

# Machine Learning

## Lecture 5: Decision trees and Ensembles

Radoslav Neychev



# Outline

1. Decision tree: intuition
2. Decision tree construction procedure
3. Information criteria
4. Pruning
5. Decision trees special highlights
  - Decision tree as linear model
  - Dealing with missing data
  - Categorical features
6. Bootstrap and Bagging
7. Random Forest

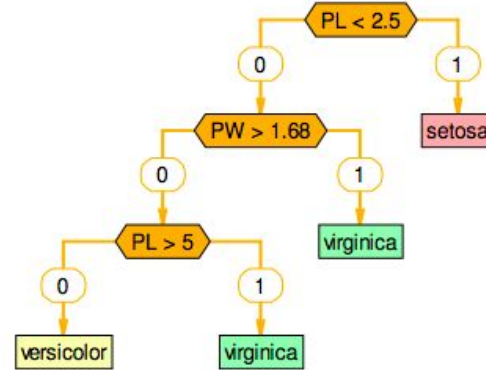
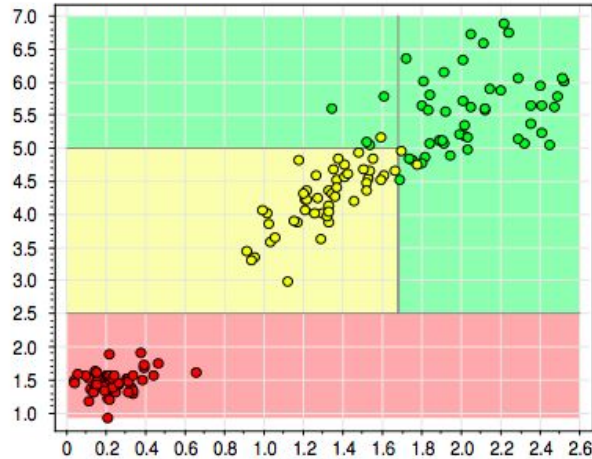
# Decision Tree: intuition

---

girafe  
ai

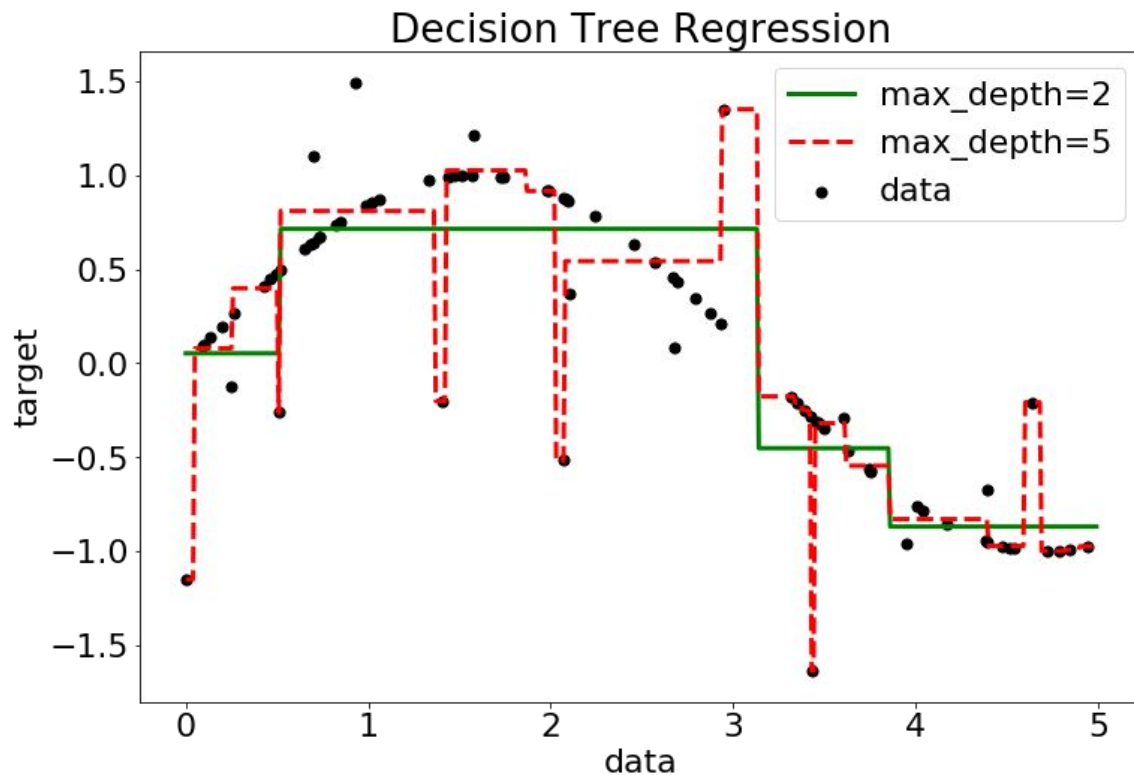
01

# Decision tree for Iris data set



setosa	$r_1(x) = [PL \leq 2.5]$
virginica	$r_2(x) = [PL > 2.5] \wedge [PW > 1.68]$
virginica	$r_3(x) = [PL > 5] \wedge [PW \leq 1.68]$
versicolor	$r_4(x) = [PL > 2.5] \wedge [PL \leq 5] \wedge [PW < 1.68]$

# Decision tree in regression



Green - decision tree of depth 2

Red - decision tree of depth 5

Every leaf corresponds to some constant.

