Data Mining
(Mining Knowledge from Data)

Decision Trees

Marcel Jiřina, Pavel Kordík

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

FIT

Construct a classifier that determines whether to play tennis

Day	Outlook	Temp.	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

By means of 1NN algorithm

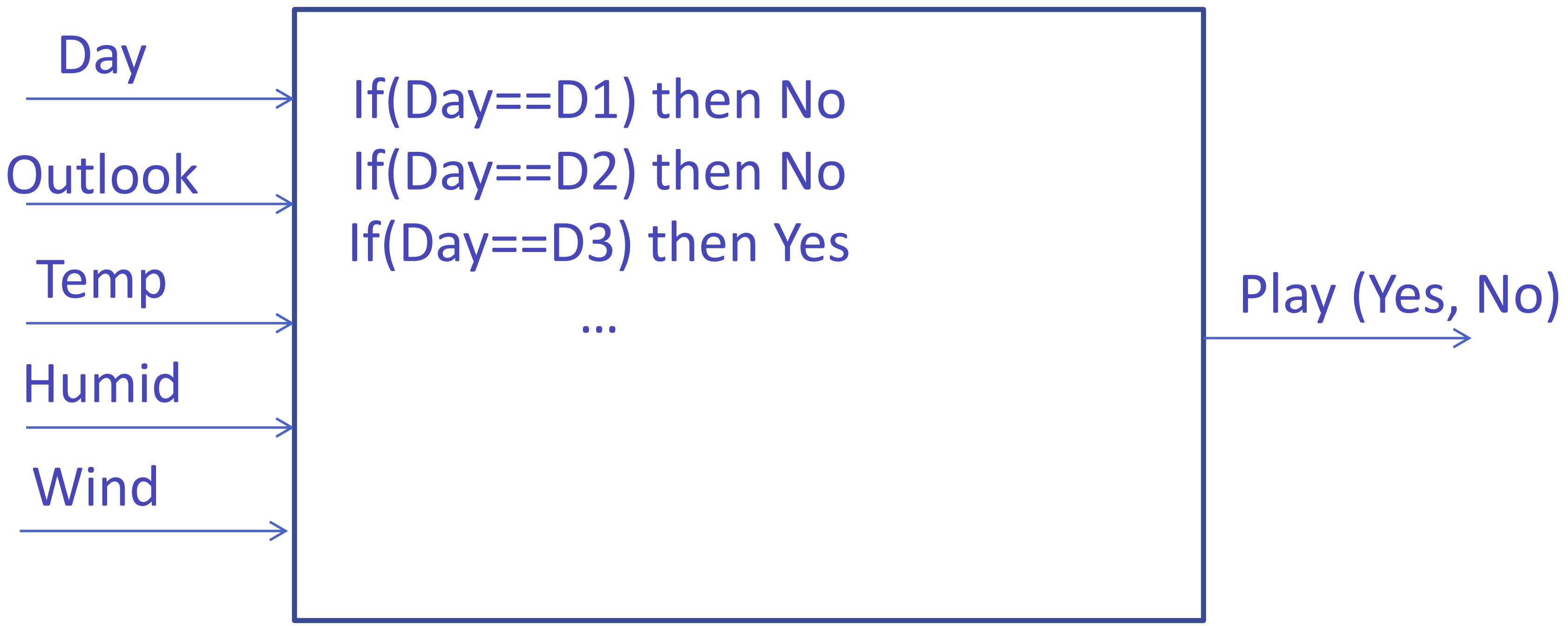
- How do I calculate distances of instances?
- Coding into the interval (0 = Cool, 0.5 = Mild, 1 = Hot)
  - Disadvantages?
  - I need an expert ...
- Coding 1 of N (new binary attribute Cool)
  - Disadvantages?
  - I need to encode all the classes - a large number of attributes
- The Hamming distance?
- The curse of dimensionality

Construct a classifier that determines whether to play tennis

- Another idea?

