Tree Structure and Mobility Tree

Gilbert Ritschard

Department of Econometrics and Laboratory of Demography, University of Geneva
http://mephisto.unige.ch

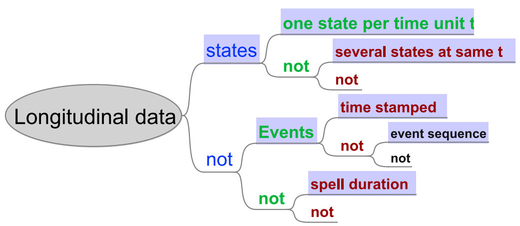
Workshop on Sequence Analysis, Lund, May 8-9, 2008

UNIVERSITÉ DE GENÈVE
FACULTÉ DES SCIENCES ÉCONOMIQUES ET SOCIALES
Département d'économétrie

Plan

Organizing knowledge in tree form

- Giving a hierarchical presentation of knowledge of a domain in tree form facilitates understanding.
- An Aristotelean tree, splits concepts according to simple yes-no questions (analytical tree).
- Example: What kind of longitudinal data do you have?



Trees induced from data

- The previous tree is a logical analysis of possible situations.
- Here, we are interested in tree structure induced from data.
- Empirical trees derived from data.
- Aim is to partition data into groups:
  - that are as homogeneous as possible (minimal within class diversity)
  - that differ as much as possible from each other (maximal between class diversity)

Supervised and unsupervised trees

- Supervised tree: There is a (univariate or multivariate) target variable and branching is defined in terms of the values of covariates.
- Diversity is measured in the space of this target variable.
- Examples: Classification tree, regression tree, survival tree, ...

- Unsupervised tree: There is no specific target variable and no branching condition in terms of values of variables.
- Diversity is measured in the space of all considered variables.
- Example: Dendrogram representing hierarchical clustering.

Trees for sequence data

- We shall focus on supervised trees and their use for sequence data.
- How is present state related to previous states? (Mobility analysis)
- How discriminating are specific sequencing patterns, for sex, cohort, ...?
- How are typical sequencing patterns linked to covariates of interest?

- However, Will also shortly discuss typology of sequences (Hierarchical clustering)

Social mobility over 3 generations

- Using data from acts of marriage of 19th century Geneva Ryczkowska (2003)
- On each act:
  - profession of the groom
  - profession of the father (at son's marriage)
- By matching records of the groom with that of his father
  - profession of the father (at father's marriage)
  - profession of the grand-father (at father's marriage)
- 572 matched records (i.e. grooms whose father married also in Geneva)

The social statuses

6 statuses derived from the professions

- unskilled: unskilled daily workmen, servants, labourer, ...
- craftsmen: skilled workmen
- clock makers: skilled persons working for the "Fabrique"
- white collars: teachers, clerks, secretaries, apprentices, ...
- petite et moyenne bourgeoisie: artists, coffee-house keepers, writers, students, merchants, dealers, ...
- élites: stockholders, landlords, householders, businessmen, bankers, army high-ranking officers, ...

+ unknown

Social statuses in 3 categories

For further simplification we consider also Statuses into 3 categories

3 class statuses	6 class statuses
Low	unknown
	unskilled
	craftsmen
Clock	clock makers
High	white collars
	PMB
	elites

Father-Son Social Transition, Enrooted

Father to son social transition rates, Geneva 1830-1880, enrooted population (572 cases)

Father	Son unkwn	unskil	craft	clock	wcolar	PMB	élite	Total
unknown	.	.	**22.2**	33.3	**22.2**	**22.2**	.	100
unskilled	.	**27.3**	9.1	**36.4**	.	**27.3**	.	100
craftsman	.	1.2	**39.5**	29.6	11.1	14.8	3.7	100
clock maker	.	**7.2**	4.8	**63.9**	8.4	13.3	2.4	100
white collar	.	.	**27.8**	22.2	**16.7**	**27.8**	5.6	100
PMB	.	**7.4**	10.3	20.6	2.9	**47.1**	**11.8**	100
élite	2.2	2.2	8.9	8.9	4.4	**31.1**	**42.2**	100
deceased	.	**6.6**	12.8	**35.0**	**17.5**	18.3	9.7	100
Total	0.2	5.8	15.4	34.3	12.2	22.0	10.1	100

