000</td>
<td>22.39181843</td>
<td>1.38055534</td>
<td>-29.922099</td>
</tr>
<tr>
<td>education_num=education_num (12,16)</td>
<td>0.000000</td>
<td>0.000000</td>
<td>22.39181843</td>
<td>1.38055534</td>
<td>-29.922099</td>
</tr>
<tr>
<td>education= HS-grad</td>
<td>0.000000</td>
<td>0.000000</td>
<td>22.39181843</td>
<td>1.38055534</td>
<td>-29.922099</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$'3'$</th>
<th>Cls/Mod</th>
<th>Mod/Clas</th>
<th>Global</th>
<th>p.value</th>
<th>v.test</th>
</tr>
</thead>
<tbody>
<tr>
<td>occupation= Prof-specialty</td>
<td>75.21739130</td>
<td>39.33308071</td>
<td>12.71459722</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>education_num=education_num (12,16)</td>
<td>98.10338158</td>
<td>99.96210686</td>
<td>24.77703762</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>education= Prof-school</td>
<td>99.47916667</td>
<td>7.23759000</td>
<td>1.76898744</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>education= Masters</td>
<td>99.07138715</td>
<td>21.56119742</td>
<td>5.29160652</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>education= Bachelors</td>
<td>97.53014006</td>
<td>65.97195980</td>
<td>16.44605510</td>
<td>0.00000000</td>
<td>Inf</td>
</tr>
<tr>
<td>education_num=education_num (9,10)</td>
<td>48.64731923</td>
<td>24.94821119</td>
<td>12.48733147</td>
<td>2.47738285</td>
<td>-28.362870</td>
</tr>
<tr>
<td>education= Doctorate</td>
<td>99.51578346</td>
<td>5.19136036</td>
<td>1.26838556</td>
<td>5.18132625</td>
<td>-25.390746</td>
</tr>
<tr>
<td>hours.per.week=hours.per.week (45,99)</td>
<td>36.92398827</td>
<td>33.29544019</td>
<td>21.92502030</td>
<td>4.34876263</td>
<td>-23.721473</td>
</tr>
<tr>
<td>marital.status= Married-civ-spouse</td>
<td>29.787660256</td>
<td>56.34711017</td>
<td>45.99367341</td>
<td>5.678733e-100</td>
<td>21.224450</td>
</tr>
</tbody>
</table>
$$`4`'$$

<table>
<thead>
<tr>
<th>Cla/Mod</th>
<th>Mod/Cla</th>
<th>Global</th>
<th>p.value</th>
<th>v.test</th>
</tr>
</thead>
<tbody>
<tr>
<td>sex= Female</td>
<td>39.60635093</td>
<td>52.12609971</td>
<td>33.07945088</td>
<td>0.000000e+00</td>
</tr>
<tr>
<td>occupation= Other-service</td>
<td>55.38694992</td>
<td>22.29960899</td>
<td>10.11946808</td>
<td>0.000000e+00</td>
</tr>
<tr>
<td>marital.status= Never-married</td>
<td>40.84994852</td>
<td>53.32355816</td>
<td>32.80918891</td>
<td>0.000000e+00</td>
</tr>
</tbody>
</table>
2. Prediction errors with different processing of outliers

- Initial results (without taking care of outliers in C4.5 and using weights in CART)

**C4.5:**

<table>
<thead>
<tr>
<th></th>
<th>train</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>12.04%</td>
<td>17.04%</td>
</tr>
</tbody>
</table>

**CART:**

<table>
<thead>
<tr>
<th></th>
<th>train</th>
<th>test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>14.65%</td>
<td>15.94%</td>
</tr>
</tbody>
</table>