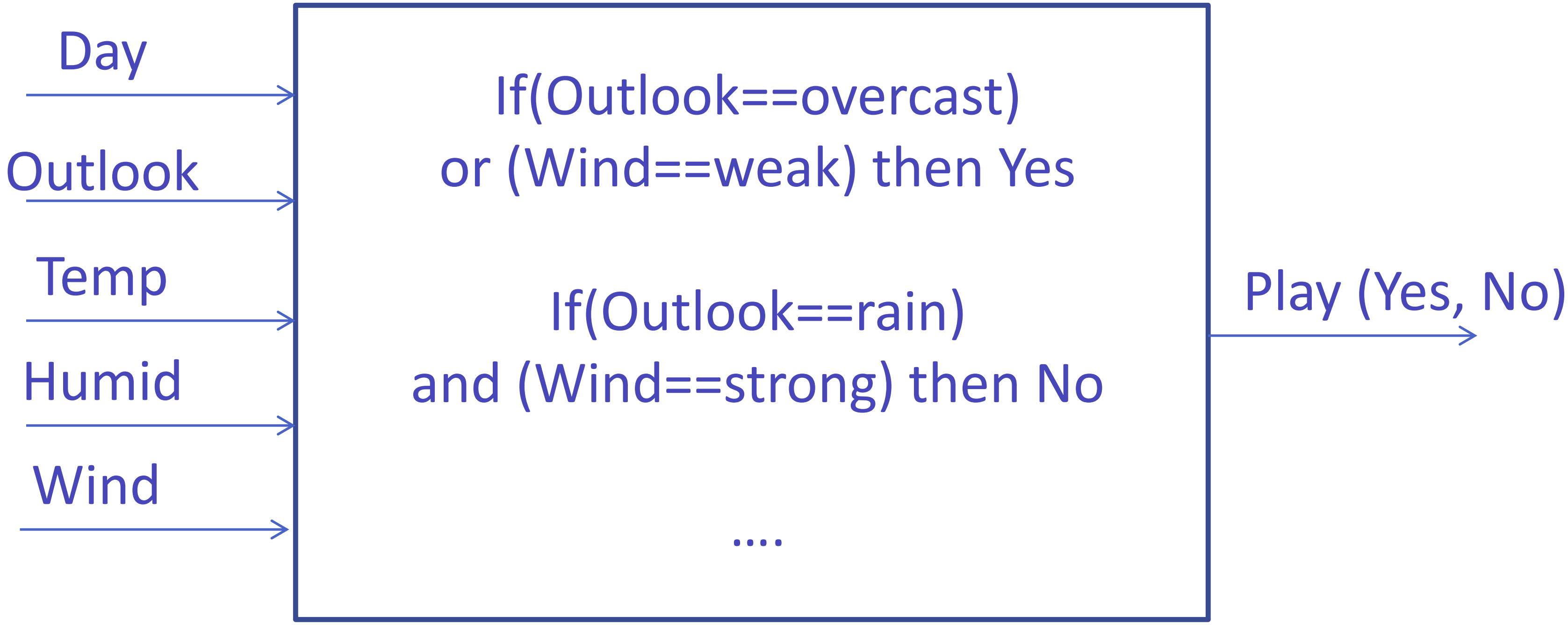
Day	Outlook	Temp.	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Design a classifier based on a set of rules



- Any problems?
- Another idea?

