Machine Learning
Lecture 5: Decision trees

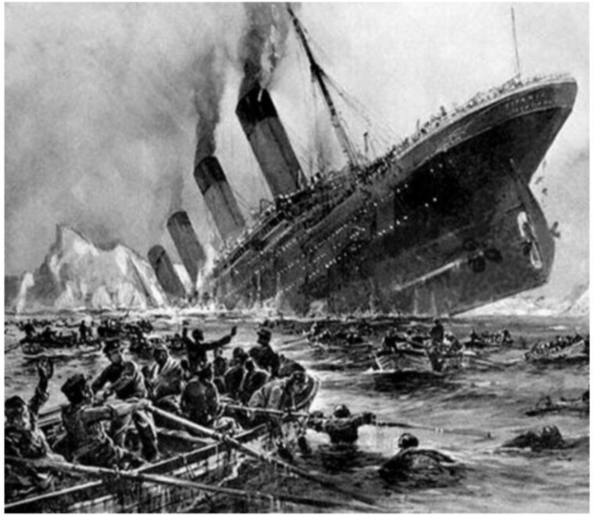
Harbour.Space University
February 2020

Radoslav Neychev

Outline

1. Decision tree definition
2. Decision trees in classification and regression
3. Constructing decision tree
4. Information criterions
5. Pruning
6. Boostrap
7. Random Forest

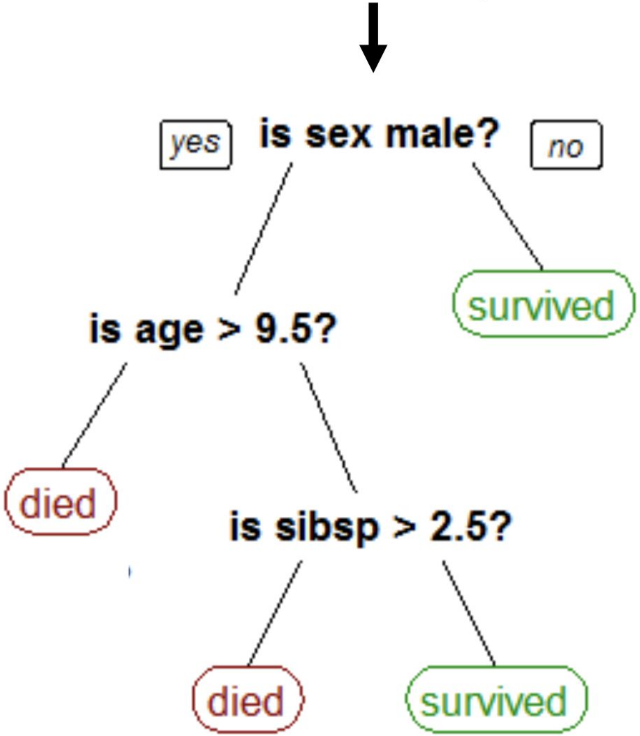
The dataset



- Titanic Dataset - the “Hello world” dataset in ML
- Target is binary: survived or not
- A lot of great tutorials with this dataset
  - E.g. challenge on Kaggle
- You will meet it in Lab 1

Decision tree

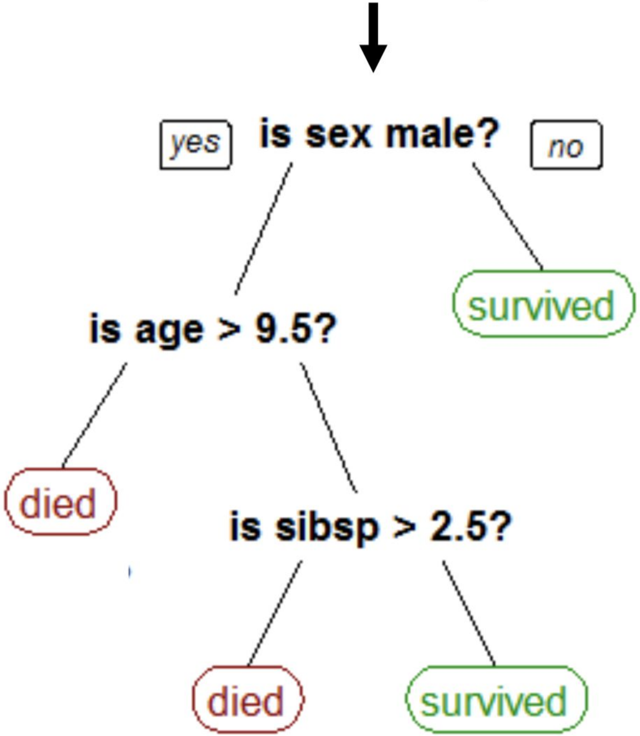
$$x = (9, \quad male, \quad 3)$$



Decision tree

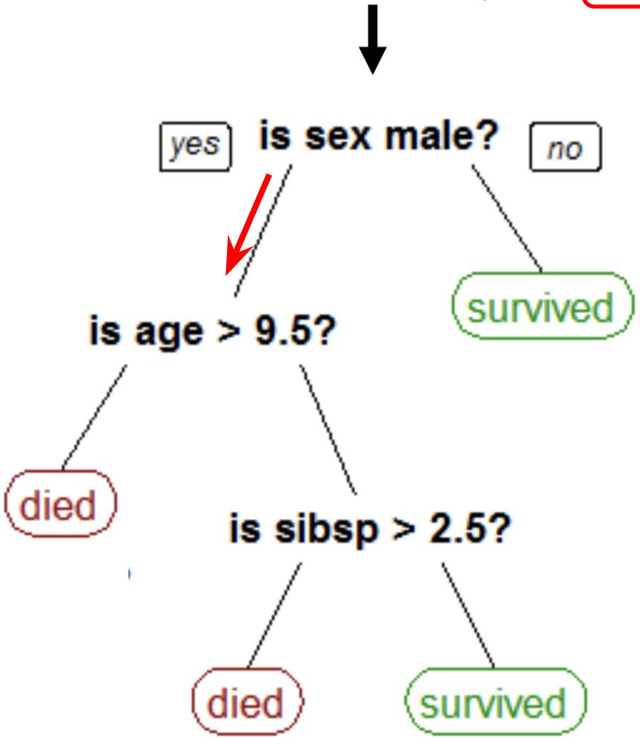
age sex sibsp

$$x = (9, \quad male, \quad 3)$$



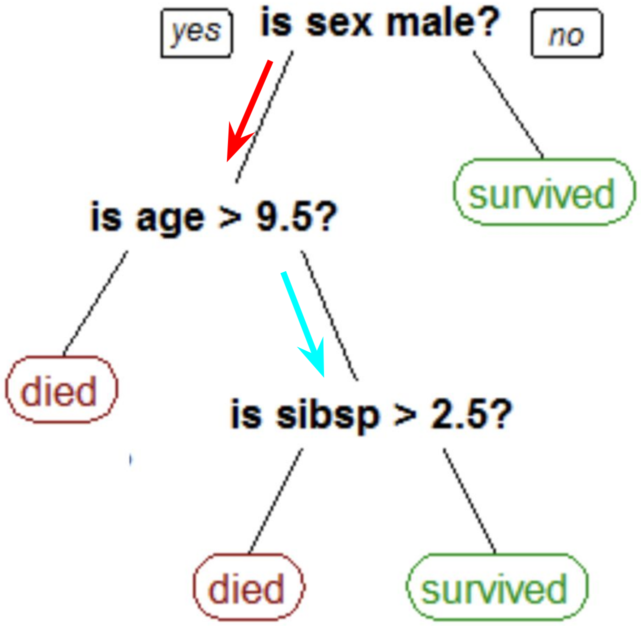
Decision tree

$$\begin{array}{cccc} & \text{age} & \text{sex} & \text{sibsp} \\ x = (& 9, & \boxed{male}, & 3) \end{array}$$

