Introduction to Artificial Intelligence: Methods, Models, Algorithms

Aleksandr I. Panov and Konstantin S. Yakovlev

National Research University Higher School of Economics

20 July 2018 – Summer University

apanov@hse.ru

Linear models

$$a(x) = w_0 + \sum_{j=1}^{d} w_j x^j$$

Weights can be interpreted if features are scaled

Example

- The prediction value of the apartment
- Features: area, floor, number of rooms

$$a(x) = 10 \cdot (\text{area}) + 1.1 \cdot (\text{floor}) + 20 \cdot (\text{number of rooms})$$

Example

- Dependence on the floor is hardly linear
- Quadratic features:

$$a(x) = 10\cdot(\text{area}) + 1.1\cdot(\text{floor}) + 20\cdot(\text{number of rooms}) - 0.2\cdot(\text{area})^2 + 0.5\cdot(\text{area}\cdot\text{number of rooms}) + \ldots$$

Example

- With cubic features will be even better
- How to interpret the feature area $\cdot$ number of rooms^2?
- A total of 20 such features

Example

- You can binarysoul features: $[x^j > t]$
- $(\text{floor} > 1), (\text{floor} > 2), \ldots, (\text{floor} > 30)$
- Features will be orders of magnitude more
- Easier to interpret:
  $-2[\text{floor} > 3][\text{area} < 40][\text{number of rooms} < 3]$
- You can use L_1-regularization

Logical rules

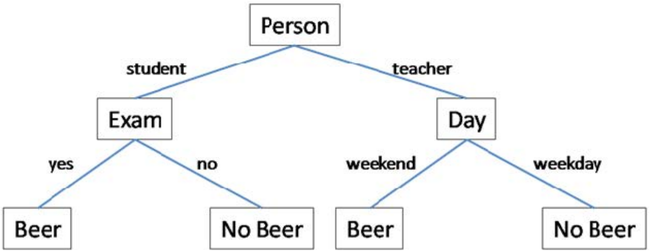
$$[\text{floor} > 3][\text{area} < 40][\text{number of rooms} < 3]$$

- Easy to explain to the customer (if ≤ 5 conditions)
- Allow you to extract knowledge from data
- Not the fact that they are optimal in terms of quality

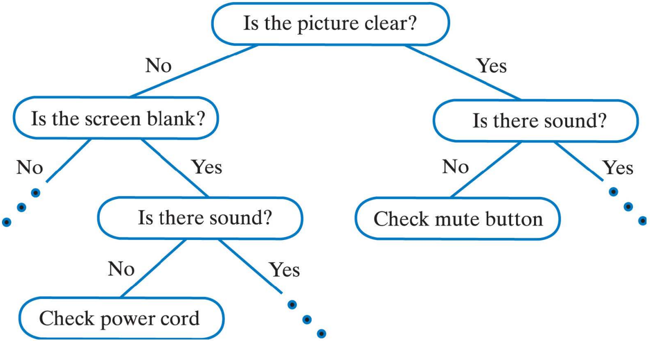
Logical rules

- How to construct them?
- Linear model
- Busting, greedy build-up
- Decision trees

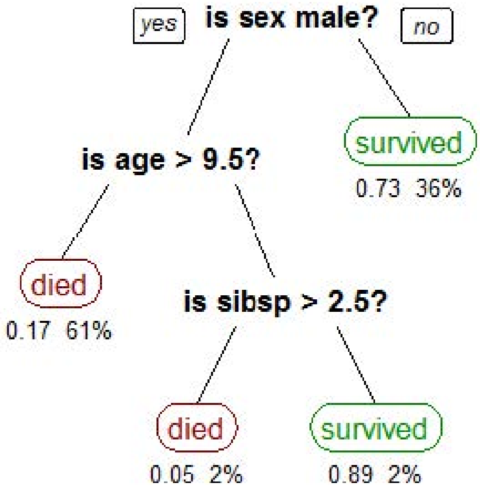
Decision making



The scheme of dialogue with the client

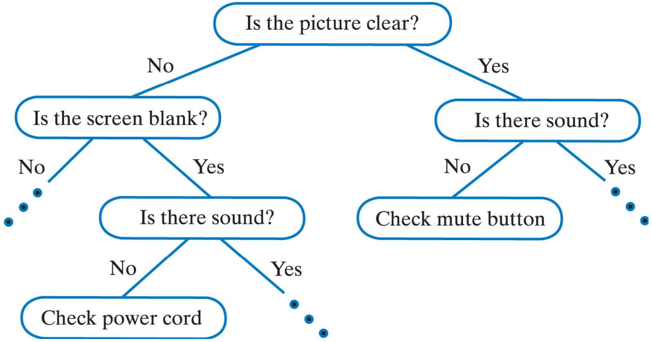


The passengers of the Titanic



Decision tree

- Binary tree
- Each inner node contains a condition
- Each leaf contains prediction (solution)