Design a classifier based on a set of rules

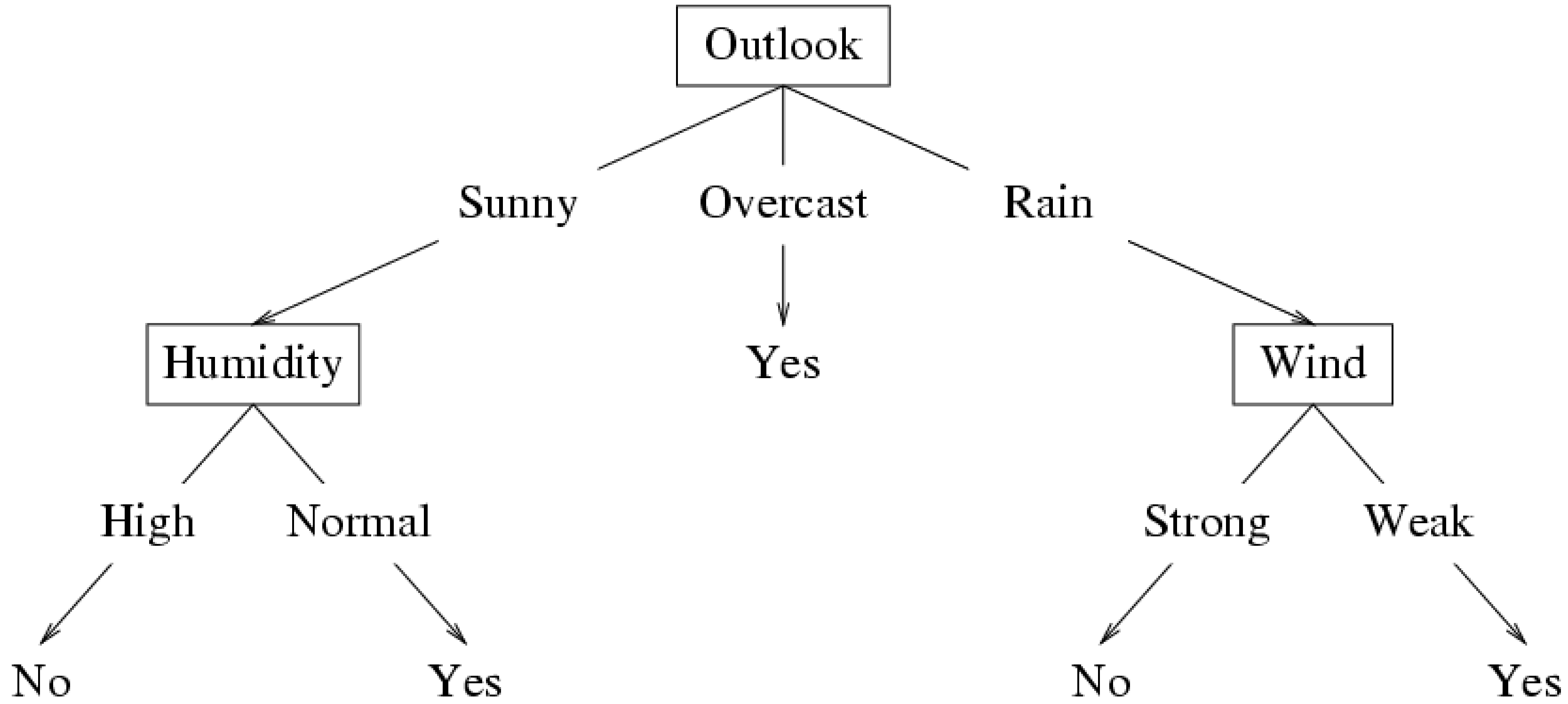


- Any problems?
- Another idea?

Association rules

Day	Outlook	Temp.	Humidity	Wind	PlayTennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Decision tree for the "Play tennis" task



What if some attributes are numeric?

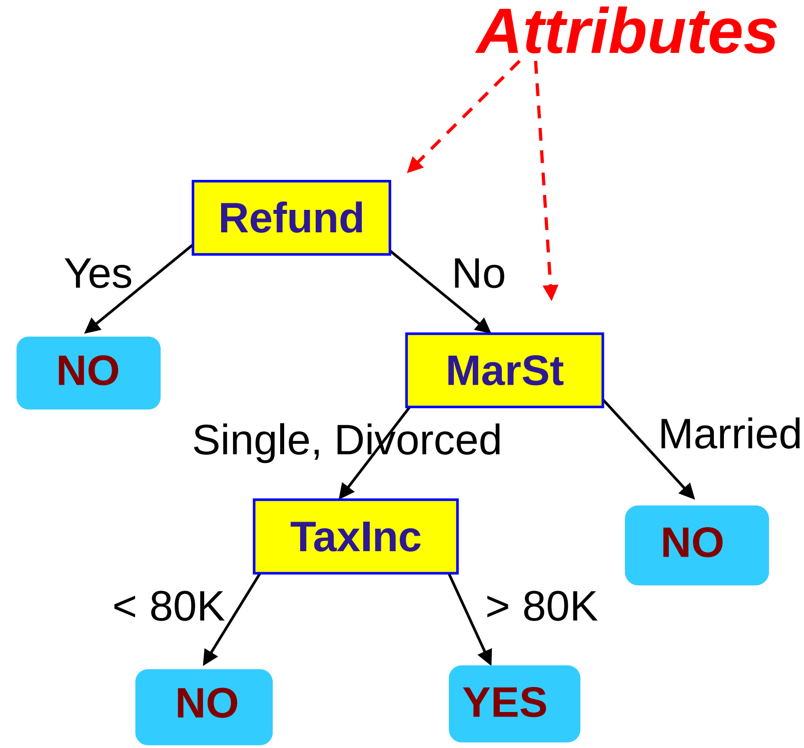
- Input attributes are numerical:
  - Discretization into classes