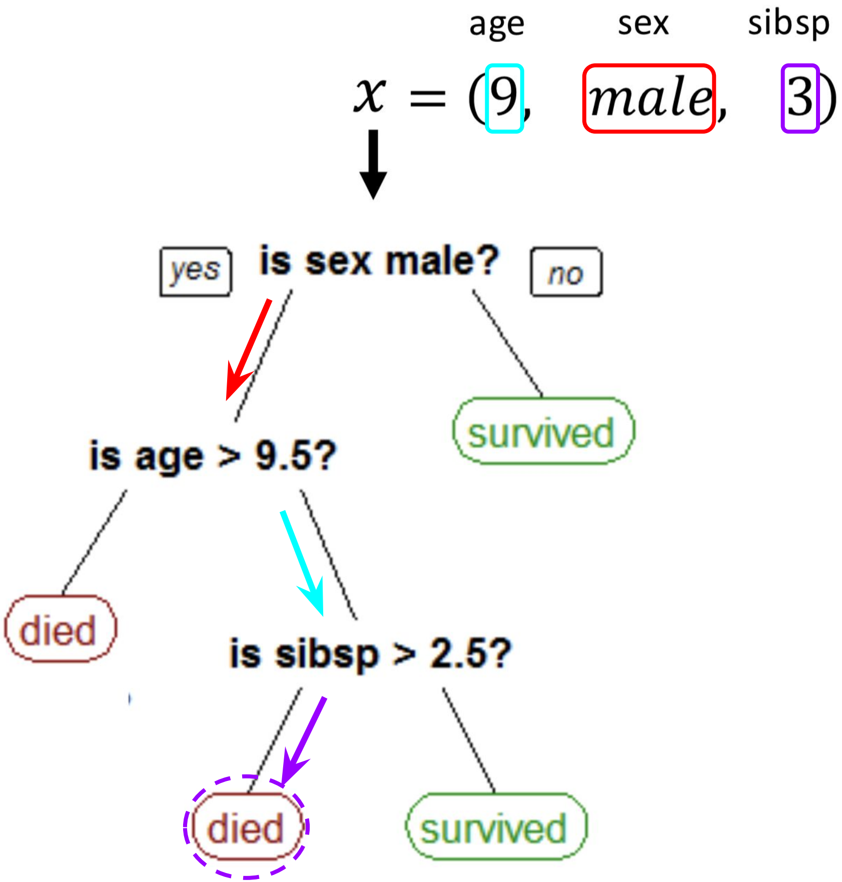


Decision tree

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

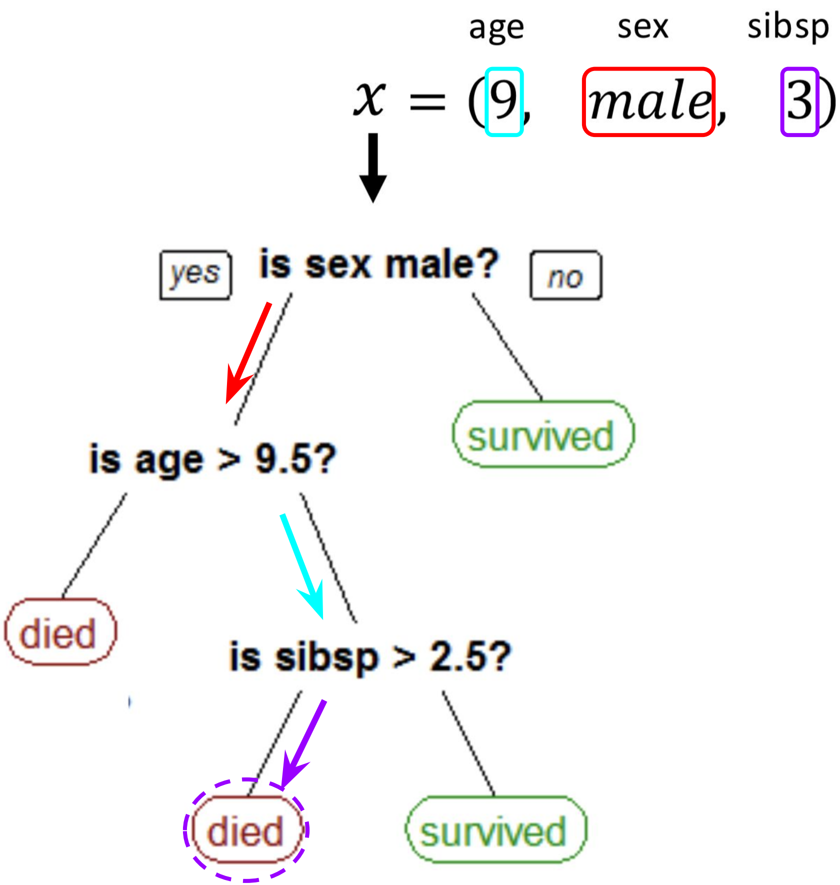


Decision tree



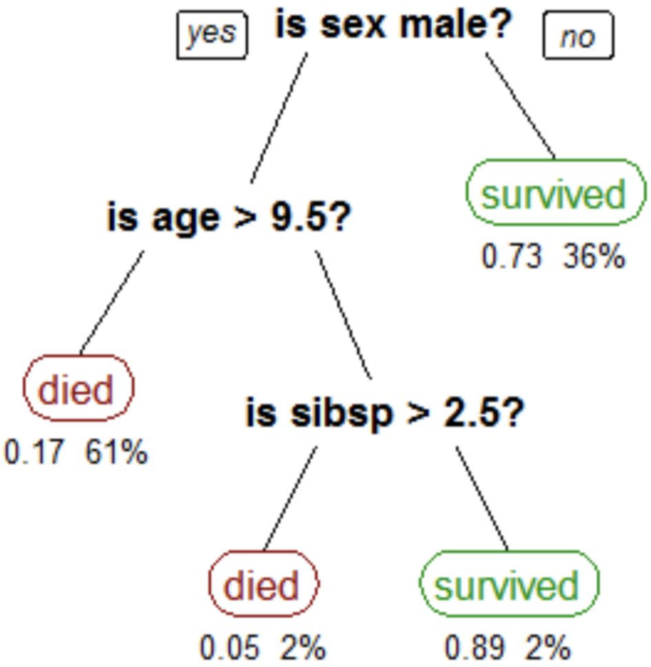
Decision tree

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

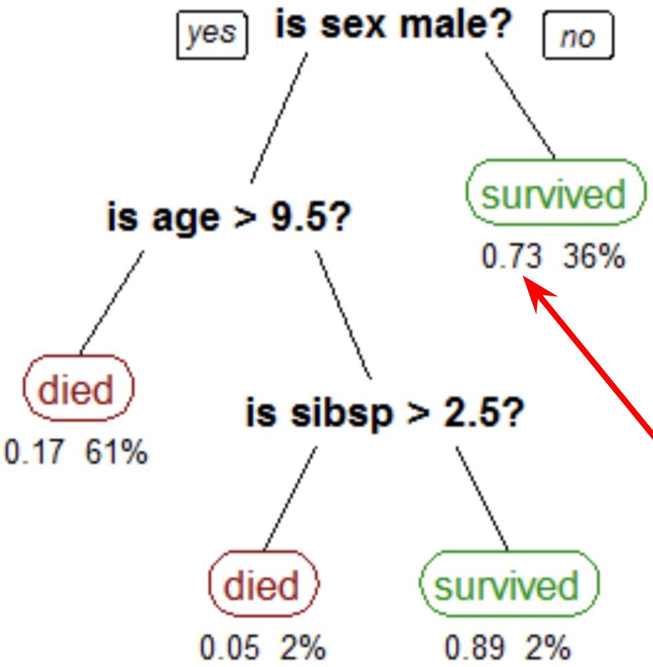


$$y = died$$

Decision tree in classification

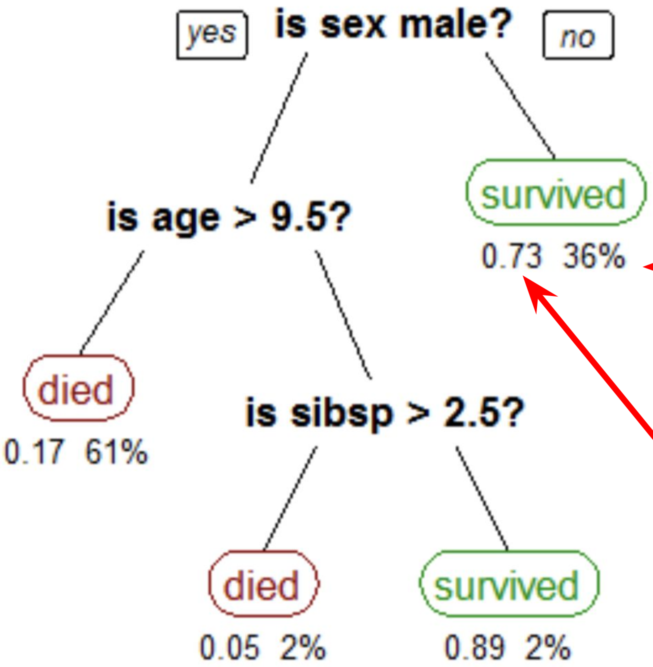