Three generations social transitions

- Classical approach: Markov model up to order 3:
  - Status at t depends on statuses at $t-1$, $t-2$ and $t-3$:

    $$p(s_t \mid s_{t-1}, s_{t-2}, s_{t-3})$$

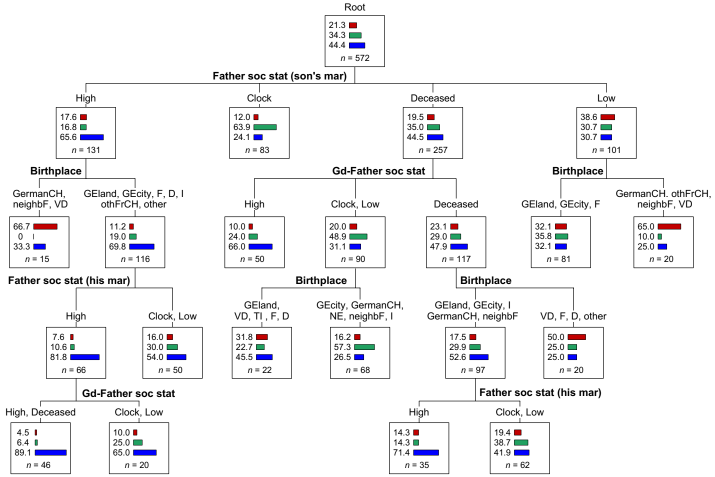
    holds for any status s_t.
  - For 3 statuses, there are $3^3 = 27$ different conditions.
  - Many free parameters (54) $\Rightarrow$ modeling probabilities in term of fewer parameters (Berchtold and Raftery, 2002)
  - Can be done with Berchtold and Berchtold (2004)'s March software.
- Mobility tree: Flexible Markov model
- Each s_t depends only on significant previous statuses.
- Classification tree for which the present status s_t is the target, and previous statuses $s_{t-1}, s_{t-2}, s_{t-3}$ are predictors.
- Easy to account for other covariates.

Covariate: Geographical Origin

Code	Place
GEcity	Geneva city
GEland	Geneva surrounding land
neighbF	neighboring France
VD	Vaud
NE	Neuchatel
otherFrCH	other French speaking Switzerland
GermanCH	German speaking Switzerland
TI	Italian speaking Switzerland
F	France
D	Germany
I	Italy
other	other

Mobility tree for the 3 generations problem

Son's Status: Low (workers and craftsmen), Clock Maker, High



Tree quality

- Error rate: 42.4%, i.e. 24% reduction of the classification error rate of the initial node
- Goodness of fit

Tree	$G2$	df	sig	BIC	AIC	pseudo R^2
Indep	482.3	324	0.000	2319.6	812.3	0
Level 1	408.2	318	0.000	1493.9	750.2	0.14
Level 2	356.0	310	0.037	$\boxed{1492.5}$	714.0	0.23
Level 3	327.6	304	0.168	1502.2	697.6	0.28
Fitted	312.5	300	0.298	1512.5	$\boxed{690.5}$	0.30
Saturated	0	0	1	3104.7	978.0	1

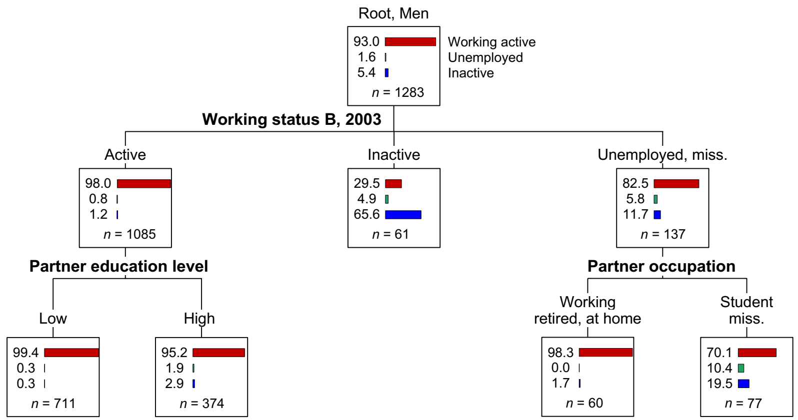
Mobility over working statuses

- (SHP Data, Waves 1 to 6 (1999-2004), aged between 20 and 64 in 2004.)
- How does working status (occupied active, unemployed, inactive) in 2004 depend on
  - working status in previous year (1999 to 2003)
  - other factors (attained education level, partner working status, partner education level, ...)

  and what are main interaction effects?