- The output attribute is numerical:
  - Regression trees

An example of a decision tree learning

nominal nominal continuous class

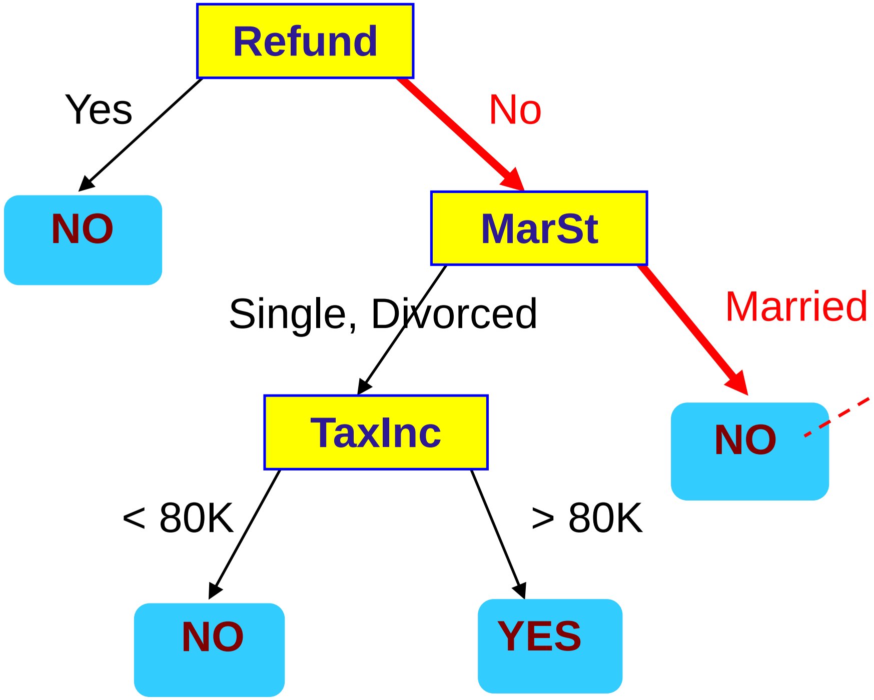
Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**



Training data

Model: Decision tree

An example of the model use (applying model)



Test data

Refund	Marital Status	Taxable Income	Cheat
No	Married	80K	?

The model does not cheat in this case

How to create a tree?

- Manually or using an algorithm for induction of decision trees
- There are dozens of related algorithms, often quite similar to each other, for example:
  - CHAID
  - CART
  - ID3 a C5
  - QUEST
  - GUIDE
  - MARS
  - TreeNet

Tree construction

- Top-down approach
  - Go through the training data and find the attribute that best divides the data into classes.
  - Divide the data by the values of the attribute.
  - Treat each group recursively until it is composed of one class only.

- Bottom-up approach is also possible

Algorithm

BuildTree(Node *t*, Training database *D*,
Split Selection Method ***S***)

(1) Apply ***S*** to *D* to find splitting criterion
(2) **if** (*t* is not a leaf node)
(3) Create children nodes of *t*
(4) Partition *D* into children partitions
(5) Recurse on each partition
(6) **endif**

Algorithm specification

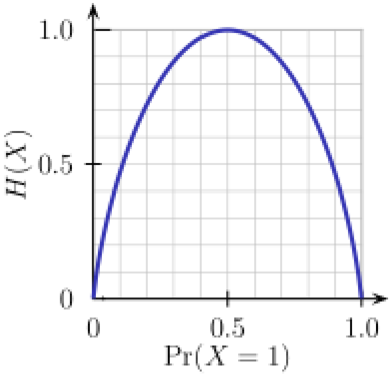
- Two issues that need to be resolved:
  1. The mechanism of division. Usually the "information gain“, Gini index or entropy is measured
  2. Regularization. Either directly a stopping rule or a rule for trimming the tree

Entropy

- Entropy describes the level of disorder. The amount of information within the set **S** can be described as:

$$H(S) = -\sum_{i=1}^{n} P(s_i) \log_2 P(s_i)$$

- where **P(S$_i$)** is the probability that an arbitrary pattern in **S** is of type **S$_i$**.
- Chart of entropy:



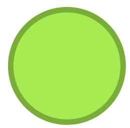
- Select the attributes that minimizes the maximal entropy in the group.

Intuition

- Entropy 1 = random behavior, no useful information.
- Entropy = 0 divides the data according to the classes, significant information.
- Ideally, let’s find an attribute that divides data on “good” and “bad”.

Calculation of entropy

Attributes: shape, color



$$p_{green} = 1$$

$$p_{red} = 0$$

$$H(E_{circle}) = 1\log 1 + 0 \log 0$$

$$= 0$$

$$p_{green} = 1/3$$

$$p_{red} = 2/3$$

$$H(E_{square}) = \frac{1}{3}\log\frac{1}{3} + \frac{2}{3}\log\frac{2}{3}$$

$$= 0{,}92$$

Weighted means of entropies:

$$\sum_{j \in \{circle, square\}} \frac{|E_j|}{|D|} \cdot H(E_j) = \frac{1}{4} \cdot 0 + \frac{3}{4} \cdot 0{,}92 = 0{,}69$$

Information gain

- Information gain compares the entropy before and after the split. Thus, measures how much information we have obtained by the distribution according to the selected attribute.

- $$\text{Information gain} = \text{Entropy(before)} - \text{Entropy(after)}$$

- The calculation is performed for each node of the tree and all its attributes. Attribute with the highest information gain is chosen for the split.

Crisis of information gain

The problem with the information gain is that it prefers attributes with *many* values. For example, it would prefer “Person_ID”, which takes a different value for each row (pattern), despite the fact that for the classification it is the *least useful* attribute of all!