Decision tree in classification



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

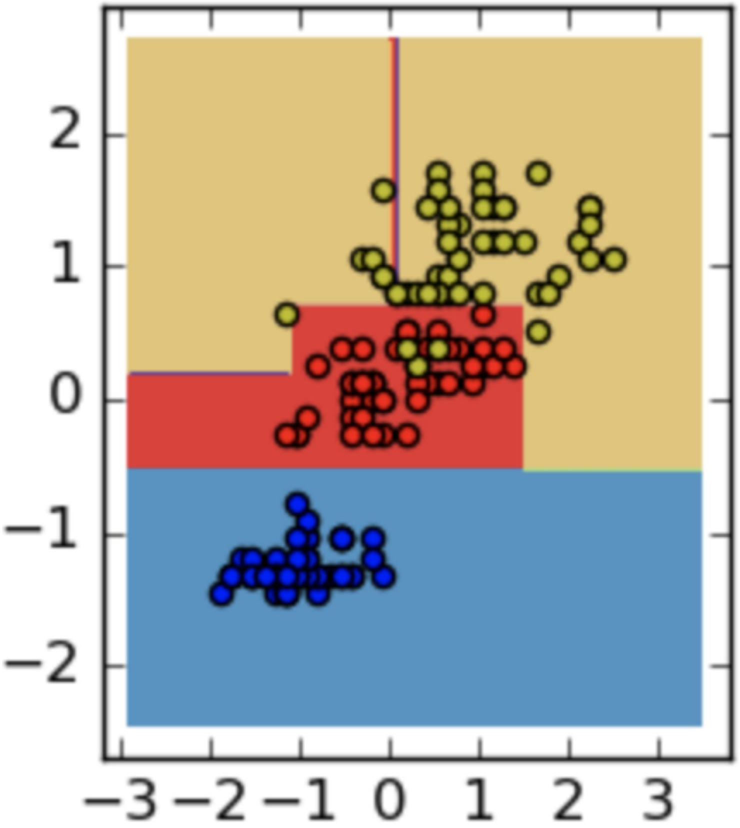
Decision tree in classification



Part of dataset in this leaf

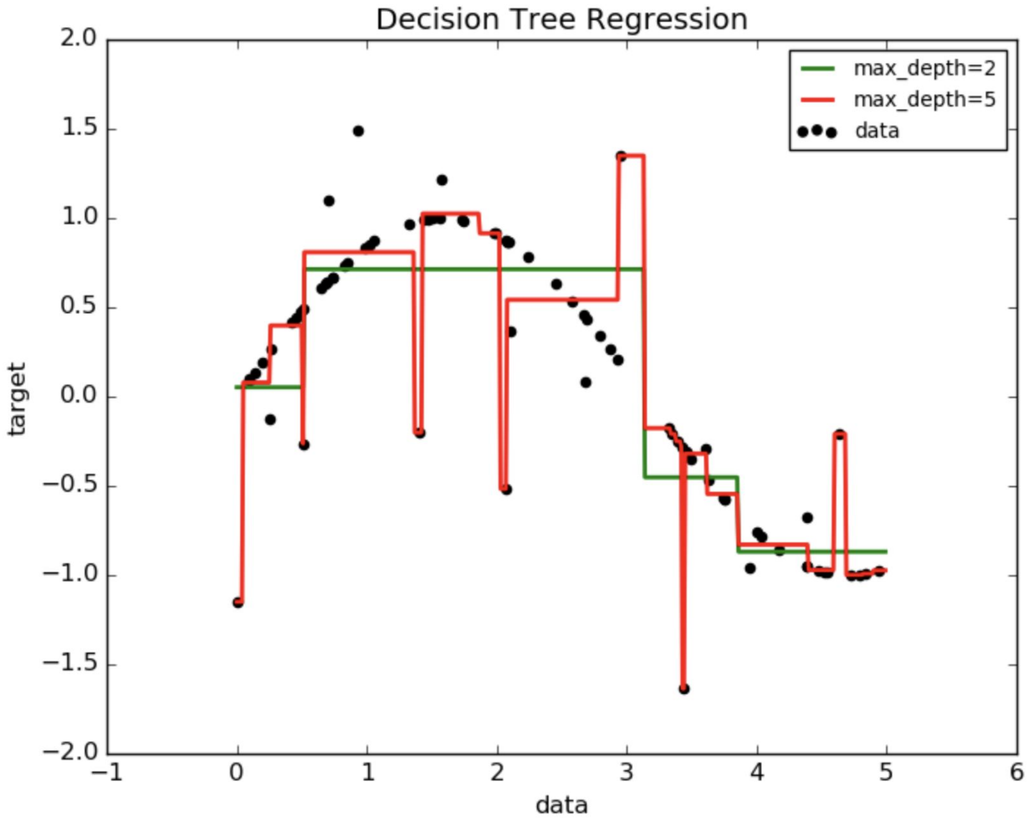
Part of survived passengers
= part of objects of 1st class

Decision tree in classification



Classification problem with 3 classes and 2 features.