# Decision Tree construction procedure

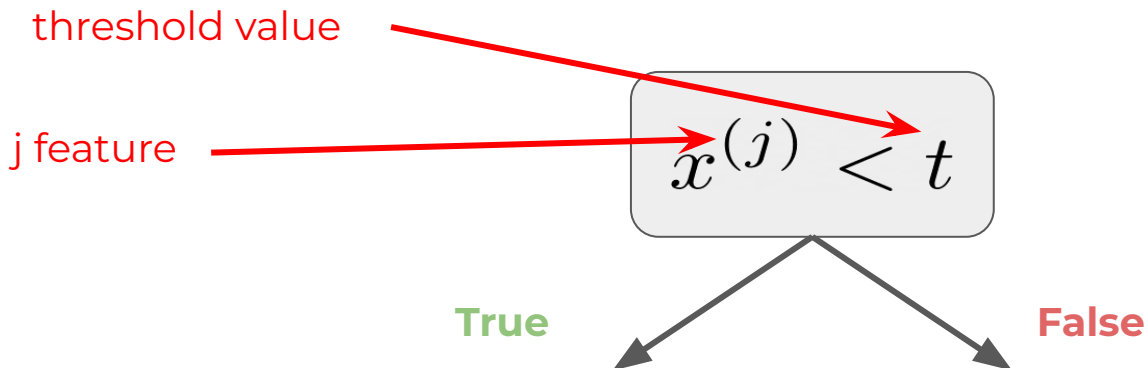
---

girafe  
ai

02



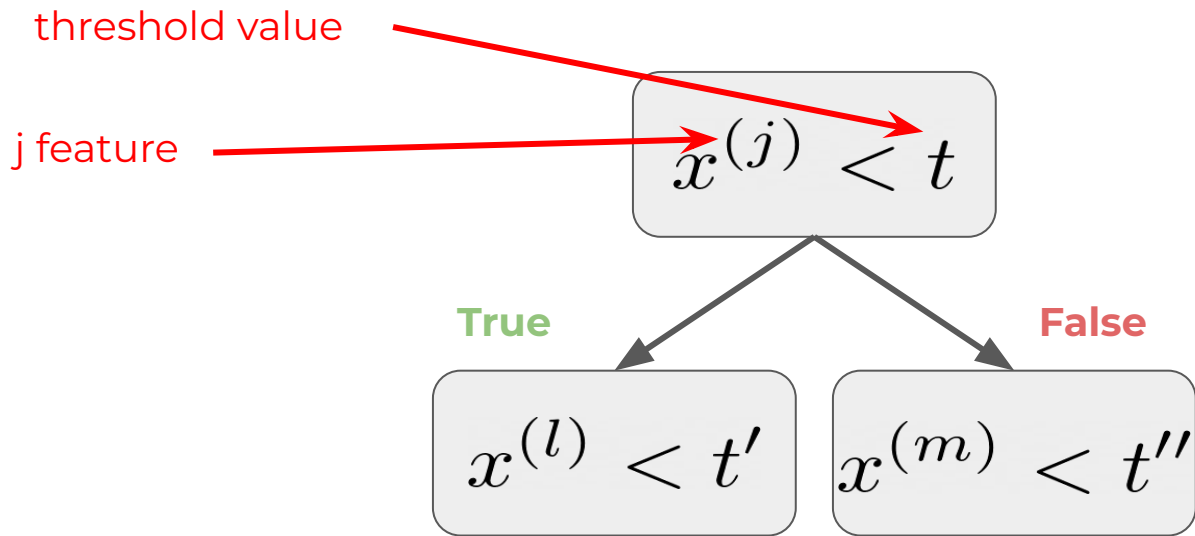
# Constructing decision trees



1. Make a split



# Constructing decision trees

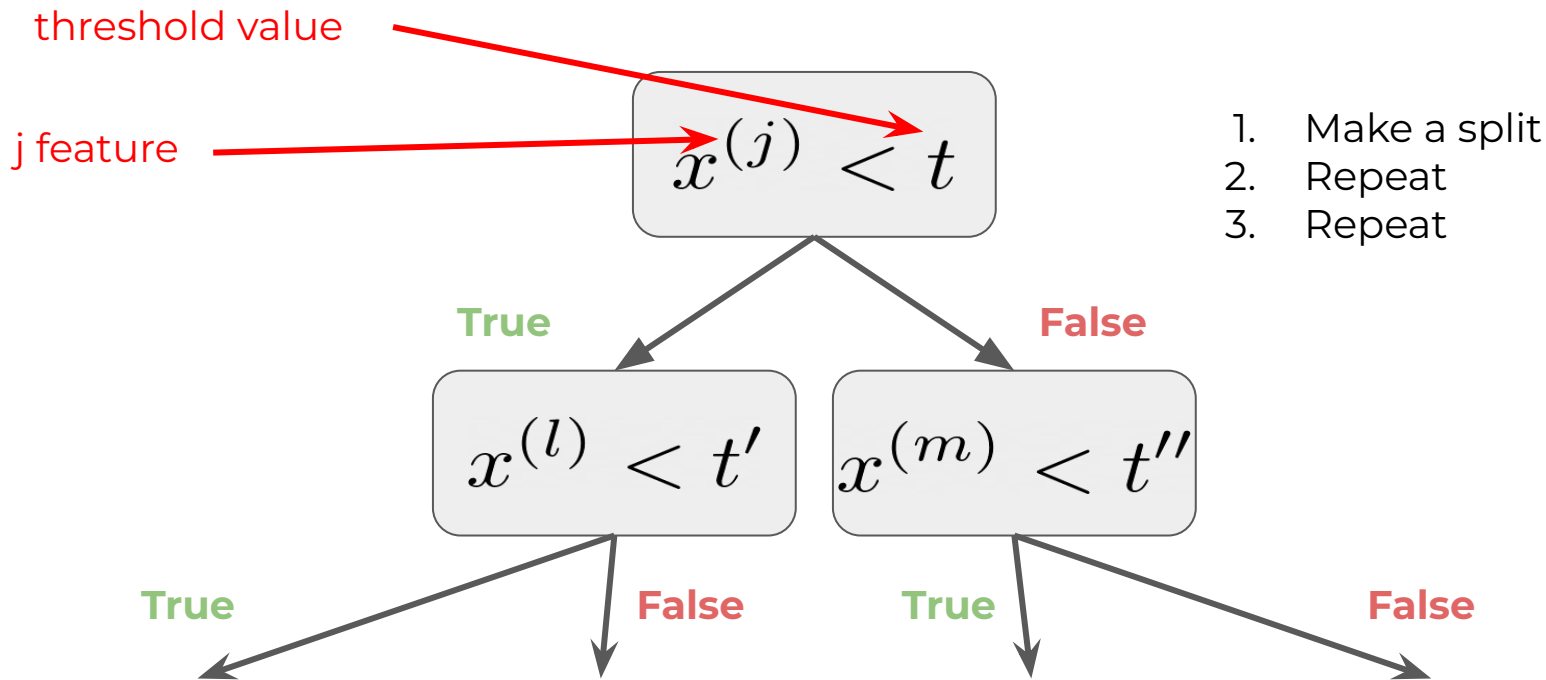


1. Make a split
2. Repeat



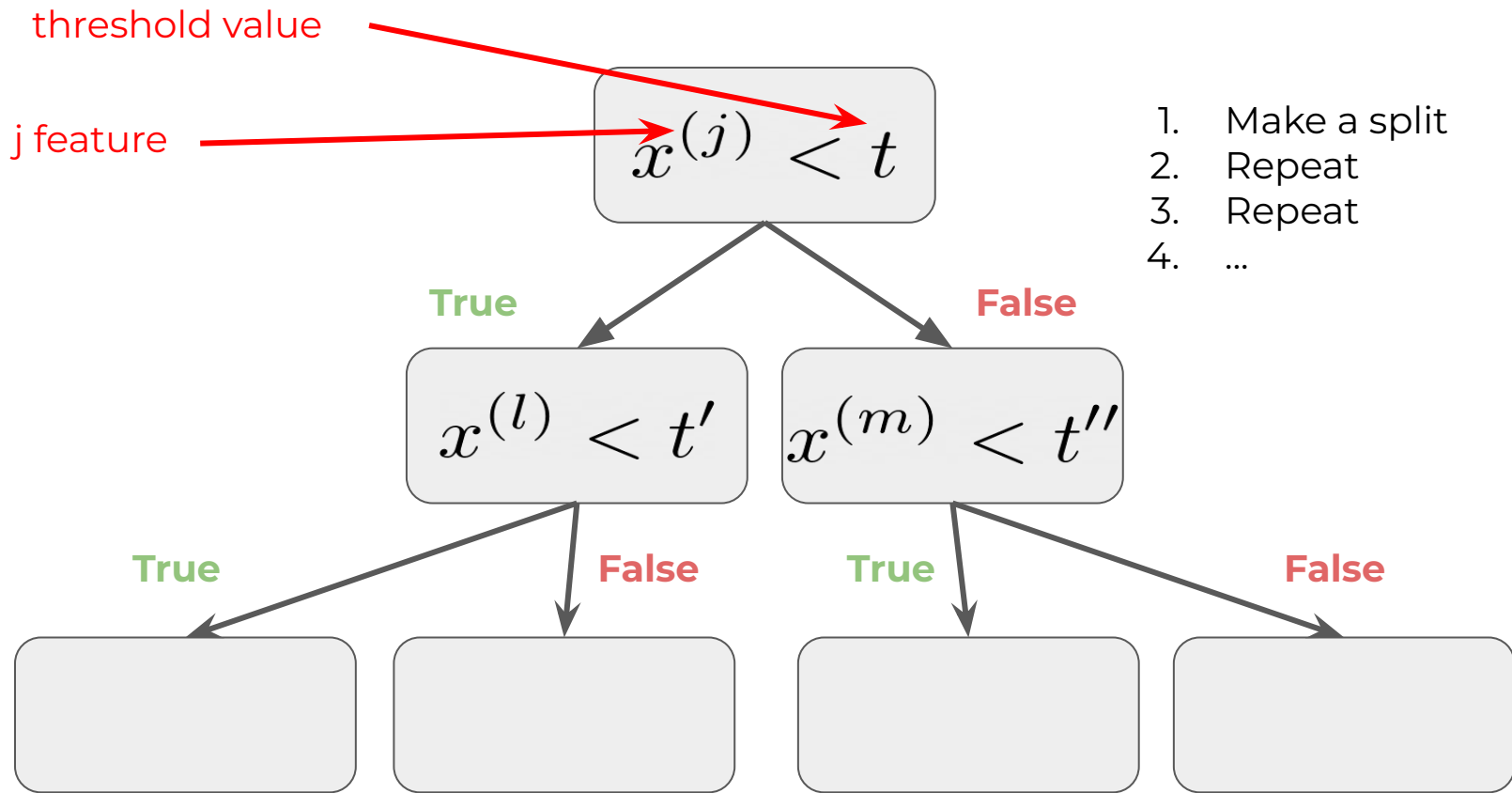


# Constructing decision trees





# Constructing decision trees

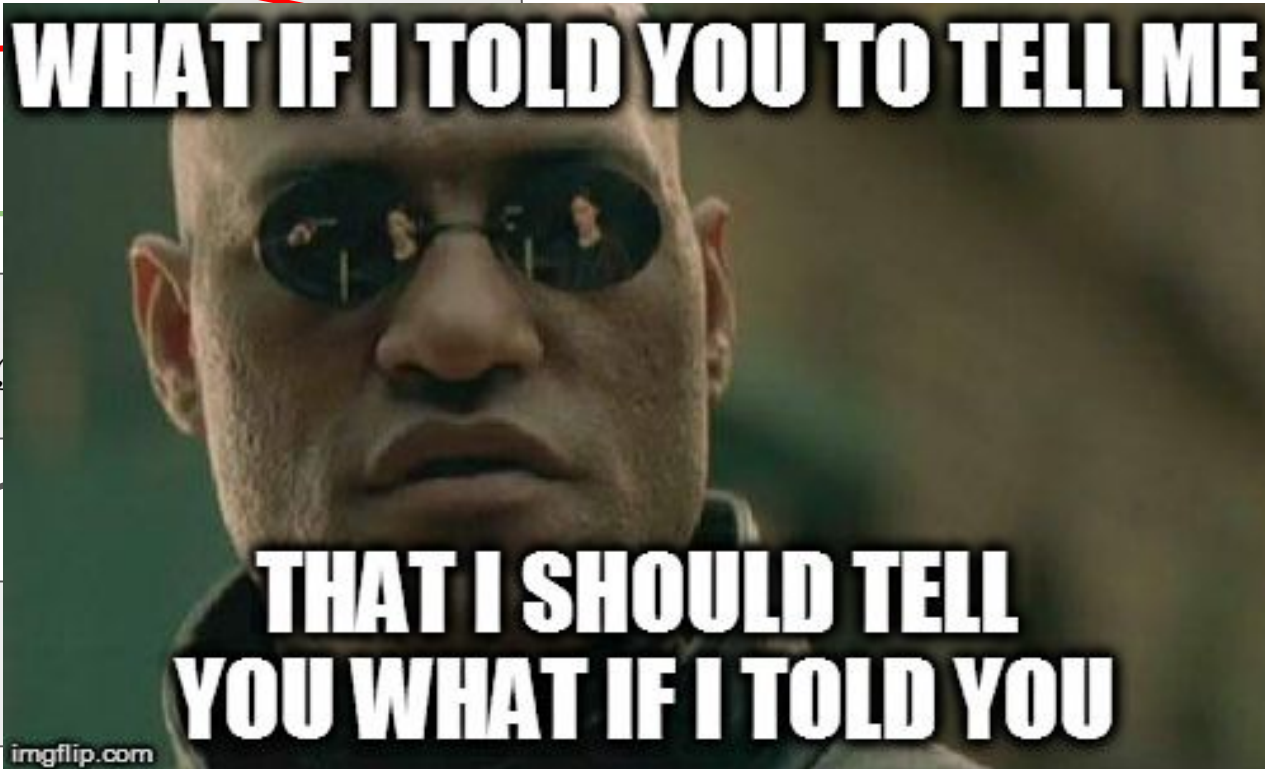




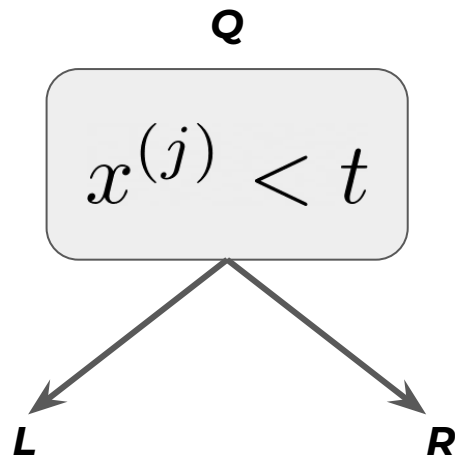
threshold value

j feature

True



# How to split data properly?



What is H?

$$\frac{|L|}{|Q|} H(L) + \frac{|R|}{|Q|} H(R) \longrightarrow \min_{j,t}$$

# Information criteria

---

girafe  
ai