Example – Crisis of information gain

- Does a wife use a contraception?
- We know these parameters:

	IG *reality*	IG *sample*
1. Wife's age: {16, 17, …, 49}.	0,045	0,771
2. Wife's education: {low, med-, med+, high}.	0,044	0,495
3. Husband's education: {low, med-, med+, high}.	0,018	0,281
4. Number of children ever born: {0, 1, …, 16}.	0,113	0,571
5. Wife's religion: {non-islam, islam}.	0,004	0,079
6. Wife working status: {employed, unemployed}.	0,001	0,020
7. Husband's occupation: {low, med-, med+, high}.	0,006	0,020
8. Standard of living: {low, med-, med+, high}.	0,018	0,210
9. Media exposure: {adequate, inadequate}.	0,015	0,144

Example – Crisis of information gain

- Because the sample is *small*, there are only few women at each age. And if there is only one woman at any age, the attribute “age” has the maximal information gain and is *seemingly the best*.

Ratio of information gain

- **The ratio information gain** modifies the information gain so not to tend to selection of attributes with many values.

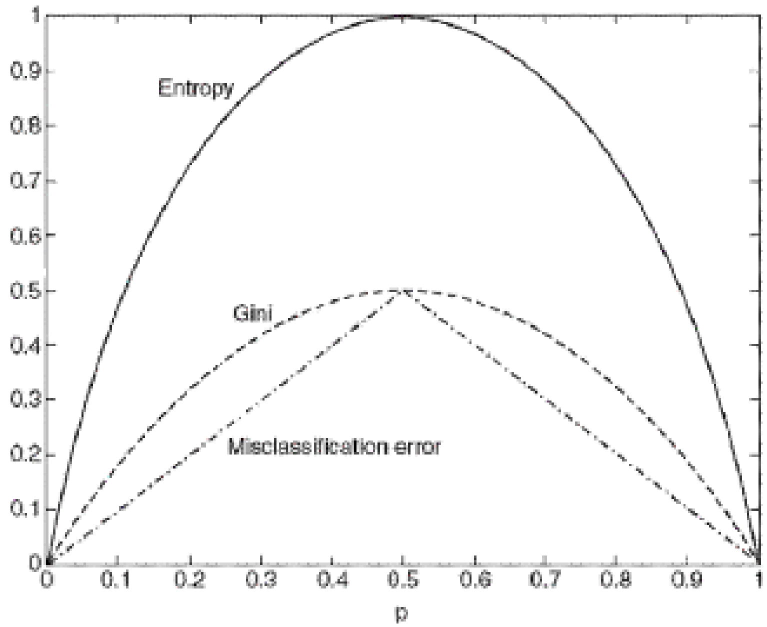
$$\text{Ratio of information gain} = \frac{\text{Information gain}}{\text{Entropy of distrinution of instances to branches}}$$

Gini index of diversity

- **Gini** index expresses "impurity" of a node.

$$Gini = 1 - \sum P_i^2$$

where P_i are relative frequencies in nodes



- The difference between entropy and Gini index is in the shape of the curve.
- Entropy penalizes mixed nodes little more than Gini index, but otherwise they are interchangeable.