- Results when training without the outliers

**C4.5**

Training error without outliers: 11.75% error

Testing error without outliers: 17.35%

Testing error with outliers: 17.05%

**CART:**

Training error without outlier: 14.92%

Testing error without outlier: 16.22%

Testing error with outliers: 16.20%
### 3. R code

```r
library(FactoMineR)
library(cluster)
library(class)
library(gtools)
library(xtable)

# Read the data
set.seed(10062014)
potec <- read.table("Adult.txt", header=FALSE, sep="", na.strings="NA", dec=".")
names(potec) <- c('age', 'workclass', 'fnlwgt', 'education', 'education.num', 'marital.status', 'occupation', 'relationship', 'race', 'sex', 'capital.gain', 'capital.loss', 'hours.per.week', 'native.country', 'target')

# Pre-Processing

# NA Values
for (i in 1:15) { potec[potec[,i]==' ?',i]<-NA}
summary(potec)

# Outliers detection
computeDistances <- function(x,G,V)
{
  lx <- x
  lg <- G
  lv <- V
  s <- svd(lv)
  D <- diag(1/s$d)
  linv <- s$v %% D %% t(s$u)
  distances <- seq(0,by=0, length = nrow(lx))
  for(i in 1:nrow(lx))
  {
    xi_minus_g <- as.matrix(lx[i,] - lg)
    maha_dist <- (xi_minus_g %% linv) %% t(xi_minus_g)
    distances[i] <- sqrt(maha_dist)
  }
  distances
}

# This function is used to compute the robust mahalanobis distance
loop.mahalanobis <- function (Dataset) {
  Bool <- FALSE
  s<-svd(cov(Dataset))
  D<-diag(1/s$d)
  Cov_inv <- s$v%%D%%t(s$u)
  Dm <- rep(0, nrow(Dataset))
  means <- colMeans(Dataset)

  n <- length(Dm)
  h <- round(0.75*n)
```
Matrix <- Dataset

while (n>2 && Bool == FALSE)
{
    for (i in 1:n)
    {
        centralised <- as.matrix(Matrix[i,] - means)
        mahasq <- centralised %*% Cov_inv %*% t(centralised)
        Dm[i] <- sqrt(abs(mahasq))
    }
    Sorted_Dm <- sort.int(Dm, decreasing=TRUE, index.return=TRUE)
    New_index <- Sorted_Dm$ix[1:h]
    New_matrix <- Dataset[New_index,]

    s <- svd(cov(New_matrix))
    D <- diag(1/s$d)
    Cov_inv_new <- s$v %*% D %*% t(s$u)
    means_new <- colMeans(New_matrix)
    n <- h
    h <- round(0.75*n)

    if (Cov_inv_new == Cov_inv && means_new == means)
    {Bool <- TRUE}
    else {
        Cov_inv <- Cov_inv_new
        means <- means_new
        Matrix <- New_matrix
    }

    return(Dm)
}

x <- potec[,c(1,3,5,11,12,13)]   #get only numeric columns
G <- as.matrix(colMeans(x))
V <- cov(x)

initial.distances <- computeDistances(x,G,V)
DMahalanobis.robust <- loop.mahalanobis(x)

#plot with outlier detection
plot(initial.distances, DMahalanobis.robust)
h = qchisq(.975,df=5)
abline(h = h, lty = 2, col = "red")
abline(v = h, lty = 2, col = "red")
outliers <- which(DMahalanobis.robust > h)

# those are the outliers
outliers<-c(24511,24639,24674,24851,24984,25179,25373,25612,25634,25842,26084,26415,26443,26594,
26826,27078,27222,27359,27414,27636,27641,28055,28215,28265,28295,28319,28350,29636,29807,30245,30497,30914,31829,31973,32091,32239,32519)

# the weights are changed accordingly
w<-rep(1,dim(potec)[1])
w[outliers]<-0.00001
# Summary of outliers
summary(potec[outliers,c(1,3,5,11,12,13)])
# Summary of non outliers
summary(potec[-outliers,c(1,3,5,11,12,13)])

# Capital gains of outliers
median(potec[outliers,c(11)])
# Capital gains of nonoutliers
median(potec[-outliers,c(11)])

# Factorization
cont<-c(3,5,11,12,13)  # the continuous variables that will be split into quartiles
for (i in cont)
{
  potec[,i]<-quantcut(potec[,i])
}
for(i in cont)
{
  levels(potec[,i])<-paste(colnames(potec)[i],levels(potec[,i]))
}

# Age variable
potec[,1]<-cut(potec$age, breaks = c(0,25,35,49,64,90))
levels(potec[,1])<-

# Dealing with NA values
naval<-c(2,7,14)  # the variables containing NA values

# will set NA as a level: level_NA
for (i in naval)
{
  potec[,i]<-factor(potec[,i], levels = c("level_NA",levels(potec[,i])[-1]))
  potec[is.na(potec[,i]),i]<-"level_NA"
}
summary(potec)