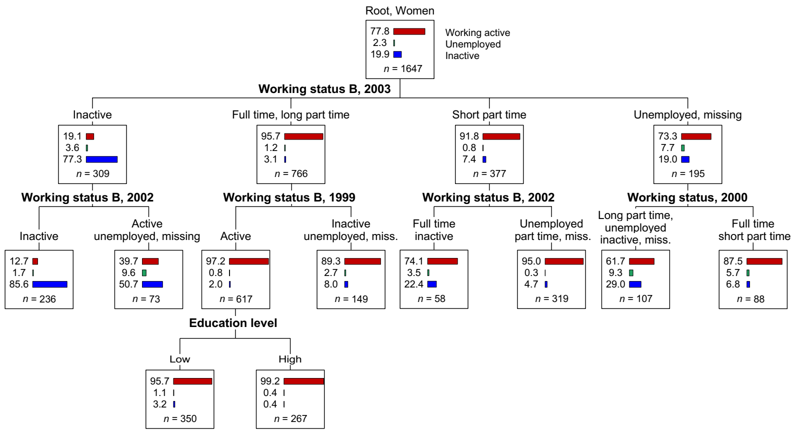
- Mobility trees are alternative to Markovian transition models.
- Growing separate classification trees for women and men highlights gender differences.

Mobility tree, Men



Working status B (full time, long part time, short part time, unemployed, inactive)

Mobility tree, Women



Working status B (full time, long part time, short part time, unemployed, inactive)

Induction Trees: Introduction (1)

- Trees induced from data.
- Recursive partitioning, segmentation, ...
- Most often used for classification: classification tree, when target is a categorical variable.
- Regression tree, when response variable is measurable at interval or ratio scale.
- Objective: Partition data according to explanatory factors (attributes, predictors, covariates) so that the distribution of the response variable (dependent variable to be predicted):
  - is the purest possible in each class (maximize class homogeneity = minimize within class differences)
  - differs as much as possible from one class to the other (maximize between class differences);

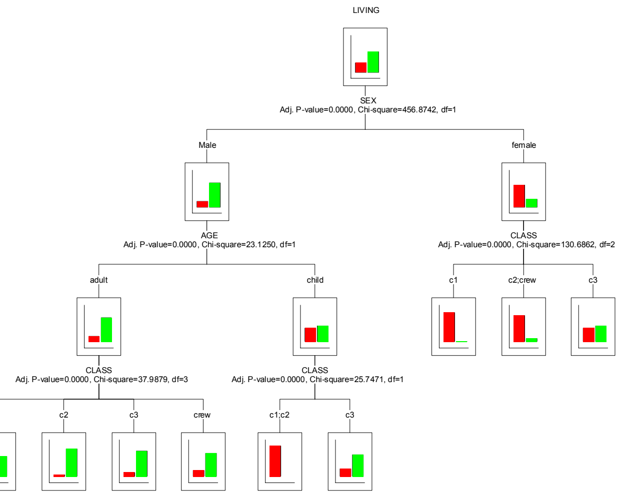
Induction Trees: Introduction (2)

Singles out interactions of covariates in their effect on the response variable

Results:

- visual (a tree);
- no coefficients measuring the effect of covariates;
- classification rules.

Illustration: Titanic



Supervised learning

- Based on a learning sample $\{(\mathbf{x}_\alpha, y_\alpha)\}_{\alpha=1,\dots,n}$,
  - where y_α is the value (class) of the response (dependent, ...) variable for case α,
  - and $\mathbf{x}_\alpha = (x_{\alpha 1}, \dots, x_{\alpha p})$ is the profile of α in terms of the covariates.
- Build a predictive function (or classification function)

$$y = f(\mathbf{x})$$

  with which we can predict the value or class y when only the profile $\mathbf{x}$ is known.
- Example: predict whether a passenger of the Titanic survives from the sole knowledge of sex, age (child/adult) and navigation class.