Regularization

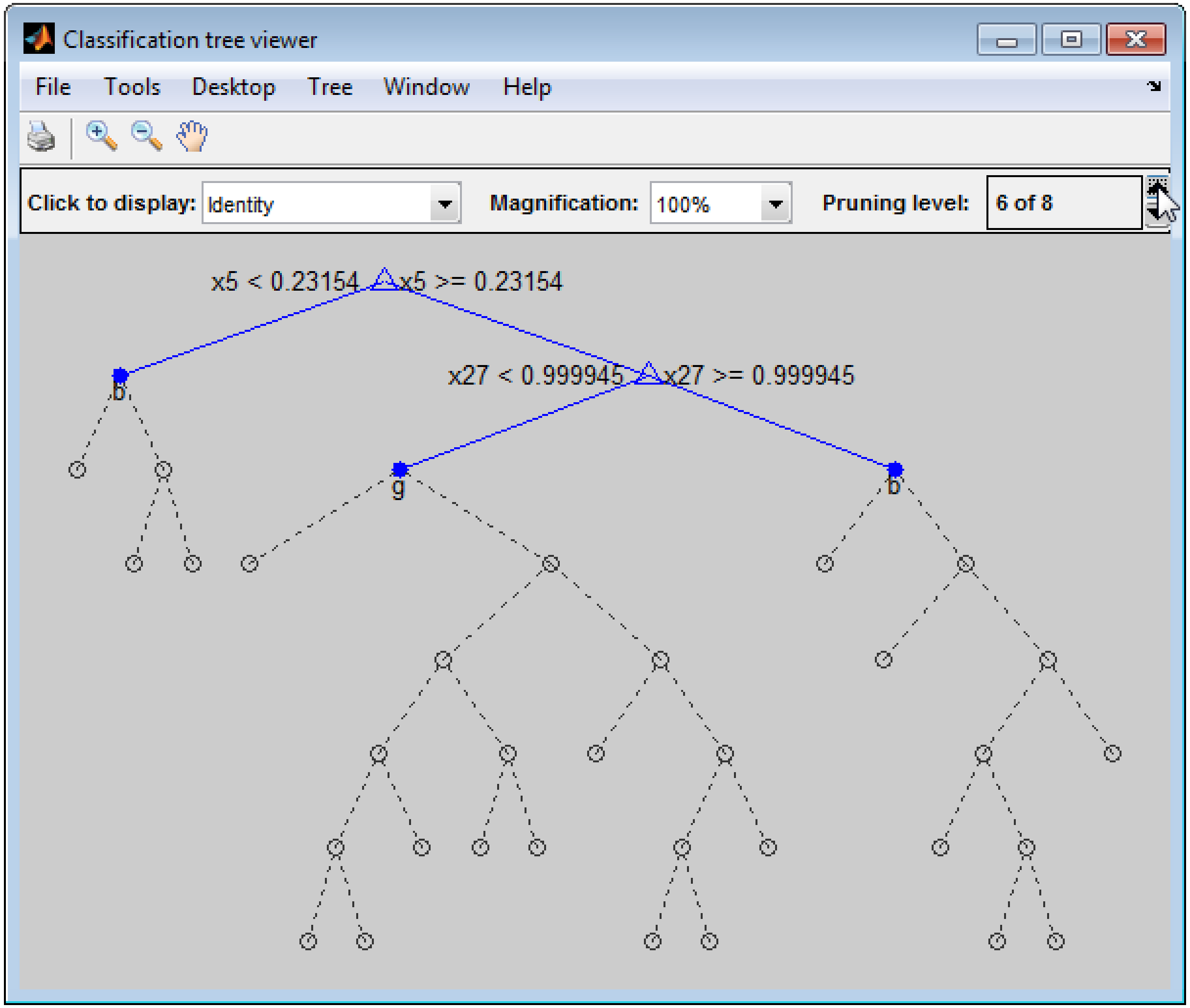
- Can the decision tree be overfitted on training data?
- How to avoid overfitting?

Condition of division stopping

- We do not require absolutely "perfect" tree, which would classify training data with 100 % success, because the resulting tree would tend to be too *large* and *overfitted*.
- Two methods are used:

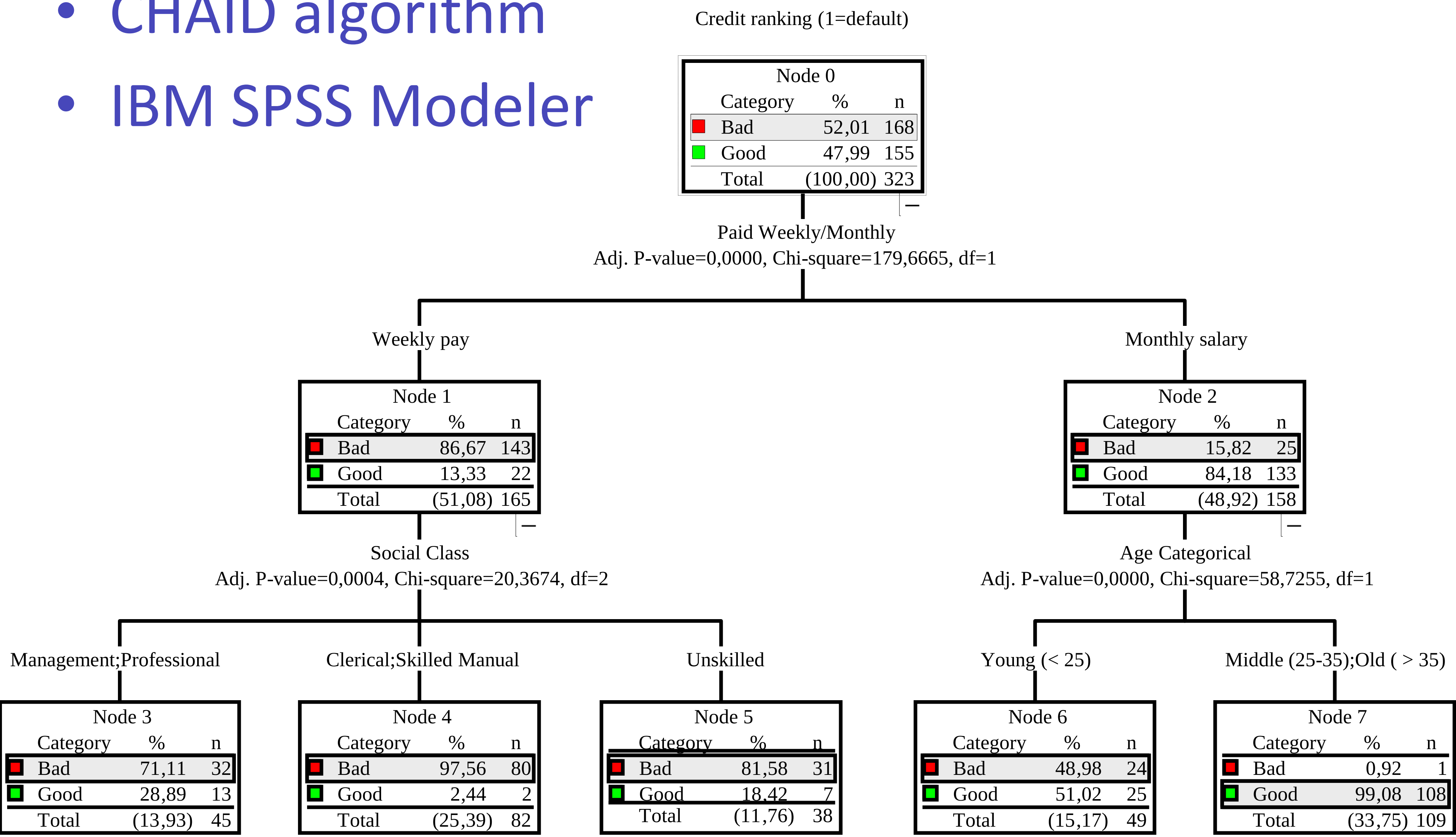
1. **Stopping rule**
   - Stops when there is no statistically significant difference => a leaf is created
   - Typically, the condition of the minimum number of cases in the node and/or leaf is added and/or the maximum depth of the tree is prescribed
2. **Pruning**
   - The tree is allowed to grow to a maximum width
   - This leads to overfitting
   - Therefore , we retroactively remove the leaves and branches, which, according to a properly chosen statistical criterion, can not be considered as significant (cross-validation is often used)
   - Pruning reduces the complexity of the model

Pruning


Classification tree viewer
File Tools Desktop Tree Window Help
Click to display: Identity
Magnification: 100%
Pruning level: 6 of 8
x5 < 0.23154
x5 >= 0.23154
b
x27 < 0.999945
x27 >= 0.999945
g
b