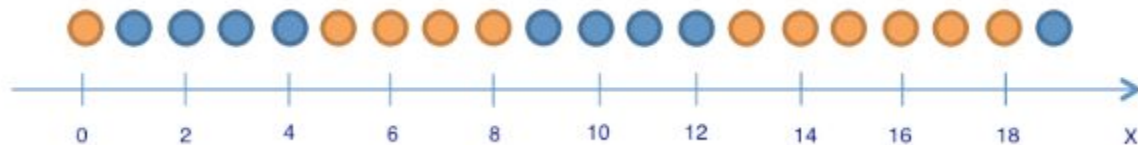
03



# Information criteria

$H(R)$  is measure of “heterogeneity” of our data.

Consider binary classification problem:

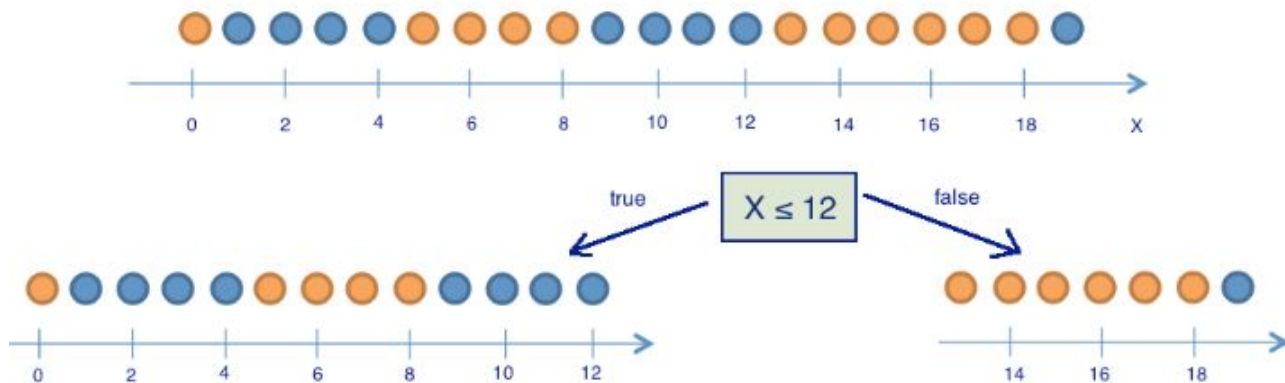




# Information criteria

$H(R)$  is measure of “heterogeneity” of our data.

Consider binary classification problem:





# Information criteria

$H(R)$  is measure of “heterogeneity” of our data.

Consider **binary classification** problem:

Obvious way:

$$H(R) = 1 - \max\{p_0, p_1\}$$

Misclassification criteria:

1. Entropy criteria:  $H(R) = -p_0 \log p_0 - p_1 \log p_1$

2. Gini impurity:  $H(R) = 1 - p_0^2 - p_1^2 = 1 - 2p_0p_1$





# Information criteria

$H(R)$  is measure of “heterogeneity” of our data.