Target Table

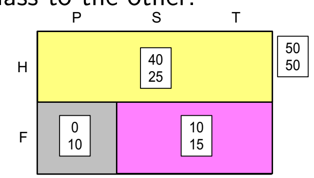
- Assuming all variables are categorical, we can represent the data with a contingency table that cross tabulates the response variable with a composite variable defined by the cross tabulation of all covariates.
- Example of a target contingency table **T**.
- Response variable is marital status, predictors are sex and sector of activity

	man			woman			
married	primary	secondary	tertiary	primary	secondary	tertiary	total
no	11	14	15	0	5	5	50
yes	8	8	9	10	7	8	50
total	19	22	24	10	12	13	100

Constructing the rules

An induction tree (like a logistic regression) determines the rule $f(x)$ in two steps

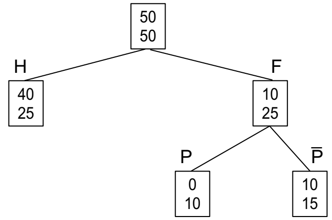
1. Determine a partition of the possible profiles $\mathbf{x}$ such that the distribution p_y of the response Y is as different as possible from one class to the other.



2. The rule consists then in assigning to each case the most frequently observed value y in the class defined by the values of x.

$$\hat{y} = f(\mathbf{x}) = \arg\max_i \hat{p}_i(\mathbf{x})$$

Induced Tree



- Partitions are determined by successive splits of nodes.
- Starting with the root node (formed by the set of all cases), we seek the covariate that permits the better split according to a given criterion (greatest entropy reduction, strongest association with the response.)
- Operation is repeated at each new obtained node until fulfilment of some stopping criterion (a minimal node size or a minimal gain in the criterion).

Splitting criteria

Criteria from

- Information Theory: Entropies (uncertainty) prediction made from the resulting distribution

  Shannon's entropy: $h_S(p) = -\sum_{i=1}^{c} p_i \log_2 p_i$

  Quadratic entropy (Gini): $h_Q(p) = \sum_{i=1}^{c} p_i(1-p_i) = 1 - \sum_{i=1}^{c} p_i^2$

  $\Rightarrow$ maximizing entropy reduction
  (maximizing within leaves homogeneity)

- Statistical associations: Pearson's Chi-square, measures of association.
  $\Rightarrow$ maximizing association,
  minimizing the p-value of the no-association test.
  (maximizing diversity between leaves)

Gain of information (1)

- Splitting the root node by sex, we get two nodes.
- The distribution in each node is that of the corresponding column of Table below

Marital status by sex

age	man	woman	total
married	40	10	50
not married	25	25	50
total	65	35	100

- What information brings "sex"?

Gain of information (2)

- Gain = reduction of uncertainty
- Uncertainty: Shannon's entropy

$$H(\text{marital status}) = -\sum_{i=1}^{c} p_i \log_2 p_i$$
$$= -\left(\frac{50}{100}\log_2\left(\frac{50}{100}\right) + \frac{50}{100}\log_2\left(\frac{50}{100}\right)\right) = \boxed{1}$$
$$H(\text{marital status}|\text{man}) = -\left(\frac{40}{65}\log_2\left(\frac{40}{65}\right) + \frac{25}{65}\log_2\left(\frac{25}{65}\right)\right) = \boxed{.961}$$
$$H(\text{marital status}|\text{woman}) = -\left(\frac{10}{35}\log_2\left(\frac{10}{35}\right) + \frac{25}{35}\log_2\left(\frac{25}{35}\right)\right) = \boxed{.863}$$
$$H(\text{marital status}|\text{sex}) = (65/100)0.961 + (35/100)0.863 = \boxed{0.927}$$
$$\text{Gain(sex)} = H(\text{marital status}) - H(\text{marital status}|\text{sex})$$
$$= 1 - 0.927 = \boxed{0.073}$$

Most popular tree growing methods

- CHAID, CHi-square based Automatic Interaction Detection (Kass, 1980; Biggs et al., 1991): n-ary trees, criterion based on Bonferroni adjusted p-values of independence tests.
  - CHAID is an extension of an earlier regression tree method called AID (Morgan and Sonquist, 1963)
- CART, Classification and Regression Tree (Breiman et al., 1984): binary trees, criterion is maximizing decrease of Gini purity measure, pruning, surrogate splits in case of missing values.
- C4.5 (Quinlan, 1993): binary trees, criterion is Information Gain, the reduction in Shannon's entropy standardized by the entropy of the predictor.
- C4.5 was designed in a less statistical and more IA perspective.