Decision tree in regression



Green - decision tree of depth 2

Red - decision tree of depth 5

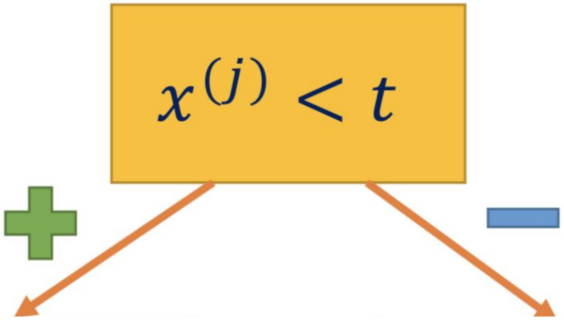
Every leaf corresponds to some constant.

Constructing decision trees

$$\boxed{x^{(j)} < t}$$

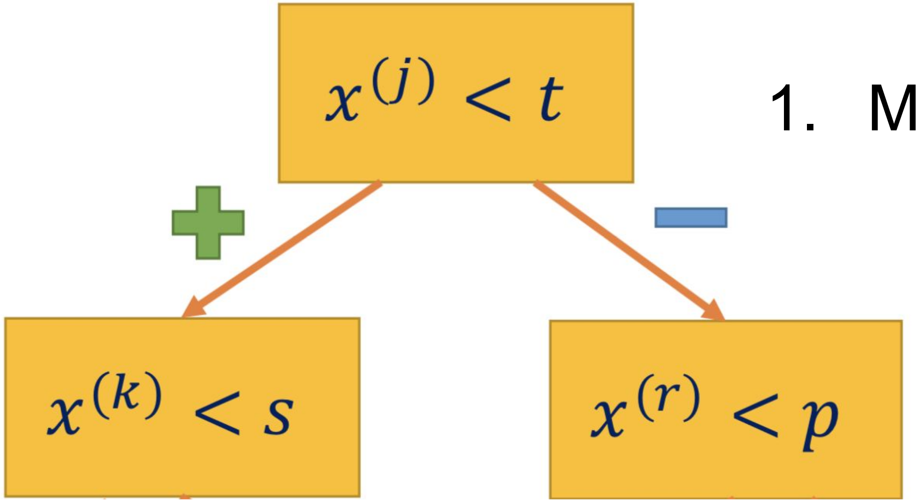
1. Make a split

Constructing decision trees



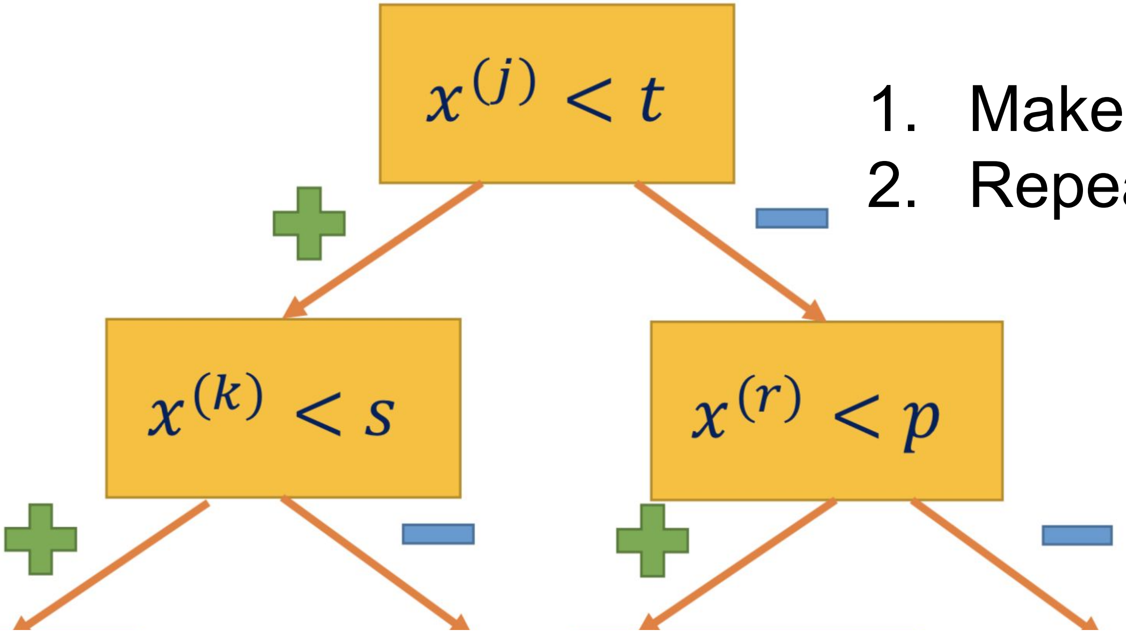
1. Make a split

Constructing decision trees



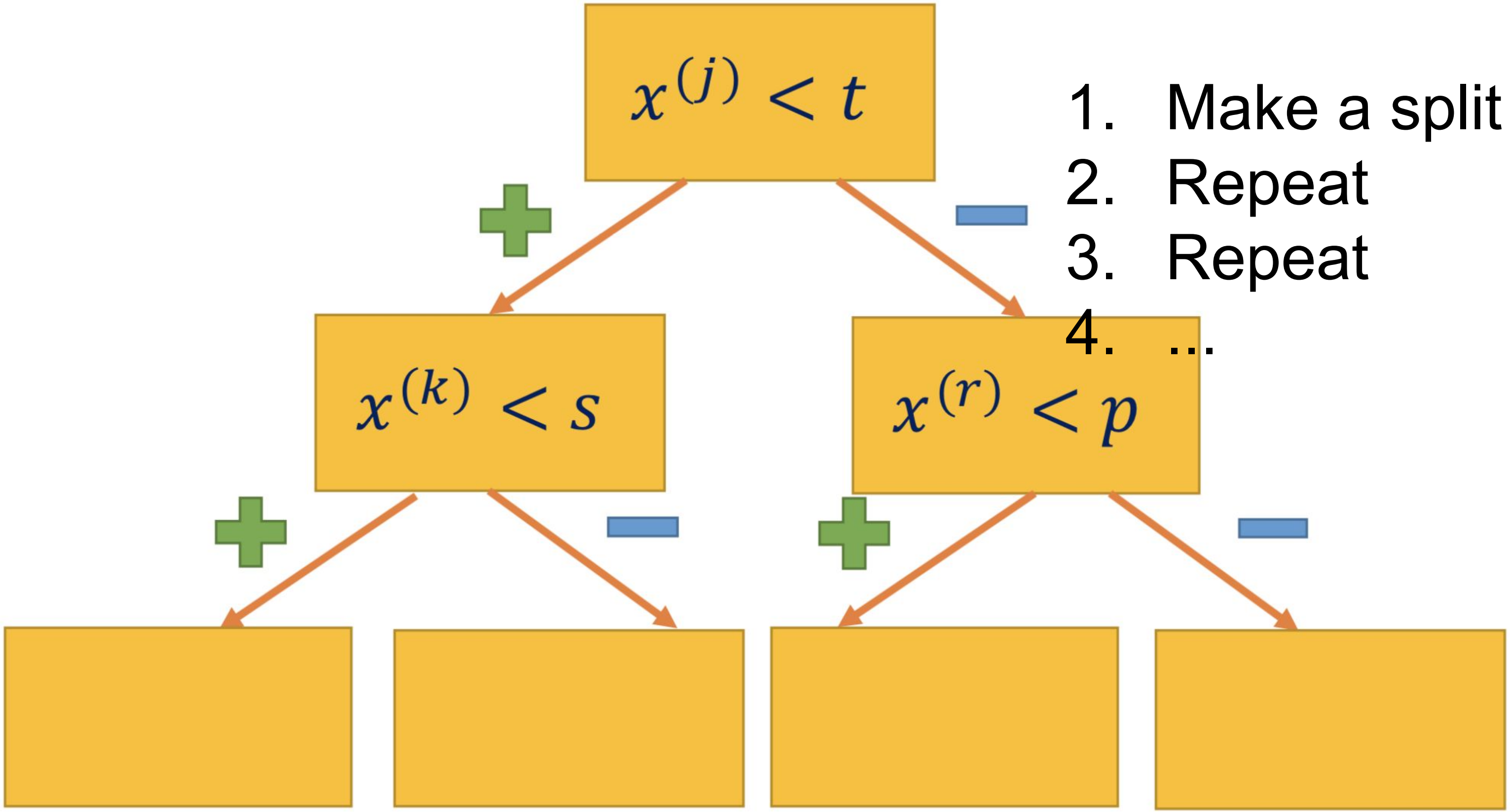
1. Make a split