Consider **multiclass classification** problem:

Obvious way:

$$H(R) = 1 - \max_k \{p_k\}$$

Misclassification criteria:

1. Entropy criteria:

$$H(R) = - \sum_{k=0}^K p_k \log p_k$$

2. Gini impurity:

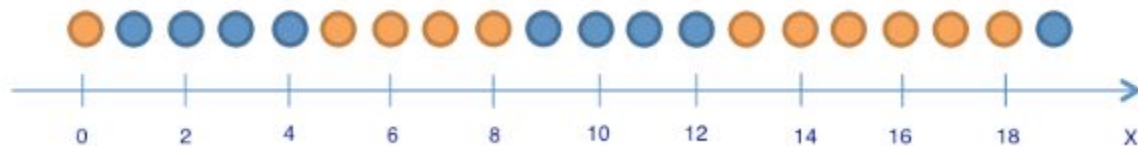
$$H(R) = 1 - \sum_k (p_k)^2$$



# Information criteria

$H(R)$  is measure of “heterogeneity” of our data.

Consider binary classification problem:

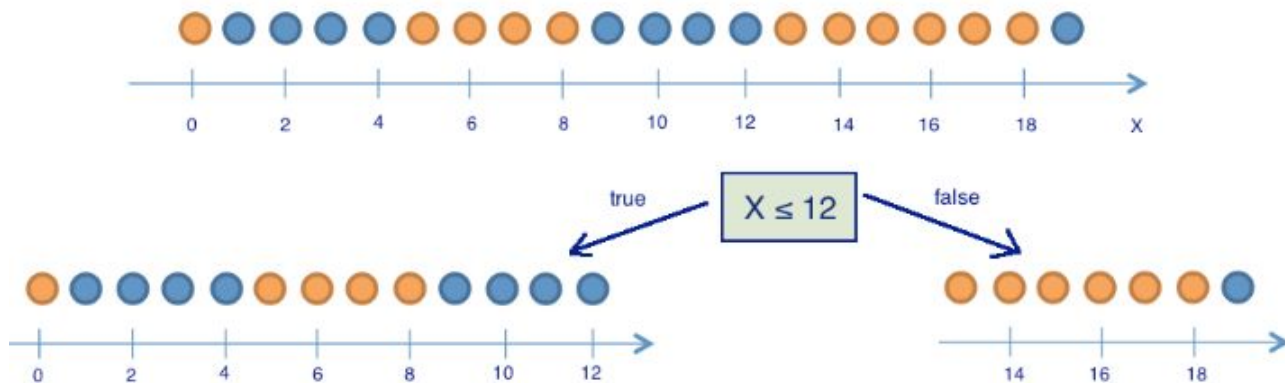


# Information criteria



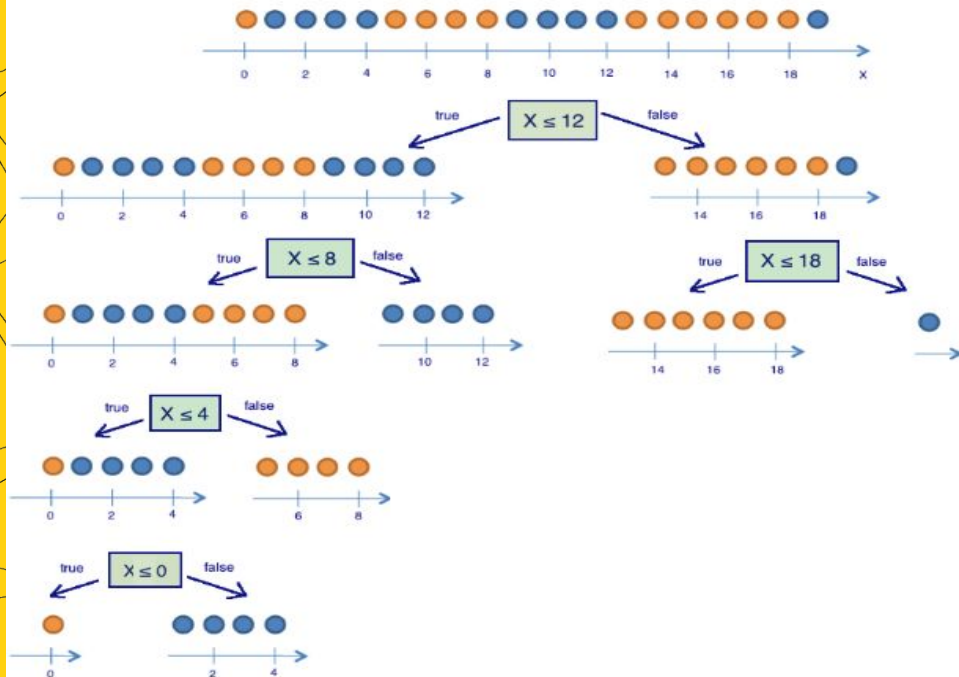
$H(R)$  is measure of “heterogeneity” of our data.

Consider binary classification problem:





# Information criteria: Entropy



$$S = -M \sum_{k=0}^K p_k \log p_k$$

In binary case  $N = 2$

$$S = -p_+ \log_2 p_+ - p_- \log_2 p_- = -p_+ \log_2 p_+ - (1 - p_+) \log_2 (1 - p_+)$$

source: <https://habr.com/ru/company/ods/blog/322534/>

# Information criteria: Gini impurity



$$G = 1 - \sum_k (p_k)^2$$

In binary case  $N = 2$

$$G = 1 - p_+^2 - p_-^2 = 1 - p_+^2 - (1 - p_+)^2 = 2p_+(1 - p_+)$$



# Information criteria

$H(R)$  is measure of “heterogeneity” of our data.

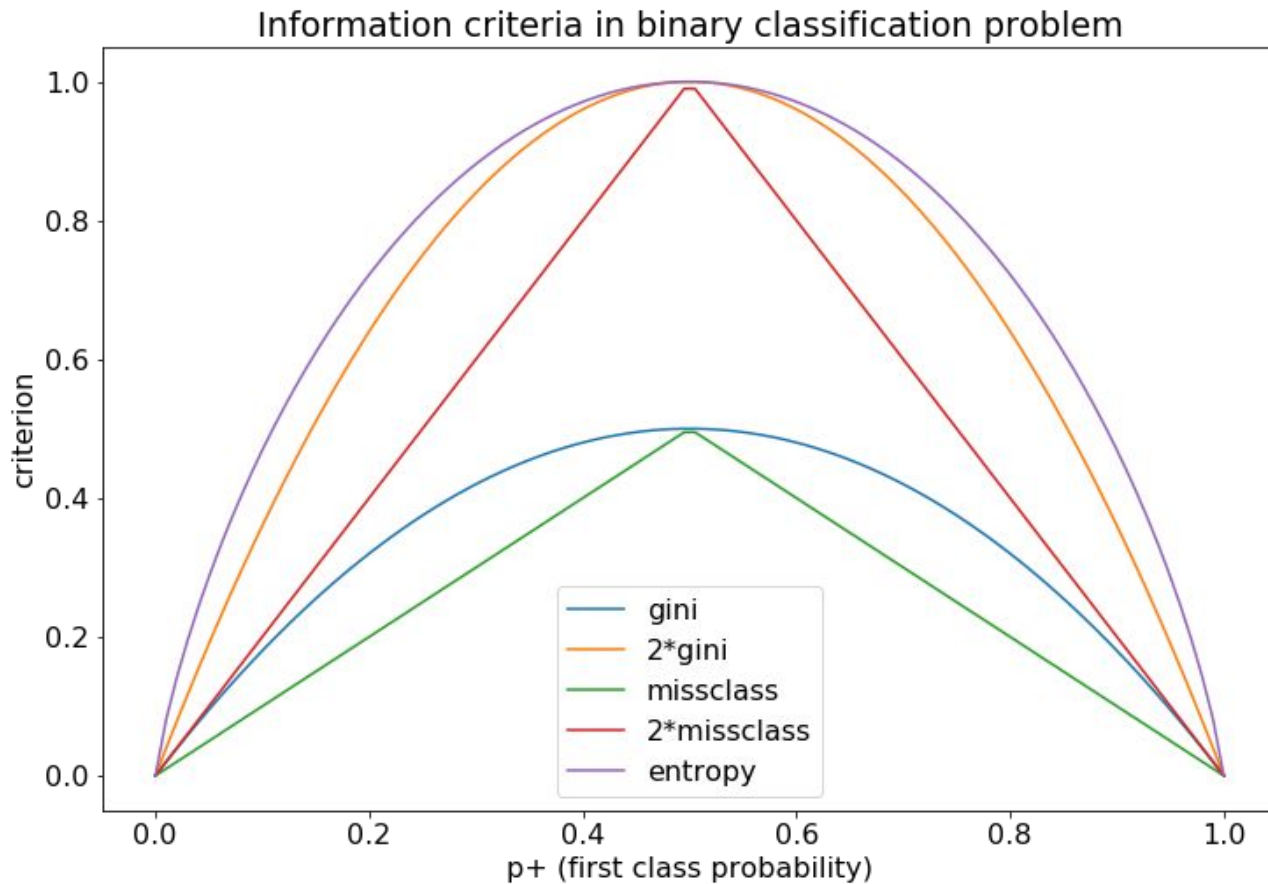
Consider **multiclass classification** problem:

Obvious way: Misclassification criteria:  $H(R) = 1 - \max_k \{p_k\}$

1. Entropy criteria:  $H(R) = - \sum_k p_k \log_2 p_k$

2. Gini impurity:  $H(R) = 1 - \sum_k (p_k)^2$

# Information criteria





# Information criteria

$H(R)$  is measure of “heterogeneity” of our data.

Consider **regression** problem:

1. Mean squared error

$$H(R) = \min_c \frac{1}{|R|} \sum_{(x_i, y_i) \in R} (y_i - c)^2$$

What is the constant?

$$c^* = \frac{1}{|R|} \sum_{y_i \in R} y_i$$



# Pruning

---

girafe  
ai

04

# Pruning



- Pre-pruning:
  - Constrain the tree before construction
- Post-pruning:
  - Simplify constructed tree

# Special highlights

---

girafe  
ai

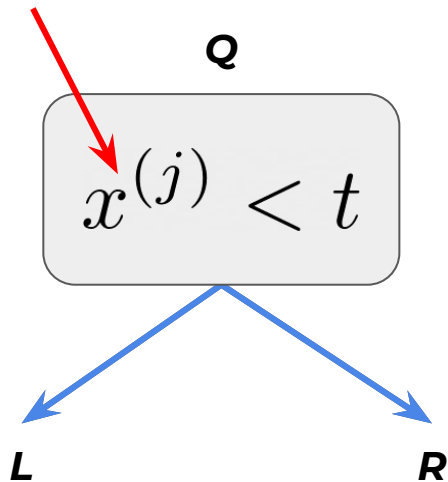
05

# Missing values in Decision Trees



If the value is missing, one might use both sub-trees and average their predictions

Missing value



$$\hat{y} = \frac{|L|}{|Q|} \hat{y}_L + \frac{|R|}{|Q|} \hat{y}_R$$

# Decision Trees as Linear models



Let  $J$  be the subspace of the original feature space, corresponding to the leaf of the tree.

Prediction takes form

$$\hat{y} = \sum_j w_j [x \in J_j]$$

# Construction algorithms: overview



- ID-3
  - Entropy criteria; Stops when no more gain available
- C4.5
  - Normalised entropy criteria; Stops depending on leaf size; Incorporates pruning
- C5.0
  - Some updates on C4.5
- CART
  - Gini criteria; Cost-complexity Pruning; Surrogate predicates for missing data;
- etc.

# Bootstrap and Bagging

---

girafe  
ai

06



# Bootstrap

Consider dataset  $X$  containing  $m$  objects.

Pick  $m$  objects with return from  $X$  and repeat in  $N$  times to get  $N$  datasets.

Error of model trained on  $X_j$ :  $\varepsilon_j(x) = b_j(x) - y(x), \quad j = 1, \dots, N,$

Then  $\mathbb{E}_x(b_j(x) - y(x))^2 = \mathbb{E}_x \varepsilon_j^2(x).$

The mean error of  $N$  models:  $E_1 = \frac{1}{N} \sum_{j=1}^N \mathbb{E}_x \varepsilon_j^2(x).$



# Bootstrap



Consider the errors unbiased and uncorrelated:

$$\mathbb{E}_x \varepsilon_j(x) = 0;$$

$$\mathbb{E}_x \varepsilon_i(x) \varepsilon_j(x) = 0, \quad i \neq j.$$

The final model averages all predictions:

$$a(x) = \frac{1}{N} \sum_{j=1}^N b_j(x).$$

Error decreased by N times!

$$\begin{aligned} E_N &= \mathbb{E}_x \left( \frac{1}{N} \sum_{j=1}^n b_j(x) - y(x) \right)^2 = \\ &= \mathbb{E}_x \left( \frac{1}{N} \sum_{j=1}^N \varepsilon_j(x) \right)^2 = \\ &= \frac{1}{N^2} \mathbb{E}_x \left( \sum_{j=1}^N \varepsilon_j^2(x) + \underbrace{\sum_{i \neq j} \varepsilon_i(x) \varepsilon_j(x)}_{=0} \right) = \\ &= \frac{1}{N} E_1. \end{aligned}$$

# Bootstrap



Consider the errors ~~unbiased and uncorrelated~~:

$$\mathbb{E}_x \varepsilon_j(x) = 0;$$

$$\mathbb{E}_x \varepsilon_i(x) \varepsilon_j(x) = 0, \quad i \neq j.$$

This is a lie

The final model averages all predictions:

$$a(x) = \frac{1}{N} \sum_{j=1}^N b_j(x).$$

Error decreased by N times!

$$\begin{aligned} E_N &= \mathbb{E}_x \left( \frac{1}{N} \sum_{j=1}^n b_j(x) - y(x) \right)^2 = \\ &= \mathbb{E}_x \left( \frac{1}{N} \sum_{j=1}^N \varepsilon_j(x) \right)^2 = \\ &= \frac{1}{N^2} \mathbb{E}_x \left( \sum_{j=1}^N \varepsilon_j^2(x) + \underbrace{\sum_{i \neq j} \varepsilon_i(x) \varepsilon_j(x)}_{=0} \right) = \\ &= \frac{1}{N} E_1. \end{aligned}$$

# Bagging = Bootstrap aggregating



Decreases the **variance** if the basic algorithms are not correlated.

# Random Forest

---

girafe  
ai

07

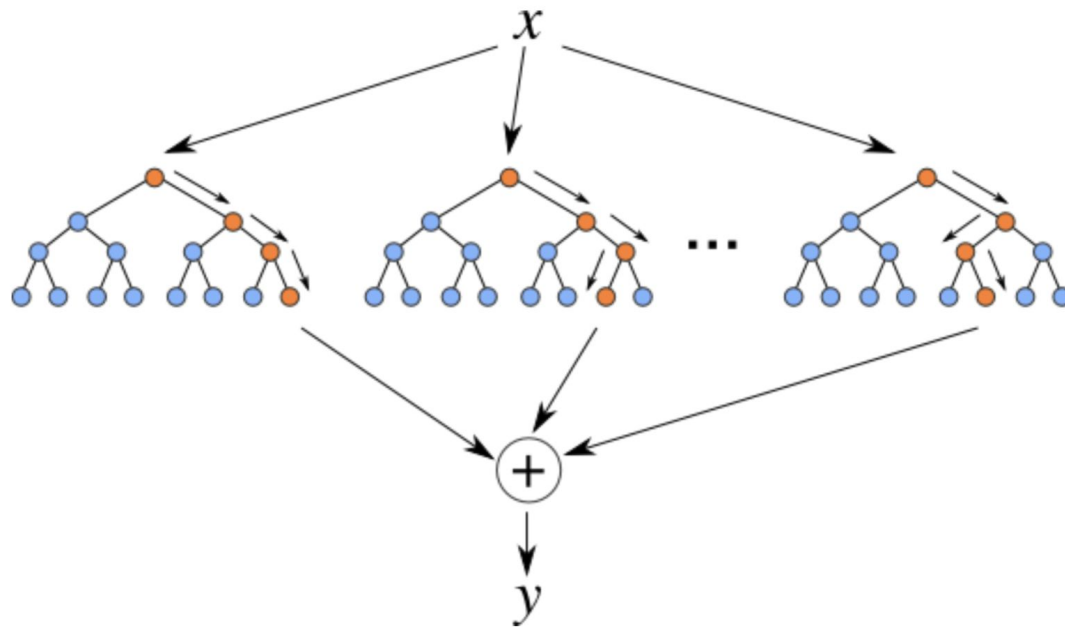
# RSM - Random Subspace Method



Same approach, but with features.