Most popular tree growing methods (2)

- CART and C4.5 were designed for prediction purposes (prediction error is a primary concern).
- CHAID and AID primary aim is interaction detection. Their aim is primary description, rather than prediction.

Party

- There at least two packages in R for growing (binary) trees:
  - `rpart` (Therneau and Atkinson, 1997): recursive partitioning CART, Relative risk trees,
  - `party` (Hothorn et al., 2006): conditional partitioning Based on a statistical conditional inference method (permutation tests)
- We discuss here only the second one
  - much more powerful and flexible.
  - better visual rendering (Plots distributions inside the nodes)

`party` principle

- `party` selects each split in two steps (to avoid bias in favor of predictors with many different values):
  - First, selects the predictor with strongest association with target,
  - Then, selects the best binary split for selected predictor.

Linear statistic and permutation test

- Both steps are based on the conditional distribution of linear statistics in a permutation test framework.
  - Linear statistic is:

$$\mathbf{T}_j = \text{vec}\left(\sum_{i=1}^{n} w_i g_j(X_{ji}) h(\mathbf{Y}_i, (\mathbf{Y}_1, \ldots, \mathbf{Y}_n))^T\right) \in \mathbb{R}^{p_j q}$$

    where $g_j(X_{ji})$ is a transformation of X_{ji}, and $h()$ an influence function.
  - $\mathbf{T}_j$ is computed for each permutation of the $\mathbf{Y}$ values among cases, and results characterize its conditional independence distribution.
  - the variable and split selection is then based on the p-value of the observed $\mathbf{t}$ under this conditional independence distribution.

Creating or reading a data set in R

- You can either create a data.frame within R

```
# creating data set in R
marr <- rbind(
  data.frame(married="yes",sex="man",  activity="primary",  weight=11),
  data.frame(married="yes",sex="man",  activity="secondary",weight=14),
  data.frame(married="yes",sex="man",  activity="tertiary", weight=15),
  data.frame(married="yes",sex="woman",activity="primary",  weight=0),
  data.frame(married="yes",sex="woman",activity="secondary",weight=5),
  data.frame(married="yes",sex="woman",activity="tertiary", weight=5),
  data.frame(married="no", sex="man",  activity="primary",  weight=8),
  data.frame(married="no", sex="man",  activity="secondary",weight=8),
  data.frame(married="no", sex="man",  activity="tertiary", weight=9),
  data.frame(married="no", sex="woman",activity="primary",  weight=10),
  data.frame(married="no", sex="woman",activity="secondary",weight=7),
  data.frame(married="no", sex="woman",activity="tertiary", weight=8) )
marr # displays content of marr
```

- It is however more convenient to read a file, for instance a csv file

```
marr <- read.csv(file="C:/data/lund/exple_married_sex_sector.csv",header=TRUE)
```


A R script for generating a tree

- You grow the tree with the ctree command

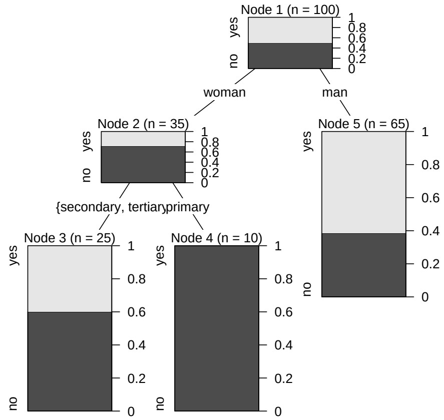
```
#loading party
library(party)

marrtree <- ctree(married ~ ., data=marr[,1:3],
    controls=ctree_control(mincriterion=.50,minsplit=0),
    weights=marr$weight)
marrtree # dispays info on tree

plot(marrtree)  # plots the tree

# Plotting same tree using some controls.
plot(marrtree,drop_terminal=F,inner_panel=node_barplot)
```


Output in R console

```
> marrtree

        Conditional inference tree with 4 terminal nodes

Response:  married
Inputs:  sex, activity
Number of observations:  12

1) sex == {woman}; criterion = 0.996, statistic = 9.791
  2) activity == {secondary, tertiary}; criterion = 0.874, statistic = 5.471
    3)*  weights = 25
  2) activity == {primary}
    4)*  weights = 10
1) sex == {man}
  5)*  weights = 65
```


Here is the first plotted tree

Response variable is: "married"