Conditions

- Most popular options:

$$[x^j \leq t] \; and \; [x^j = t]$$

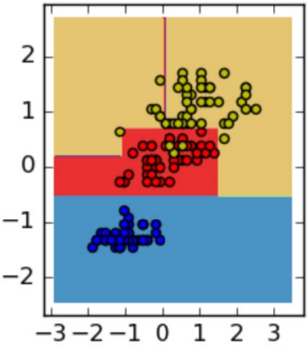
- Examples:

$$[\mathrm{floor} = 5] \; or \; [\mathrm{area} \leq 30]$$

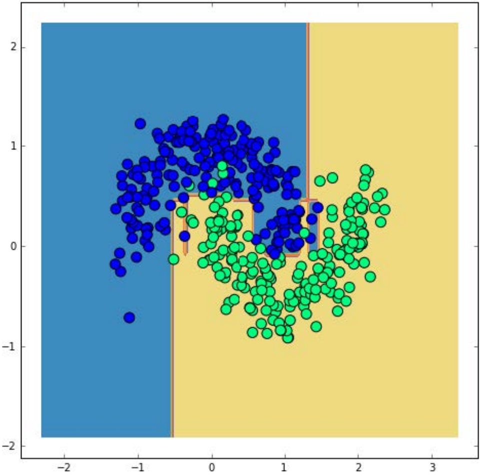
Prediction in the leaf

- Regression: Real number
- Classification: Class or Class probabilities

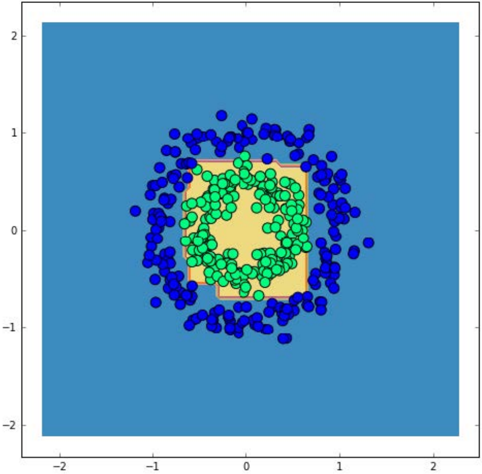
Classification



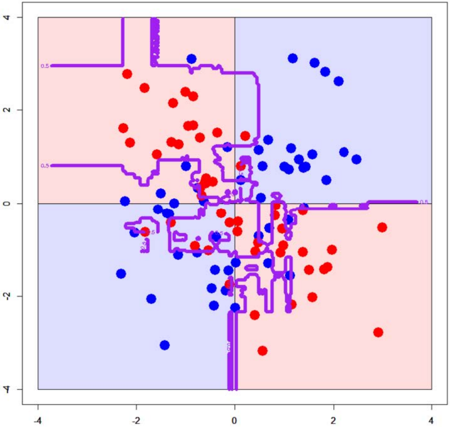
Classification



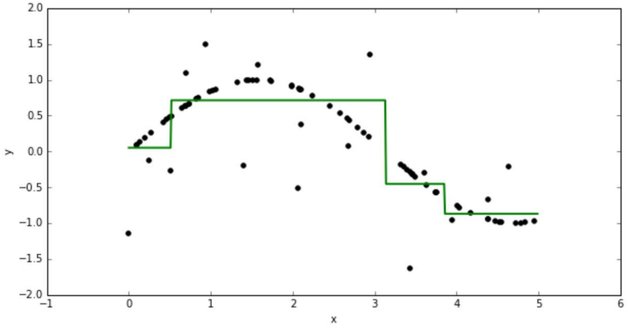
Classification



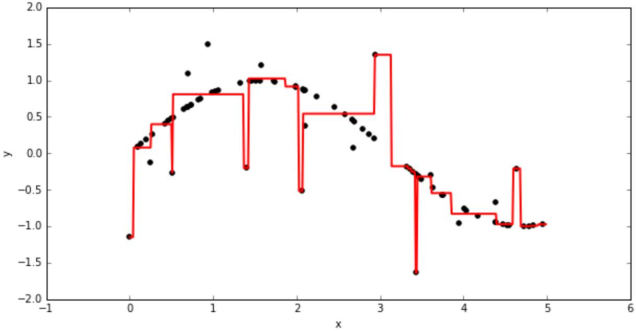
Classification



Regression



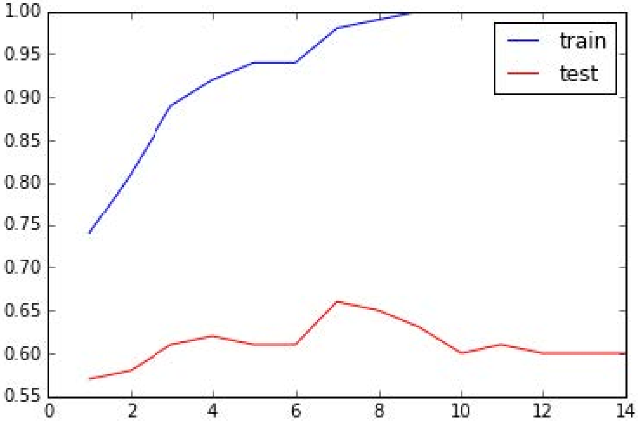
Regression



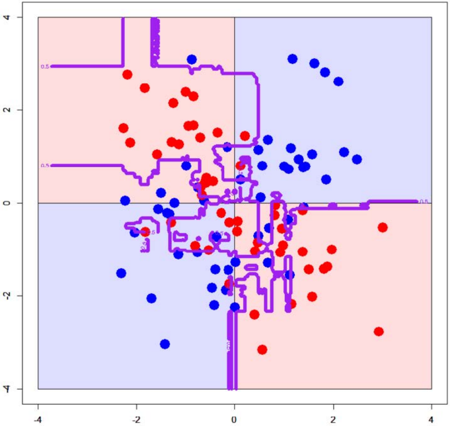
Decision tree

- Restore complex dependencies
- Can build any complex surface
- The greater the depth the more complex the surface
- Prone to overfitting