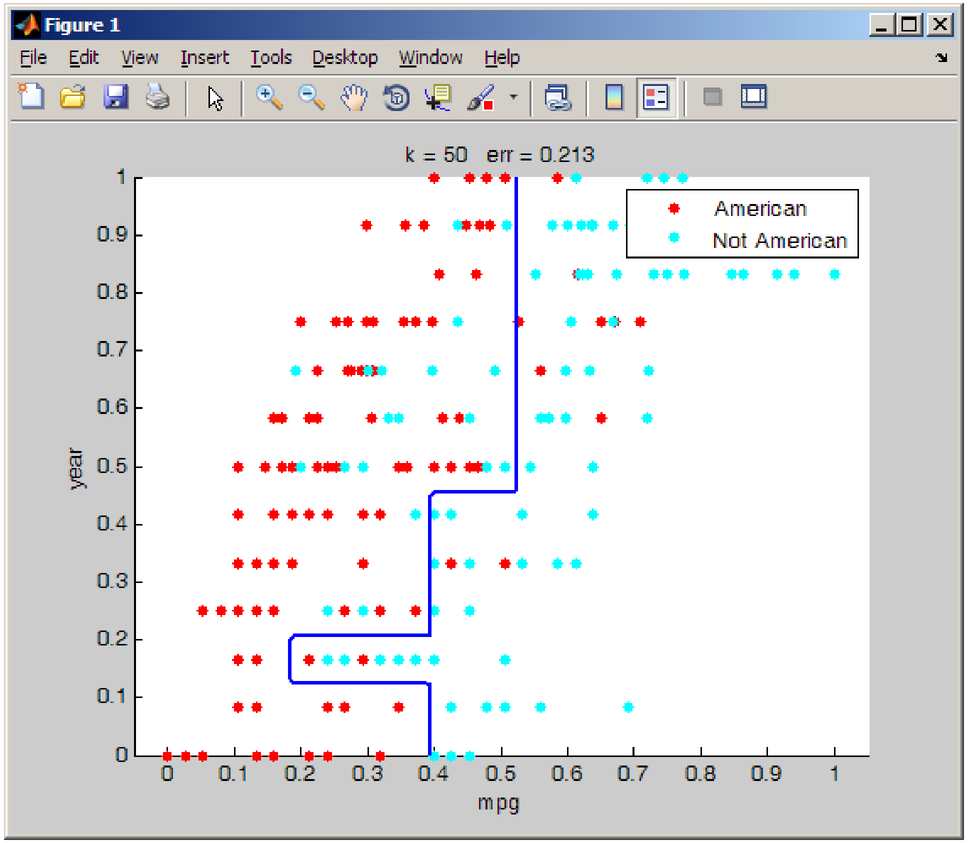
Decision tree visualization

- CHAID algorithm
- IBM SPSS Modeler



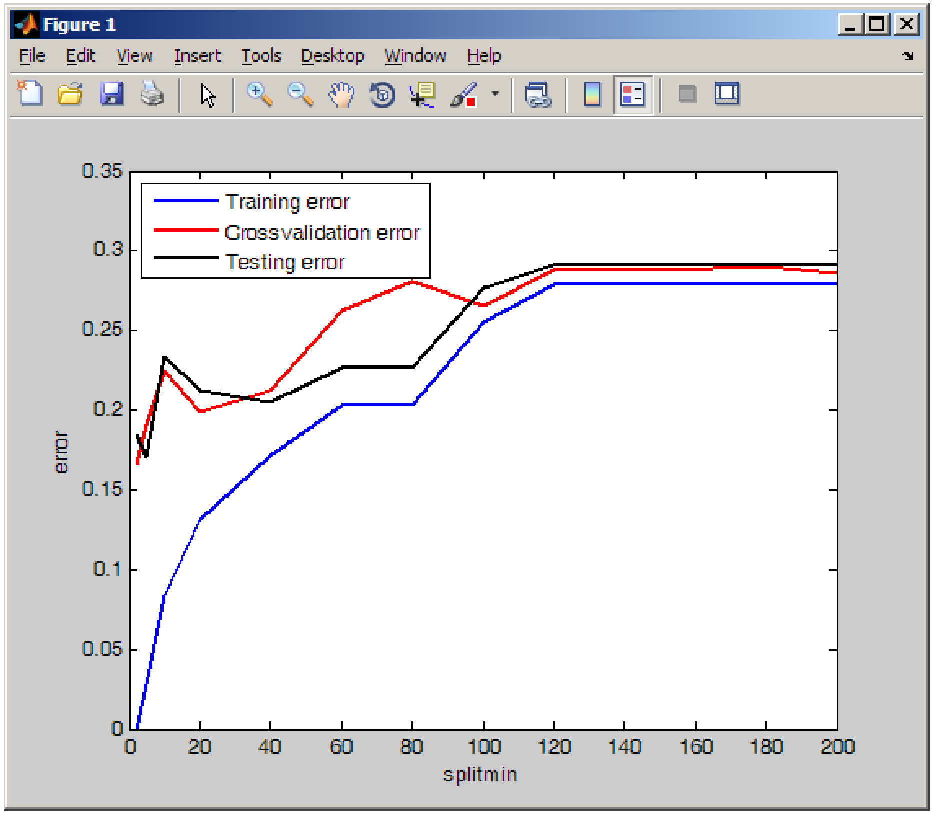
Visualization of the behavior (decision boundaries)

Draw a decision tree that has the following decision boundary:



Model plasticity

- Dependence of the tree error on the parameter “splitmin” (the minimum number of data needed to split the node)
- Why is the training error null pro *splitmin* = 2?



Discussion

- Intuitive interpretation of the results
- Decision trees are slowly learning, but their use is then very fast
- Many algorithms (such as C5.0) calculate missing data, normalize the data, ... Their use is then very simple. And yet the results are good.
- -> Favorite method especially in practice.

Comparison of algorithms in IBM SPSS Modeler

Model	C5.0	CHAID	QUEST	C&R Tree
Split	Multiple	Multiple	Binary	Binary
Continuous output?	No	Yes	No	Yes
Continuous inputs?	Yes	No	Yes	Yes
Criterion of attribute selection	Information gain	Chi-quadrat, F test for continuous variables	statistical	Gini index (purity of division, variability)
Criterion of pruning	Limit of error	Checks for overfiting	Regularization of complexity	Regularization of complexity
Interactive tree construction	No	Yes	Yes	Yes

TreeNet, decision forests

- Instead of one large tree, a single “forest” of small trees is used
- The resulting prediction is calculated by a weighted sum of predictions the individual trees
- Taylor Analogy: Developing into trees
- Poorly interpretable (black box), but robust and accurate; lower demands on the quality and data preparation opposite to neural networks or boosting of standard trees