Mobility tree on the 3 generations mobility data

```
## Mobility tree example with data from marriage acts of 19th Century Geneva

library(foreign) # library for importing data from various sources
sm_data <- read.spss(file="C:/data/lund/mobility/par_enf_tree_267.sav",to.data.frame=T)
sm_data$NC1_ST3 <- factor(sm_data$NC1_ST3) # to remove deceased category

# ordering and renaming state variables
seqs <- data.frame(GdFather=sm_data$NG1ST_P3, Father_his_M = sm_data$NP1_ST3,
       Father_son_M = sm_data$NC1ST_P3, Son_M=sm_data$NC1_ST3)

# Growing mob tree with ctree (party package)

library(party)

cl_tree <- ctree(seqs$Son_M ~ seqs$Father_son_M + seqs$Father_his_M + seqs$GdFather +
       sm_data$C1LIEU11)
plot(cl_tree)

# you may control the tree with ctree_control()

control <- ctree_control(testtype="Univariate",mincriterion=.9,minsplit=20,minbucket=10)
cl_tree <- ctree(seqs$Son_M ~ seqs$Father_son_M + seqs$Father_his_M + seqs$GdFather +
   sm_data$C1LIEU11,controls=control)
plot(cl_tree,drop_terminal=F)
```


State variables

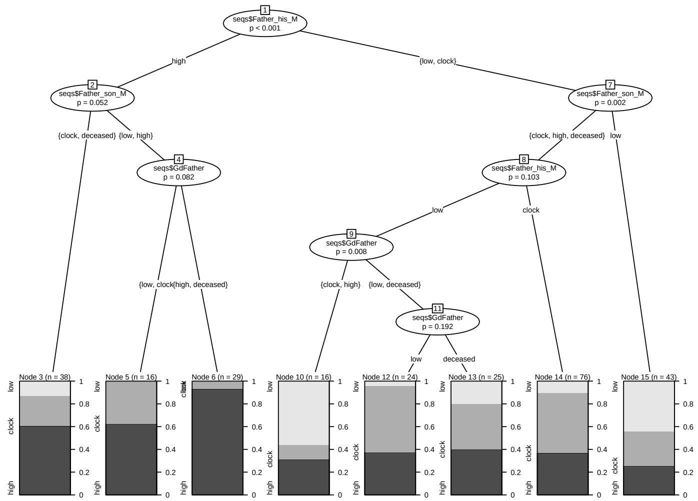
- Variables are

variable	label
GdFather	'Status Grd-father, 3 categories'
Father_his_M	'Status Father (his marr.), 3 categories'
Father_son_M	'Status Father (son"s marr.), 3 categories'
Son_M	'Status Son (his marr.), 3 categories'

Text output

```
         Conditional inference tree with 8 terminal nodes

Response:  seqs$Son_M
Inputs:  seqs$Father_son_M, seqs$Father_his_M, seqs$GdFather, sm_data$C1LIEU11
Number of observations:  267

1) seqs$Father_his_M == {high}; criterion = 1, statistic = 48.744
  2) seqs$Father_son_M == {clock, deceased}; criterion = 0.948, statistic = 12.494
    3)*  weights = 38
  2) seqs$Father_son_M == {low, high}
    4) seqs$GdFather == {low, clock}; criterion = 0.918, statistic = 6.709
      5)*  weights = 16
    4) seqs$GdFather == {high, deceased}
      6)*  weights = 29
1) seqs$Father_his_M == {low, clock}
  7) seqs$Father_son_M == {clock, high, deceased}; criterion = 0.998, statistic = 20.864
    8) seqs$Father_his_M == {low}; criterion = 0.897, statistic = 13.387
      9) seqs$GdFather == {clock, high}; criterion = 0.992, statistic = 17.472
        10)*  weights = 16
      9) seqs$GdFather == {low, deceased}
        11) seqs$GdFather == {low}; criterion = 0.808, statistic = 8.461
          12)*  weights = 24
        11) seqs$GdFather == {deceased}
          13)*  weights = 25
    8) seqs$Father_his_M == {clock}
      14)*  weights = 76
  7) seqs$Father_son_M == {low}
    15)*  weights = 43
```