Depth of trees



Overfitting of trees

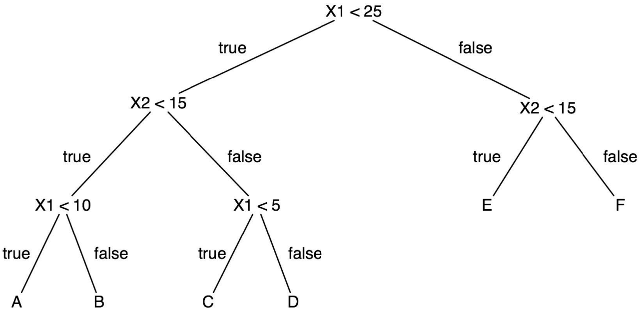


Overfitting of trees

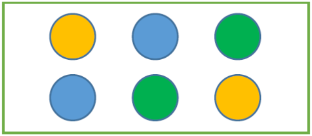
- The tree can achieve zero error on any sample
- Tackling overfitting: the minimum tree among all with zero error
- NP-complete task
- *Solution*: greedy building

Gready formation

- Grow the tree from root to leaves

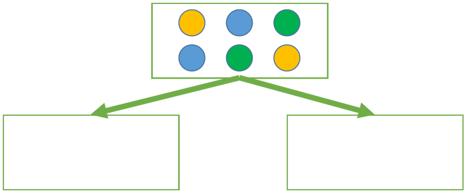


Gready formation

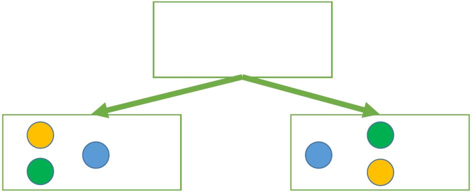


How to split the node?

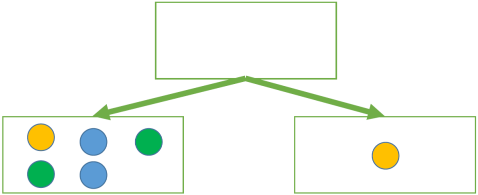
Gready formation



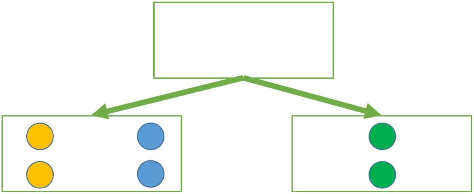
Gready formation



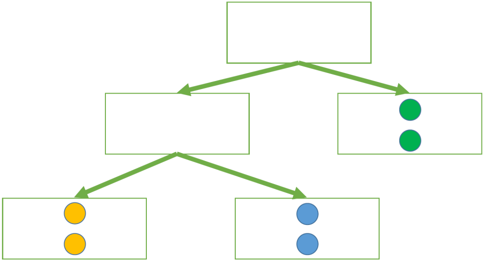
Gready formation



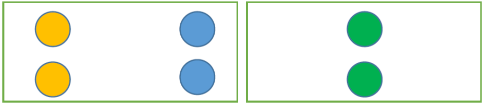
Gready formation



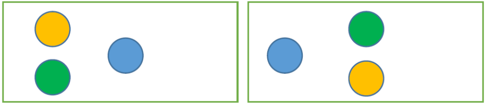
Gready formation



How to compare splits?