# Random Forest

Bagging + RSM = Random Forest





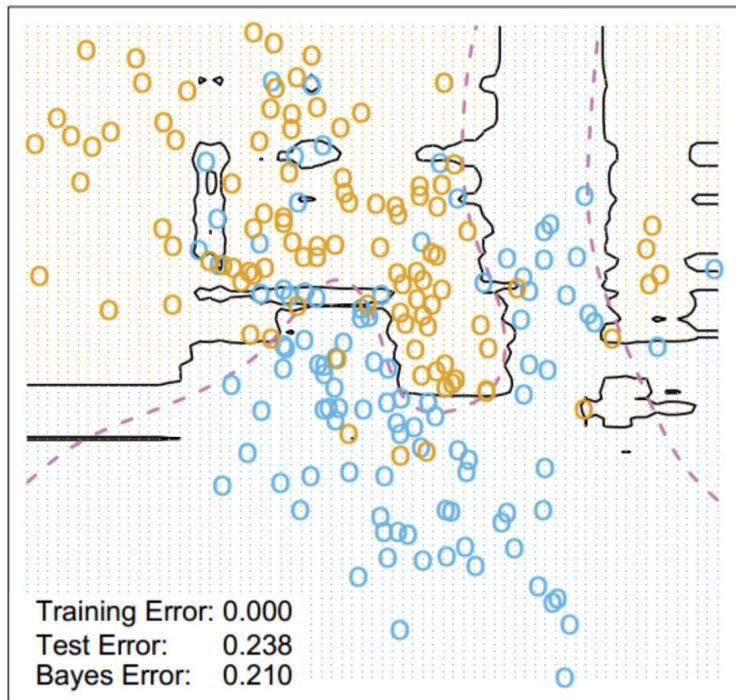
# Random Forest

- One of the greatest “universal” models.
- There are some modifications: Extremely Randomized Trees, Isolation Forest, etc.
- Allows to use train data for validation: OOB

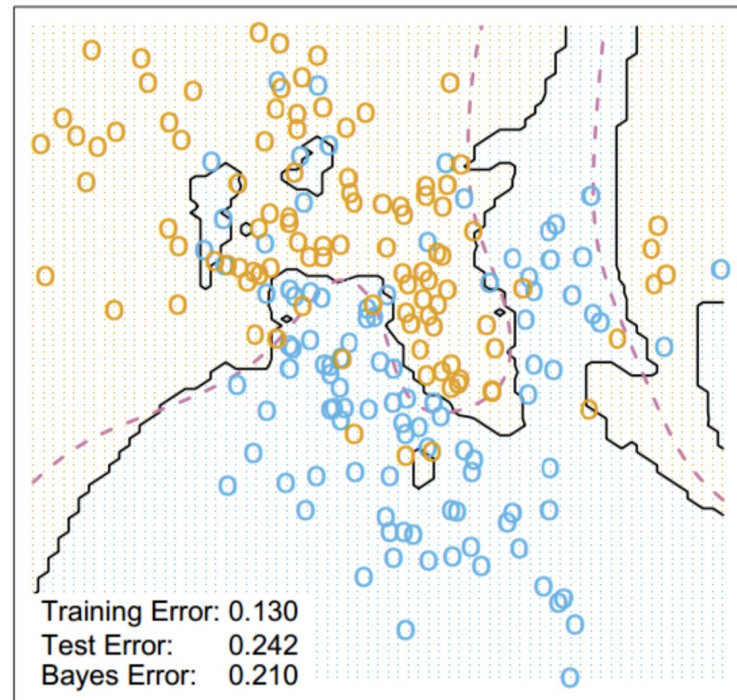
$$\text{OOB} = \sum_{i=1}^{\ell} L \left( y_i, \frac{1}{\sum_{n=1}^N [x_i \notin X_n]} \sum_{n=1}^N [x_i \notin X_n] b_n(x_i) \right)$$



## Random Forest Classifier



## 3-Nearest Neighbors





# Revise

1. Decision tree: intuition
2. Decision tree construction procedure
3. Information criteria
4. Pruning
5. Decision trees special highlights
  - Decision tree as linear model
  - Dealing with missing data
  - Categorical features
6. Bootstrap and Bagging
7. Random Forest

# Thanks for attention!

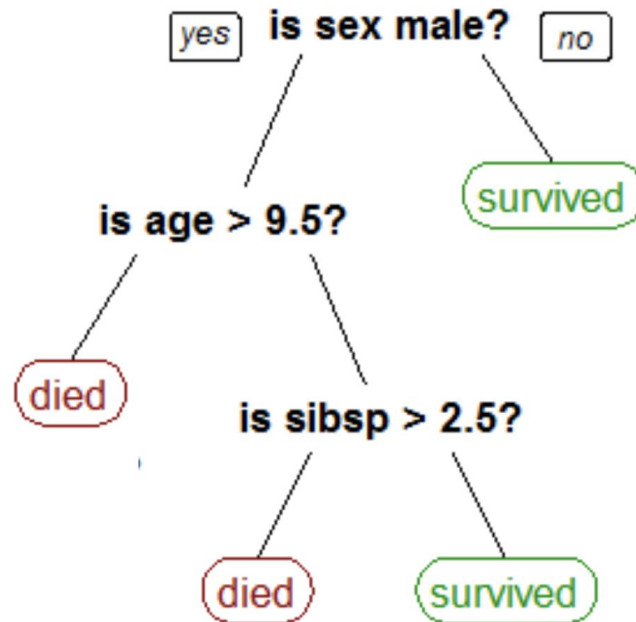
Questions?



# Decision tree



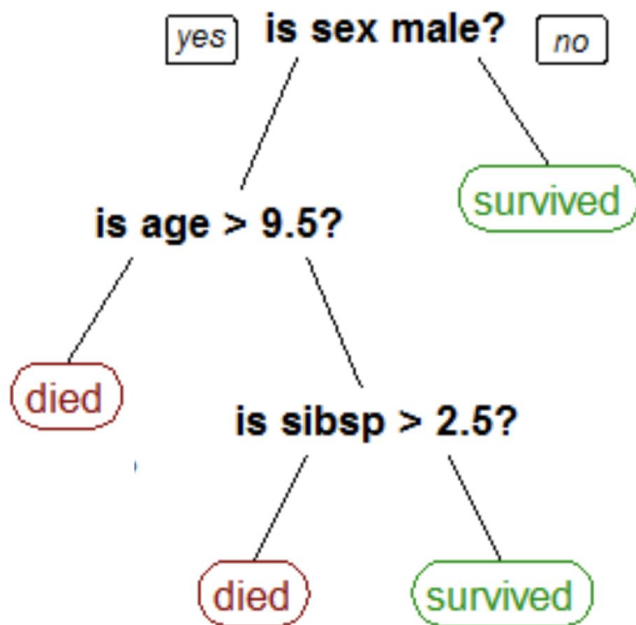
$x = (9, \text{ male}, 3)$



# Decision tree



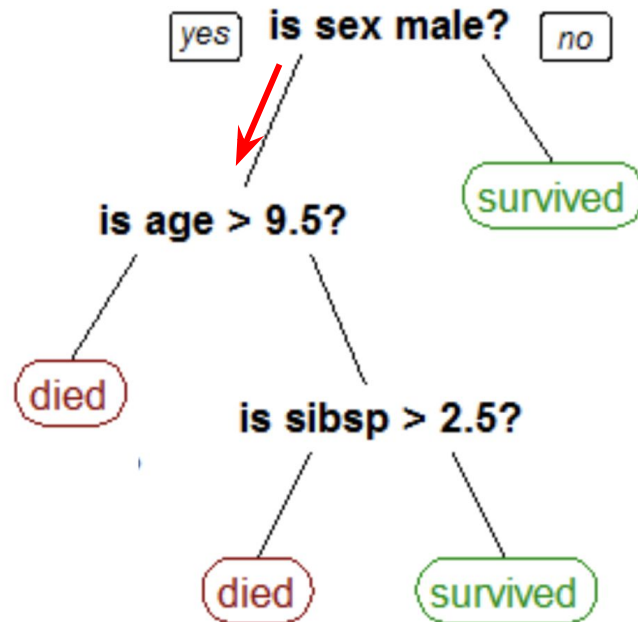
age      sex      sibsp  
 $x = (9, \text{male}, 3)$