Constructing decision trees



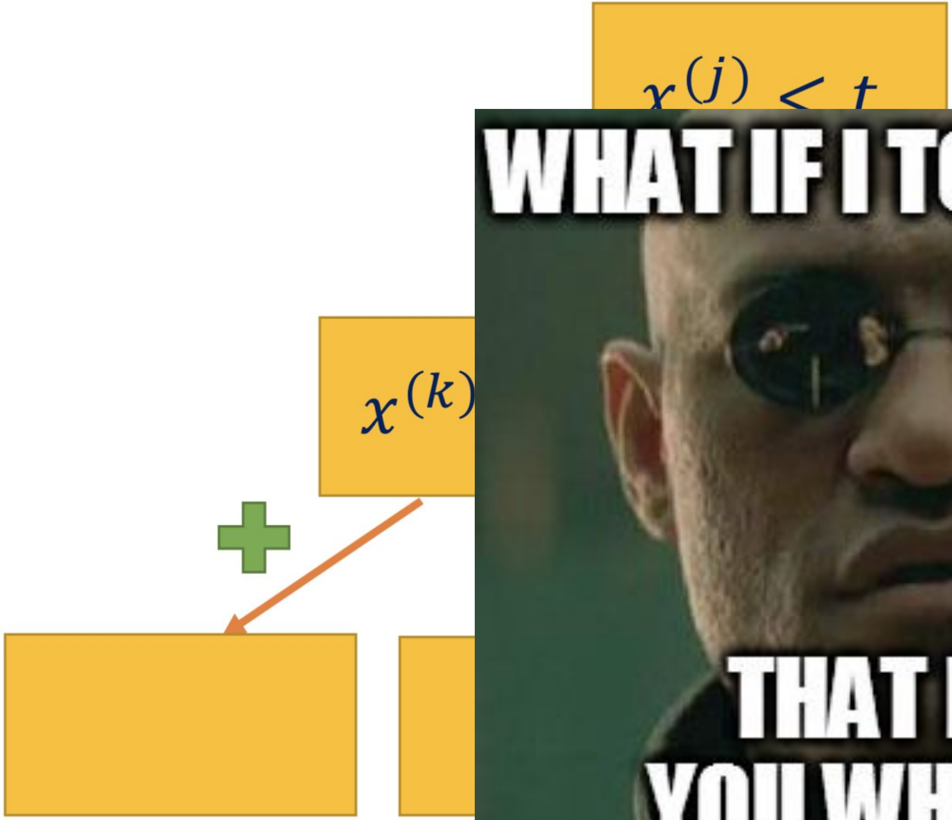
1. Make a split
2. Repeat

Constructing decision trees

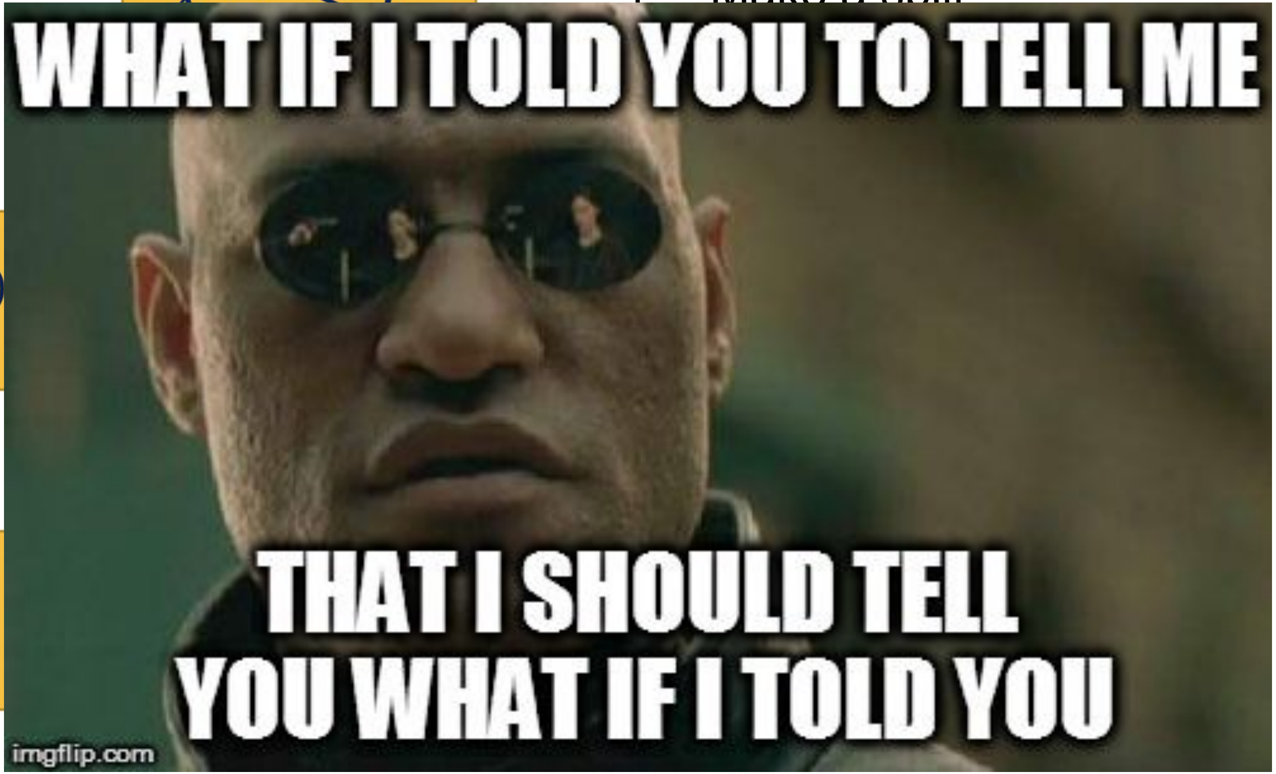


1. Make a split
2. Repeat
3. Repeat
4. ...

Constructing decision trees



1. Make a split

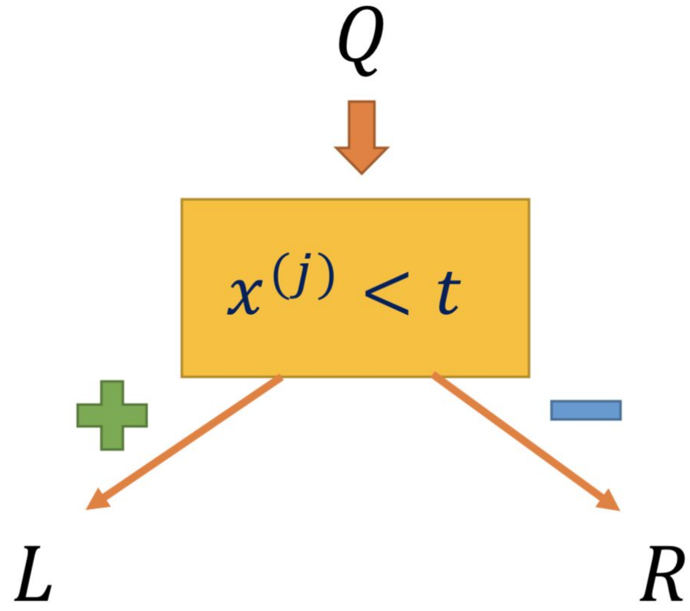


How to split data properly?

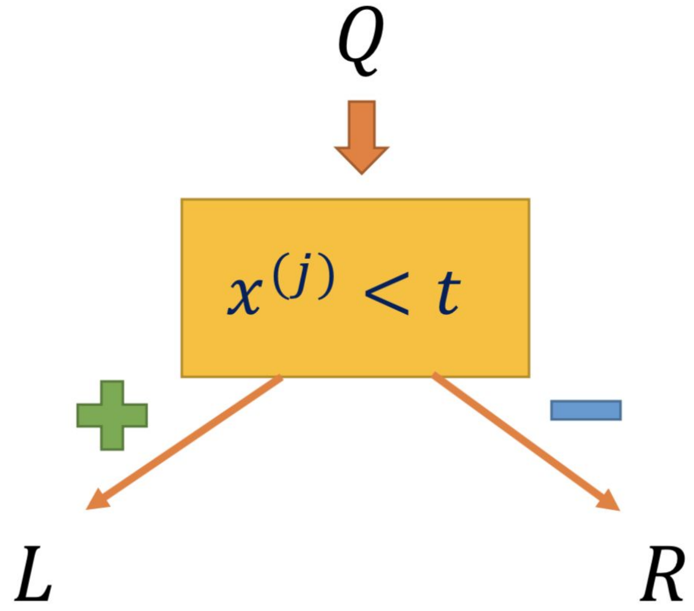
Q

$$x^{(j)} < t$$

How to split data properly?

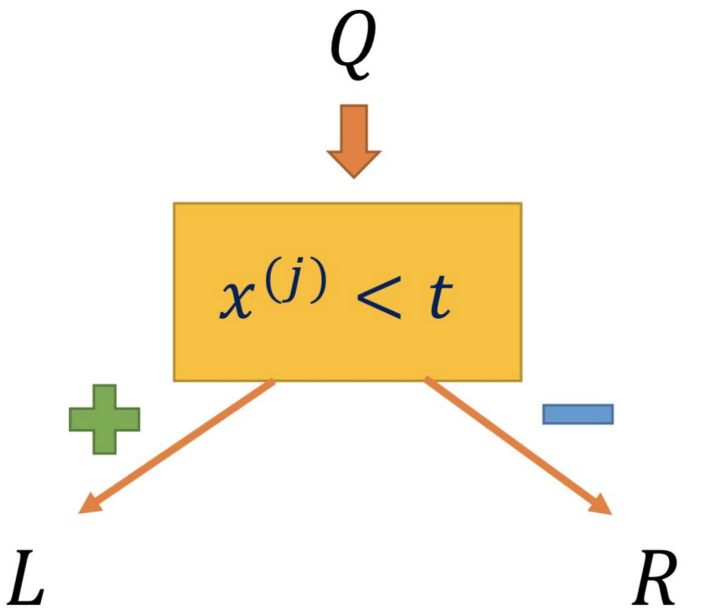


How to split data properly?