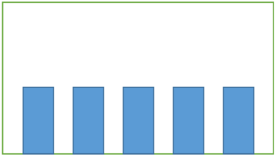
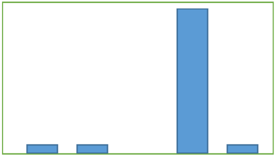


OR



Entropy

- Measure of uncertainty of distribution

Entropy

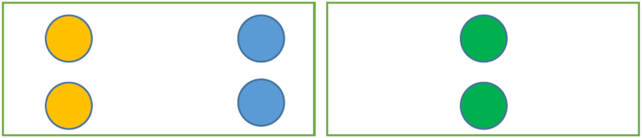
- Discrete distribution
- Accepts n values with probabilities $p_1, \dots, p_n$
- Entropy:

$$H(p_1, \dots, p_n) = -\sum_{i=1}^{n} p_i \log p_i$$

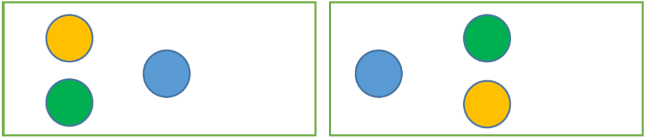
Entropy

- $(0.2, 0.2, 0.2, 0.2, 0.2) \to H = 1.60944$
- $(0.9, 0.05, 0.05, 0, 0) \to H = 0.394398$
- $(0, 0, 0, 1, 0) \to H = 0$

Entropy

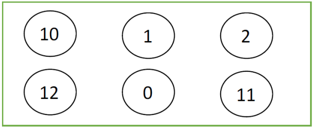


- (0.5, 0.5, 0) and (0, 0, 1)
- $H = 0.693 + 0 = 0.693$

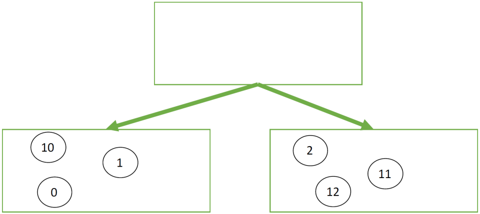


- (0.33, 0.33, 0.33) and (0.33, 0.33, 0.33)
- $H = 1.09 + 1.09 = 2.18$

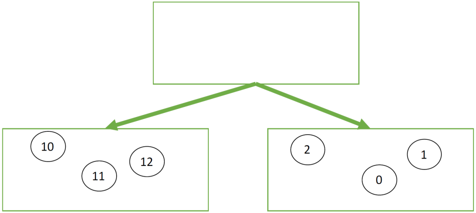
What about regression?



What about regression?



What about regression?



What about regression?

- Choose the partition with the least total variance
- The smaller the variance, the less uncertainty

Searching the partition

- Let the node m be the sample X_m
- $Q(X_m, j, t)$ - is a criteria for the condition error $[x^j \leq t]$
- Search the parameters t and j:

$$Q(X_m, t, j) \to \min_{j,t}$$

Searching the partition

- When we find the partition we split the X_m into two parts:

$$X_l = \{x \in X_m | [x^j \leq t]\}$$
$$X_r = \{x \in X_m | [x^j > t]\}$$

- Repeat the procedure for child nodes

Stop criterion

- At what point should the splitting of nodes be stopped?
- The single item at the node of ?
- Items of the same class at the node?
- Did the depth exceed a threshold?

Prediction in the leaf

- For example, I decided to make a node m leaf
- Which prediction to choose?
- Regression:

$$a_m = \frac{1}{|X_m|} \sum_{i \in X_m} y_i$$

- Classification

$$a_m = \arg\max_{y \in \mathbb{Y}} \sum_{i \in X_m} [y_i = y]$$

Prediction in the leaf

- For example, I decided to make a node m leaf
- Which prediction to choose?
- Class probabilities:

$$a_{mk} = \frac{1}{|X_m|} \sum_{i \in X_m} [y_i = k]$$

Summary

- Sometimes the model needs to be interpreted
- Decision trees are easy to explain
- Decision trees easily overfits
- Tree construction is a greedy algorithm