# Decision tree



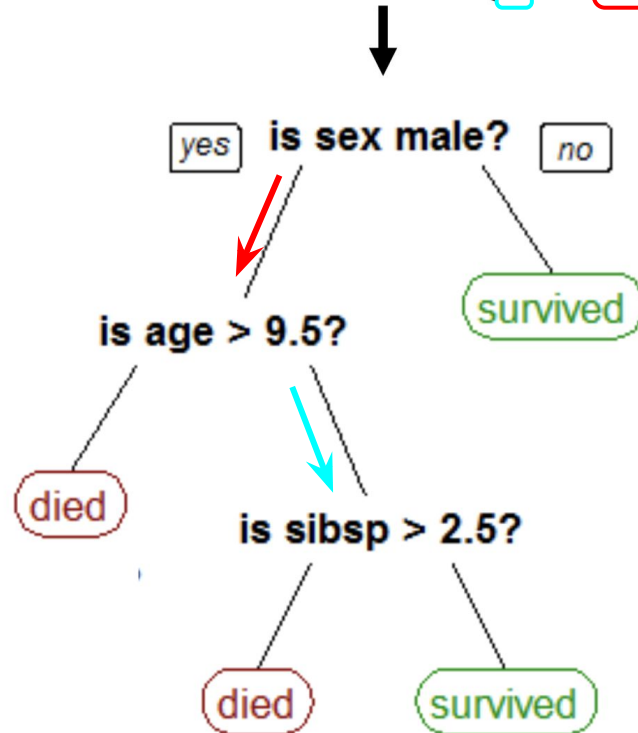
age      sex      sibsp  
 $x = (9, \text{male}, 3)$



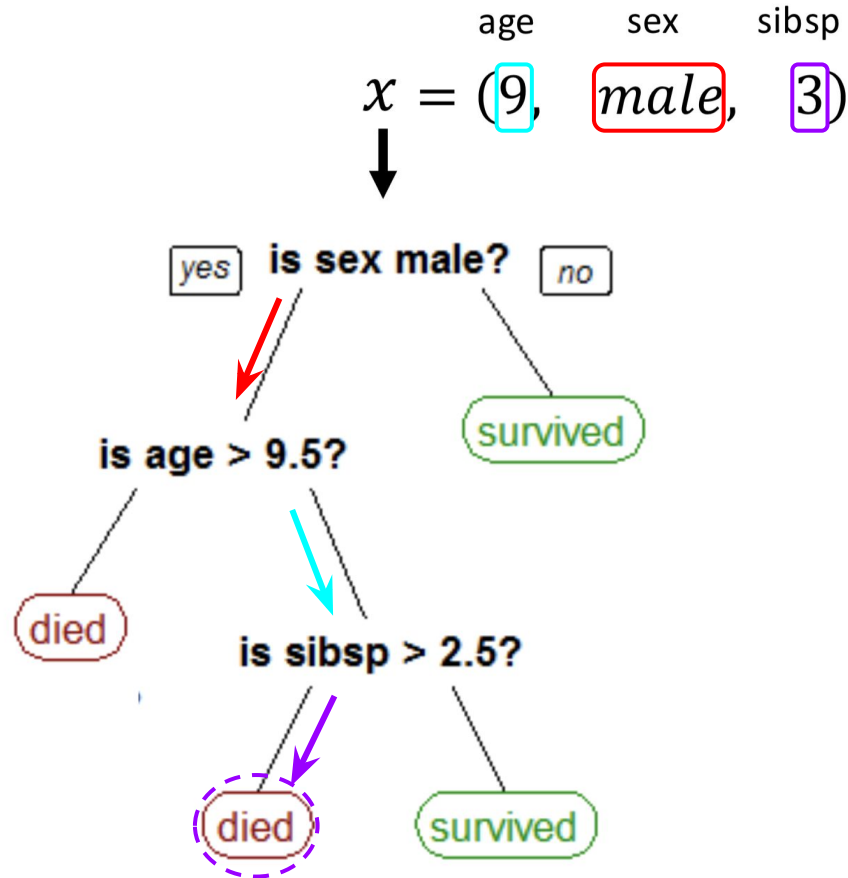
# Decision tree



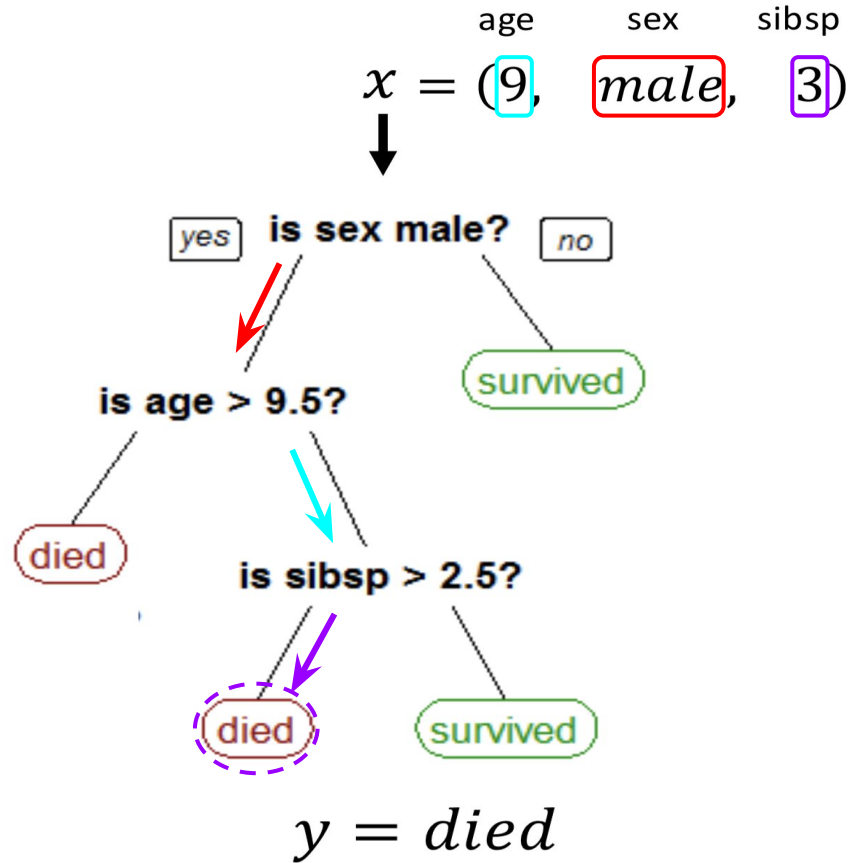
age      sex      sibsp  
 $x = (9, \text{male}, 3)$