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

How to split data properly?



What is H()?

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

Information criteria

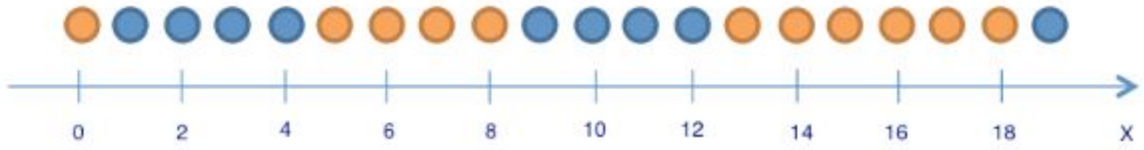
H(R) is measure of “heterogeneity” of our data.
Consider binary classification problem:

1. Misclassification criteria: $H(R) = 1 - \max\{p_0, p_1\}$

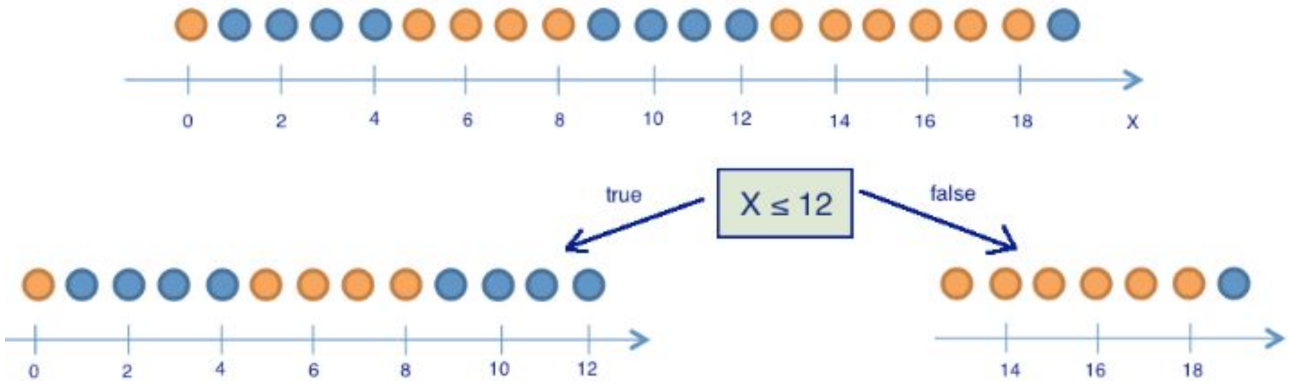
2. Entropy criteria: $H(R) = -p_0 \log_2 p_0 - p_1 \log_2 p_1$