And here is the induced tree



Transition rates

- You may get transition rates with TraMineR

```
> library(TraMineR)
> seqtrate(seqs)
 Computing transition rates between states clock deceased high low, please wait
              [-> clock] [-> deceased] [-> high]  [-> low]
[clock ->]      0.5062500     0.1562500 0.2625000 0.0750000
[deceased ->]   0.3333333     0.0000000 0.3607306 0.3059361
[high ->]       0.1641791     0.1492537 0.5621891 0.1243781
[low ->]        0.1357466     0.2352941 0.1764706 0.4524887
>
```


Quality of the tree

- When the concern is classification, i.e. predicting the value of the response variable, we look typically at the error rate.
  - Error rate should be computed on a test sample (different from learning sample).
  - Cross-validation is often used.
- For non classification purposes (as is most often the case in social sciences)
  - We can compute some deviance (Ritschard, 2006) that measures how far the obtained partition is from the finest one that can be defined with the predictors.
  - Deviance reduction between nested models can be compared with Chi-square distributions (example).

Other issues with trees

- Structure instability. The structure of the tree (selected predictors and splits) may vary when data is slightly perturbed.
  - Could be tested with resampling methods (Dannegger, 2000).
- Multi-level analysis
  - How can we account for multi-level effects in classification trees?
  - Conjecture: Should be possible to include unobserved shared effect in deviance-based splitting criteria.
- This remains all to be done ....

Complexity

- Tree complexity:
  - number of nodes
  - number of levels
  - message length (rules)
- We may reduce complexity
  - a priori by reinforcing stopping rules
    (e.g. maximal number of levels or minimal node size)
  - a posteriori through pruning
    (mainly used with non statistical splitting criteria, such as in CART)
- In statistics, complexity of model = number of free parameters
  - Can also be applied to trees.

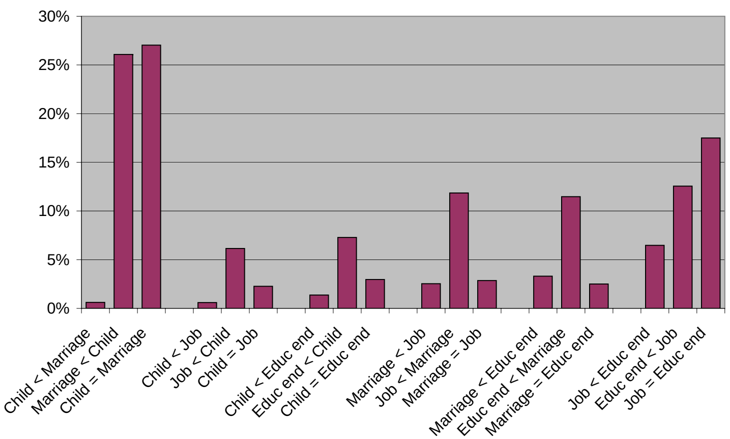
Quality of partitions

- Global improvement of criterion
  - Gain of information between root node and set of all leaves (terminal nodes).
  - Degree of association between target and final partition (GK τ, Cramer's v, ...).
  - p-value of independence test between target and partition (node numbers).
- With `party`, you may use the `where(growntree)` command to retrieve the node membership for each case.

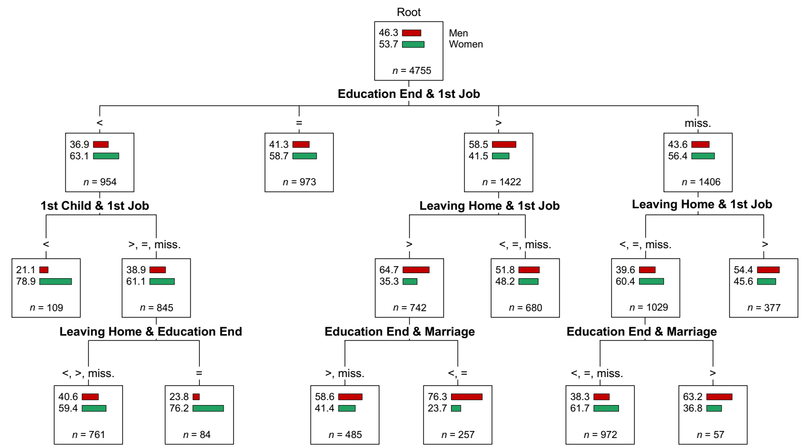
Discriminating with typical sequencing pattern