# Decision tree

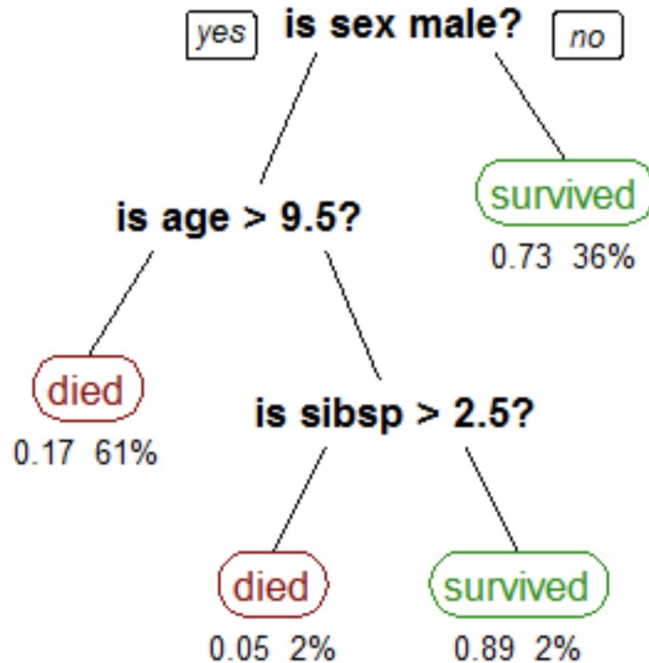


# Decision tree

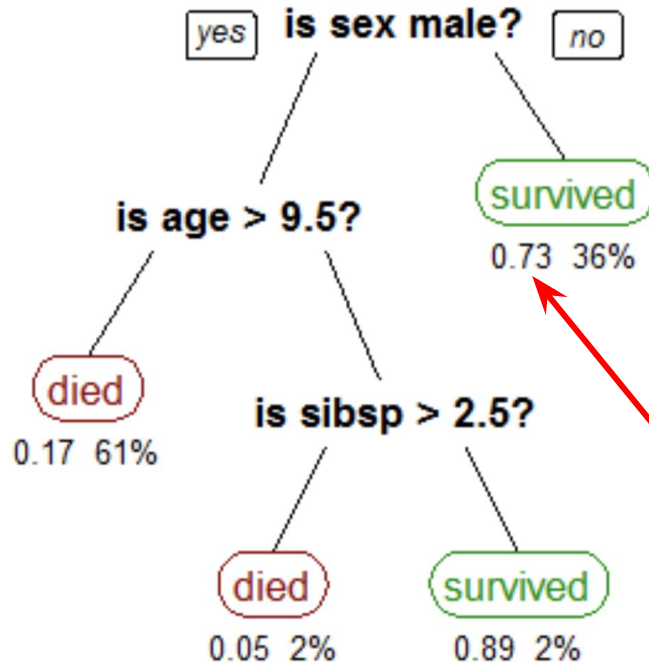




# Decision tree in classification

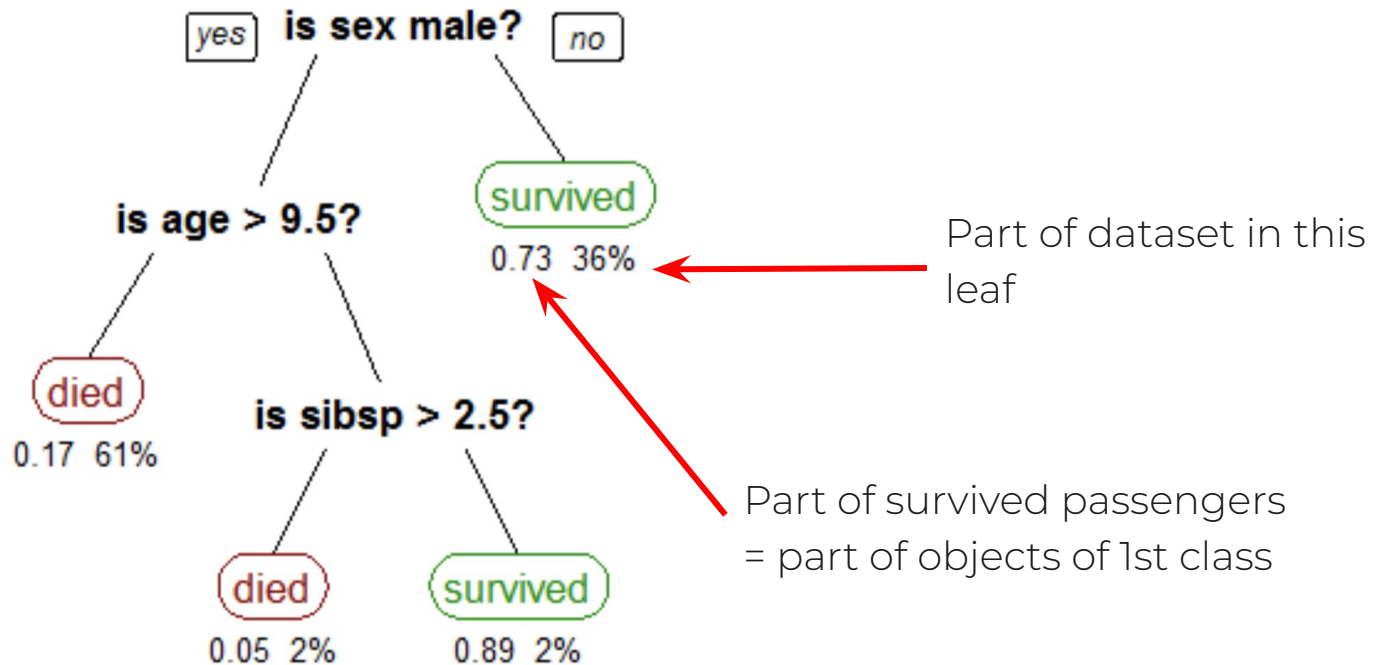


# Decision tree in classification

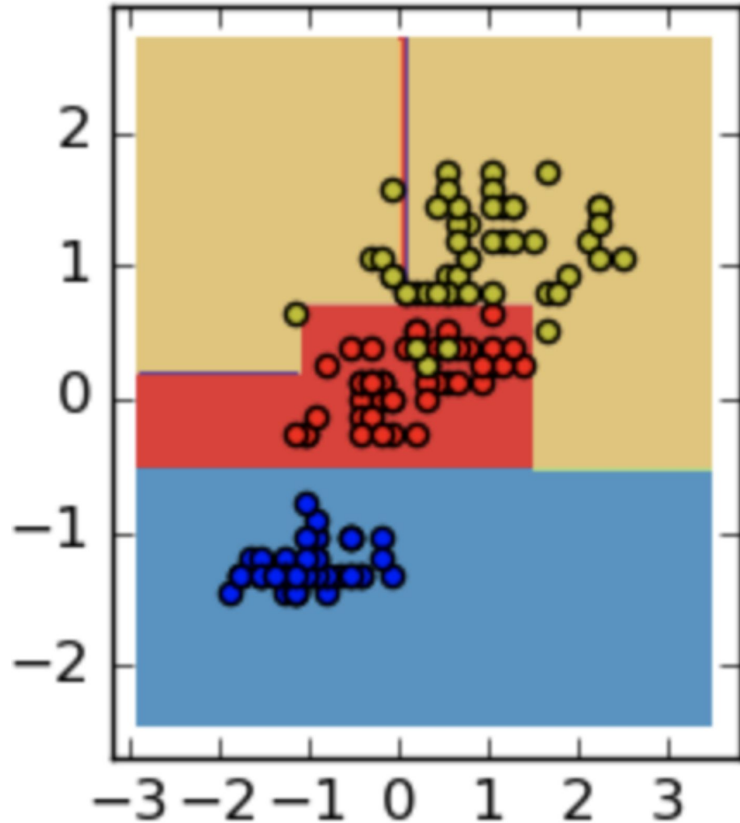


Part of survived passengers  
= part of objects of 1st class

# Decision tree in classification



# Decision tree in classification



Classification problem with 3 classes and 2 features.

# Pruning



- Pre-pruning:
  - Constrain the tree before construction.
- Post-pruning:
  - Simplify constructed tree.

Actually, it is the main trick in CART tree construction algorithm.

# Binarisation



Idea: instead selecting one threshold define several for one feature.