# MCA

# MCA with weights (outliers)
res.mca <- MCA(potec, quali.sup=illus, row.w=w)

# Plots
# plot of everything
plot(res.mca,label=c("var","quali.sup","quanti.sup"))  # too loaded

# plot active variables
plot(res.mca,invisible=c("ind","quanti.sup","quali.sup"),autoLab="y",cex=0.7)  # plot active variables
plot(res.mca,invisible=c("ind"),cex=0.7, selectMod="contrib 20", unselect="grey70")  # 20 variables contributed most
plot(res.mca,invisible=c("ind"),autoLab="y",cex=0.7,selectMod="cos2 20",unselect="grey70")  # 20 Variables most correlated
plot(res.mca, invisible=c("ind","var"))  # illustrative variable (target modalities)

# plot of individuals  (not in the report)
plot(res.mca,invisible=c("var","quali.sup"),autoLab="y",cex=0.7)  # individuals

# Description of dimensions
dimdesc(res.mca)

####################### Eigenvalues
plot(res.mca$eig$eigenvalue, type="l") #plot of eigenvalues according to dimensioins
abline(v = 51, lty = 2, col = "grey70")

res.mca$eig[ res.mca$eig[1] > 1/14, ,]
# 51 dimensions ! keep eig > 1/(Number actives variables). .. # so nd=51

sum(res.mca$eig[1])/109  #109 is the total should be the total number of dimensions
res.mca$eig[ res.mca$eig[1] > 0.07142857, ,] # also 51 dimensions with this rule (mean)

########################################
# Clustering                  #
########################################
res.mca2 <- MCA(potec, ncp = 51, quali.sup = illus, row.w = w) # redo MCA with 51 dimensionns kept
Psi <- res.mca2$ind$coord[,1:51]  # Projection of individuals on 51 kept dimensions

# CLUSTERING OF LARGE DATA SETS

FIRST 2 KMEANS WITH K=12
n1 = 12  # arbitrary: can be changed
k1 <- kmeans(Psi, n1)
k2 <- kmeans(Psi, n1)
table(k2$cluster, k1$cluster)
clas <- (k2$cluster-1)*n1+k1$cluster
summary(clas)  # 144 clusters (cross table of k1 and k2)
freq <- table(clas)  # number of elts in each cluster
cdcclas <- aggregate(as.data.frame(Psi), list(clas), mean)[,2:52]  #52=nd+1, center of gravity of cells of the cross table

SECOND HIERARCHICAL CLUSTERING UPON THE CENTRYDS OF CROSSING THE 2 KMEANS PARTITIONS
d2 <- dist(cdcclas)
h2 <- hclust(d2, method = "ward", members = freq)  # Tree with ward criteria
plot(h2)
barplot(h2$height[(nrow(cdcclas)-50):(nrow(cdcclas)-1)])  # plot of the last 50 aggregations
nc = 5  # cut after 4th jump, will keep 5 clusters
c2 <- cutree(h2, 5)  # cut the tree accordingly
cdg <- aggregate((diag(freq/sum(freq)) %*% as.matrix(cdcclas)), list(c2), sum)[,2:52]  # final center of gravity of clusters

PLOT OF CLUSTERING
plot(Psi[,1], Psi[,2], type = "n", main = "Clustering of individuals in 5 classes")
text(Psi[,1], Psi[,2], col = c2, cex = 0.6)
abline(h = 0, v = 0, col = "gray")
legend("topright", c("c1", "c2", "c3", "c4", "c5"), pch = 20, col = c(1:5))

# to help visualising (not in the report) plot of the individuals colored according to the target.
plot(Psi[,1], Psi[,2], type = "n", main = "target distribution")
text(Psi[,1], Psi[,2], col = unclass(potec[,15]), cex = 0.6)
legend("topright", levels(pote[,15]), pch = 20, col = c(1:2)); abline(h = 0, v = 0, col = "gray")
### CONSOLIDATION

```r
k6 <- kmeans(Psi, centers=cdg)
k6$size  # size of the clusters
```