- Approach used by (Billari et al., 2006) on FFS data for Italy and Austria.
- Here we consider (SHP 2002 biographical data)
- Selection of pairs of events, e.g. marriage and first job.
- For each pair, order of sequence: $<$, $=$, $>$, missing
- Which are the most typical sequences?
- Most discriminating sequences between
  - sex
  - birth cohort (1940 and before, after 1940)

Frequencies of characteristic 2-event sequences



Discriminating sex with 2-event sequences



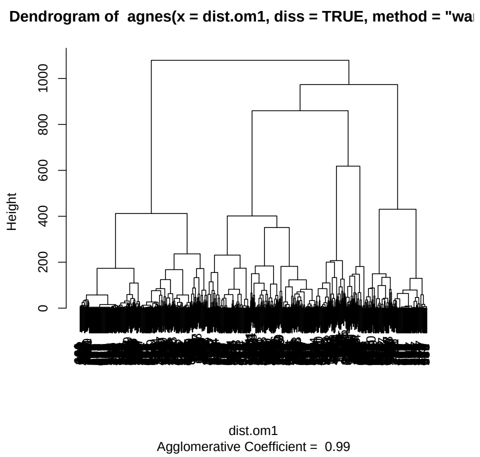
Unsupervised trees

- Compute a distance or proximity matrix between sequences (OM or other metrics).
- Can be done with CHESA (Elzinga, 2007) or with `TraMineR` in R.
- Once you have the distance matrix you can make a hierarchical clustering and produce the dendrogram.

R script: hierarchical clustering

```
library(TraMineR)
# reading mvad data from a csv file
mvad <- read.csv(file="c:/data/lund/McVicar.csv",header=TRUE)

svar <- 15:86    # sets the sequence to be considered

# Computing OM distances
# First we compute the substitution costs based on transition rates
submat <- seqsubm(mvad,svar,method= "TRATE")
# and now the OM distances
dist.om1 <- seqdist(mvad, svar, format="STS", method="OM", indel=1, submat)

# Hierarchical Clustering with Ward Method
library(cluster)
clusterward1 <- agnes(dist.om1, diss=TRUE, method="ward")
plot(clusterward1)
```


Resulting dendrogram



Bibliographie I

Berchtold, A. and A. Berchtold (2004). MARCH 2.02: Markovian model computation and analysis. User's guide, www.andreberchtold.com/march.html.

Berchtold, A. and A. E. Raftery (2002). The mixture transition distribution model for high-order Markov chains and non-gaussian time series. *Statistical Science 17*(3), 328–356.

Biggs, D., B. De Ville, and E. Suen (1991). A method of choosing multiway partitions for classification and decision trees. *Journal of Applied Statistics 18*(1), 49–62.

Billari, F. C., J. Fürnkranz, and A. Prskawetz (2006). Timing, sequencing, and quantum of life course events: A machine learning approach. *European Journal of Population 22*(1), 37–65.

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone (1984). *Classification And Regression Trees*. New York: Chapman and Hall.

Dannegger, F. (2000). Tree stability diagnostics and some remedies for instability. *Statistics In Medicine 19*(4), 475–491.

Bibliographie II

Elzinga, C. H. (2007). CHESA 2.1 User manual. User guide, Dept of Social Science Research methods, Vrije Universiteit, Amsterdam.

Hothorn, T., K. Hornik, and A. Zeileis (2006). party: A laboratory for recursive part(y)itioning. User's manual.

Kass, G. V. (1980). An exploratory technique for investigating large quantities of categorical data. *Applied Statistics 29*(2), 119–127.

Morgan, J. N. and J. A. Sonquist (1963). Problems in the analysis of survey data, and a proposal. *Journal of the American Statistical Association 58*, 415–434.

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. San Mateo: Morgan Kaufmann.

Ritschard, G. (2006). Computing and using the deviance with classification trees. In A. Rizzi and M. Vichi (Eds.), *COMPSTAT 2006 - Proceedings in Computational Statistics*, pp. 55–66. Berlin: Springer.

Ryczkowska, G. (2003). Accès au mariage et structure de l'alliance à Genève, 1800-1880. Mémoire de dea, Déartement d'histoire économique, Université de Genève, Genève.

Bibliographie III

Therneau, T. M. and E. J. Atkinson (1997). An introduction to recursive partitioning using the rpart routines. Technical Report Series 61, Mayo Clinic, Section of Statistics, Rochester, Minnesota.