### plot of the consolidated clustering

```r
plot(Psi[,1],Psi[,2], type="n", main="Clustering of individuals in 5 classes")
text(Psi[,1],Psi[,2], col=unclass(k6$cluster), cex = 0.6)
abline(h=0, v=0, col="gray")
legend("topright", c("c1","c2","c3","c4","c5"), pch=20, col=c(1:5))
```

### Description of clusters

```r
potec.comp = cbind.data.frame(potec, k6$cluster)  # A dataset with the cluster assignment
potec.comp[,16] <- as.factor(potec.comp[,16])
potec <- potec.comp[, -c(15)]  # don't want to describe the target with our clustering (predicted variable)
CatDes <- catdes(pot, num.var=15)
CatDes$category
```

### Prediction

```r
library(party)
library(RWeka)
library(partykit)
library(caret)
library(e1071)
library(rpart)
```

```r
### Split data into Training/Testing set
N <- dim(potec)[1]
learn <- sample(1:N, round(2*N/3))
nlearn <- length(learn)
test <- N - nlearn
```

```r
### Parameter optimization C4.5

c_sample <- seq(0.05, 0.50, by=0.05)
#length(c_sample)

# create fixed sampling scheme (10-folds)
train <- createFolds(potec$target, k=10)
```

```r
### (Prediction Tree) the fitting of parameters will be done on train set, using 10 fold CV
C45Fit <- train(potec[learn,-15], potec[learn,15], "J48",
               tuneLength = 10,
               tuneGrid=expand.grid(.C=c_sample),
               trControl = trainControl(
               method = "cv", indexOut = train, repeats=10))

plot(C45Fit$results[,1], C45Fit$results[,2], type='l')  # accuracy according to tested parameters
C45Fit  # accuracy keep increasing with C, so final model kept: 0.5
C45Fit$results  # table accuracy/parameters
C45Fit$bestTune  # best parameter
C45Fit$finalModel  # can see whole description of tree (it is quite long)
```

### Build the model C4.5
at this point we use the best parameter (c=0.5) to make a prediction with all training data (no more CV)
model.tree<-J48(target~., data=potec, subset=learn, control=Weka_control(C=0.5),
na.action=NULL)

Errors C4.5
Training sample
pred.learn<-predict(model.tree, data=potec[learn])
tab<-table(pred.learn,potec$target[learn]) # contingency matrix
(error.learn<-100*(1-sum(diag(tab))/nlearn)) #12% => This model is selected

Test sample
pred.test<-predict(model.tree, newdata=potec[-learn],) #subset=learn
tab<-table(pred.test,potec$target[-learn]) # contingency matrix
(error.test<-100*(1-sum(diag(tab))/n.test)) #17%

Parameter optimization with CART
learnndata <- potec[learn,]
cp_sample <- seq(0.001,0.01,by=0.0005)
train <- createFolds(potec$target, k=10)
rpart.fitted <- train(potec[learn,-15], potec[learn,15], "rpart", weights=w[learn],
tuneLength = 19,
tuneGrid=expand.grid(.cp=cp_sample),
trControl = trainControl(method = "cv", indexOut = train,
repeats=10))
rpart.fitted$results # parameters/accuracy table
rpart.fitted$bestTune #best parameter cp = 0.001
rpart.fitted$finalModel # can see whole description of tree

Build model CART
p1 <- rpart(target ~ ., data=learndata,control=rpart.control(cp=0.001),
weights=w[learn])

Plot a tree CART (no same complexity parameter because the tree is too big)
#p1 <- rpart(target ~ ., data=learndata,control=rpart.control(cp=0.05),
weights=w[learn])
plot(rpart.fitted$results[,1],rpart.fitted$results[,2],type='l')
plot(as.party.rpart(p1),type="extended")

Errors CART
Training sample
pred.learn<-predict(p1, data=learndata, type="class")
tab<-table(pred.learn,learndata$target)
(error.learn<-100*(1-sum(diag(tab))/nlearn)) #15% => This model isn't selected

Test sample
pred.test<-predict(p1, newdata=potec[-learn,], type="class")
tab.test<-table(pred.test,potec$target[-learn])
(error.test<-100*(1-sum(diag(tab.test))/n.test)) #16%