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

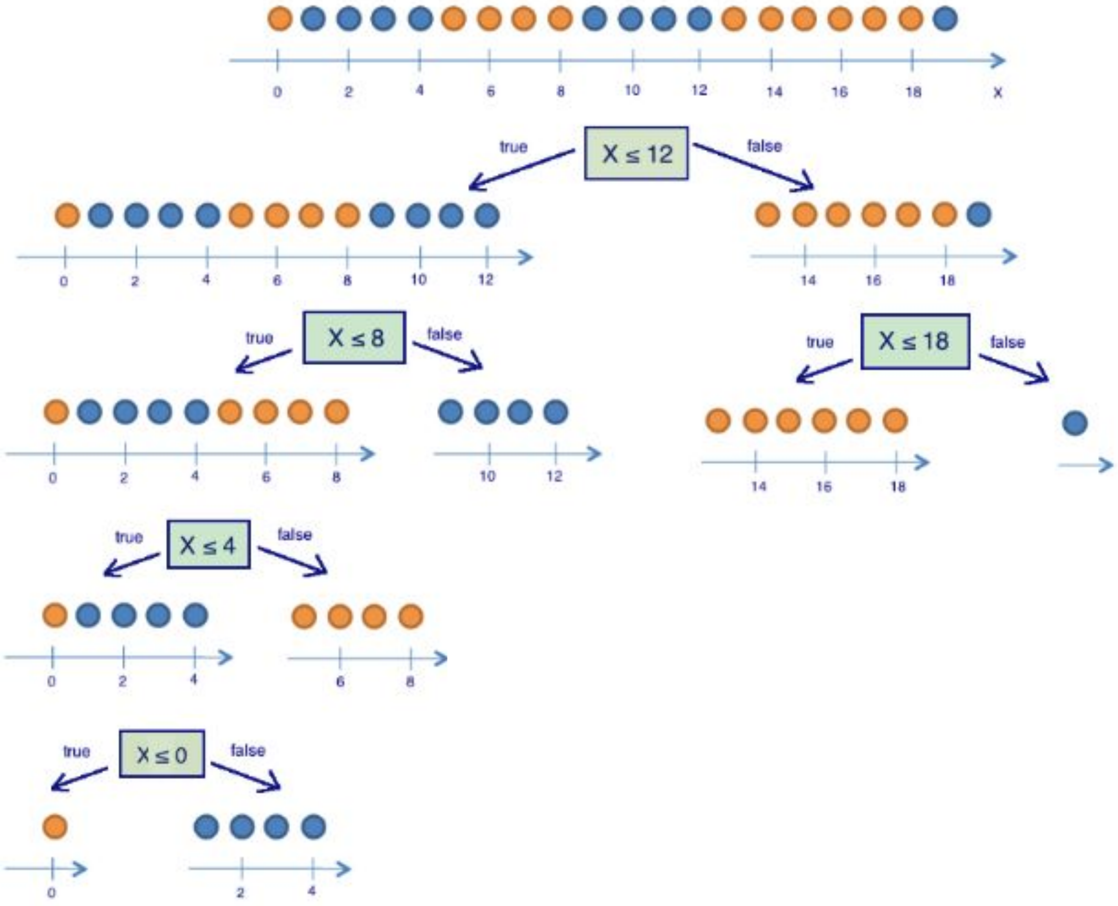
H(R) is measure of “heterogeneity” of our data.
Consider binary classification problem:



H(R) is measure of “heterogeneity” of our data.
Consider binary classification problem:

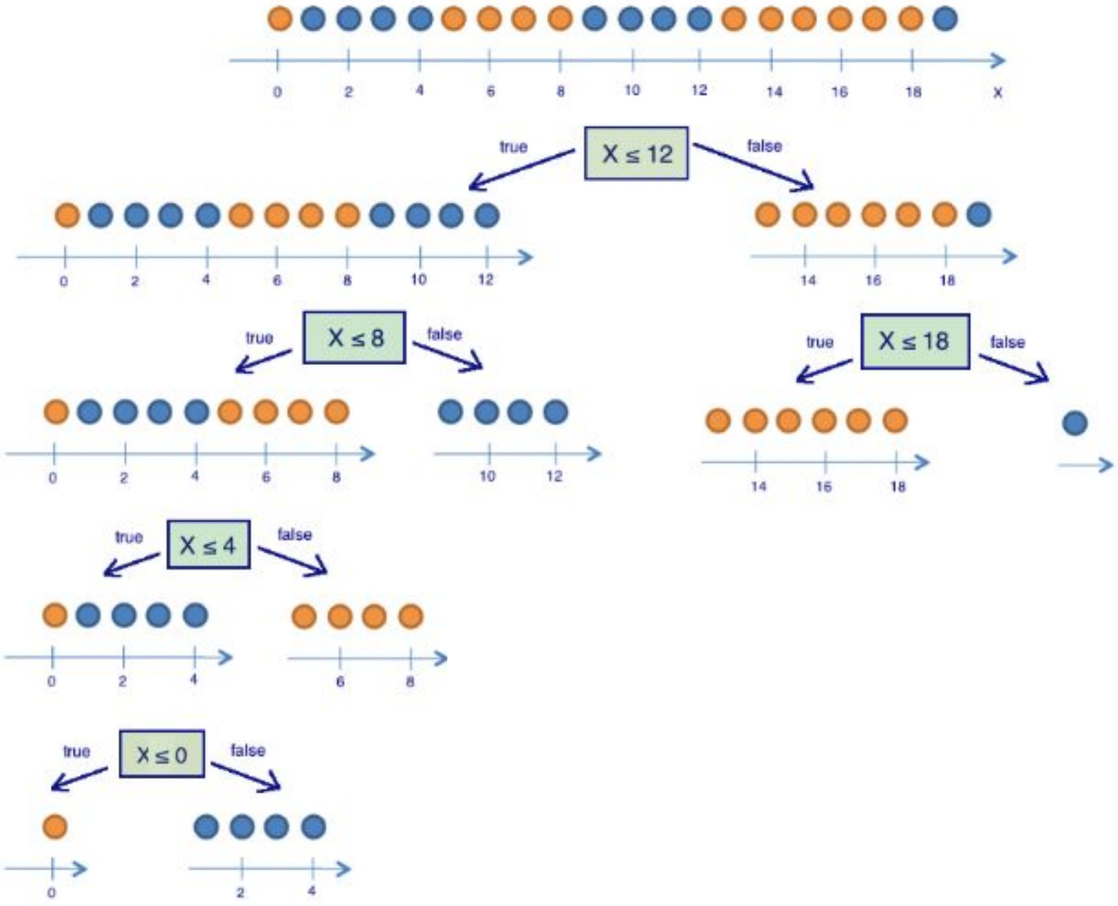


Information criteria: Entropy



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

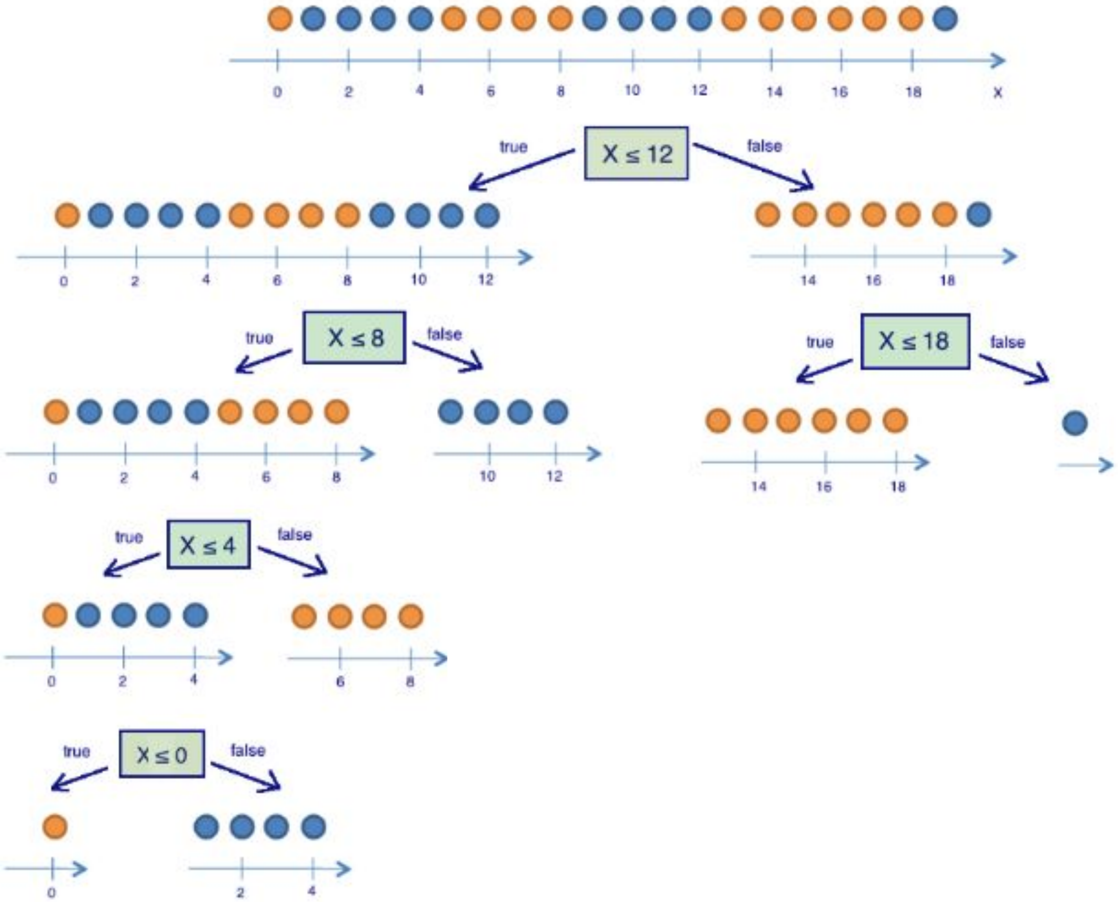
Information criteria: Entropy



$$S = -\sum_k p_k \log_2 p_k$$

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

Information criteria: Entropy



$$S = -\sum_k p_k \log_2 p_k$$

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

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

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

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

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

Information criteria

H(R) is measure of “heterogeneity” of our data.
Consider multiclass classification problem:

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

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

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

H(R) is measure of “heterogeneity” of our data.
Consider regression problem:

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

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?

Information criteria

H(R) is measure of “heterogeneity” of our data.
Consider regression problem:

1. Mean squared error

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

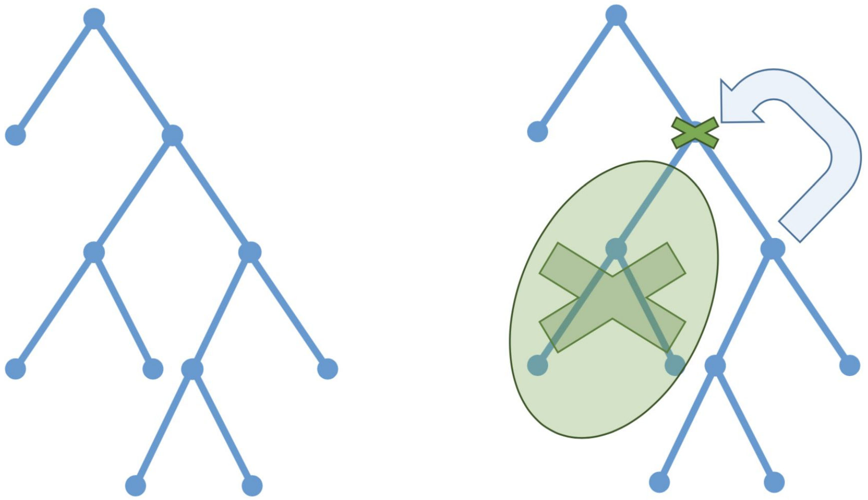
$$c^* = \frac{1}{|R|} \sum_{y_i \in R} y_i$$

Pruning

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

Pruning

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



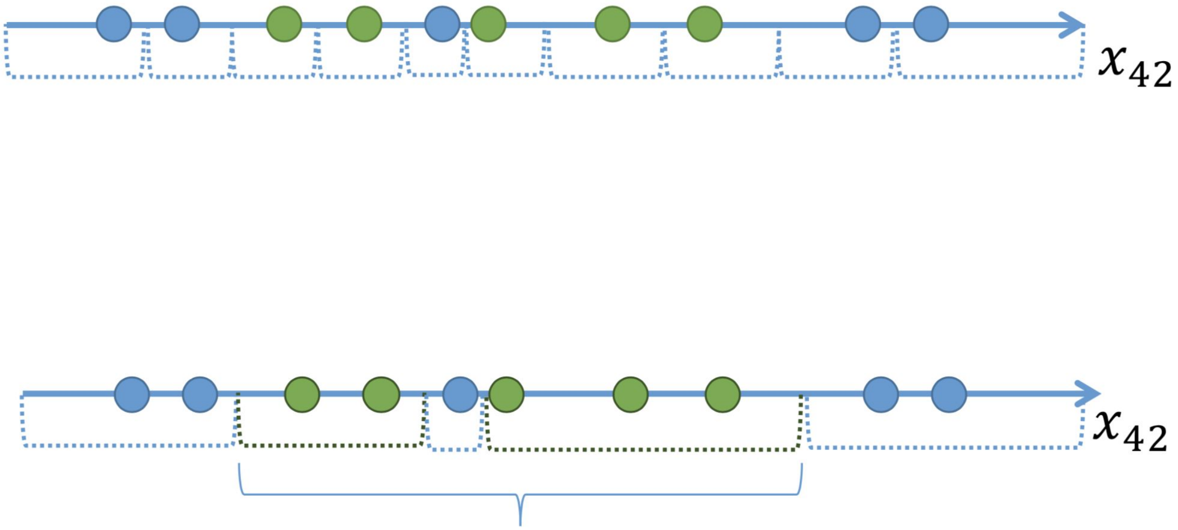
Pruning

- 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.



How the trees are actually constructed

- ID-3
- C4.5
- C5.0
- CART
- etc.

Bootstrap

Consider dataset X containing N objects.

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

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

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:

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

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:

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

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

Bootstrap

Consider the errors unbiased and uncorrelated:

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

The final model averages all predictions:

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

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

Error decreased by N times!

Bootstrap

Consider the errors ~~unbiased and uncorrelated~~:

Because this is a lie

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

The final model averages all predictions:

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

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

Error decreased by N times!

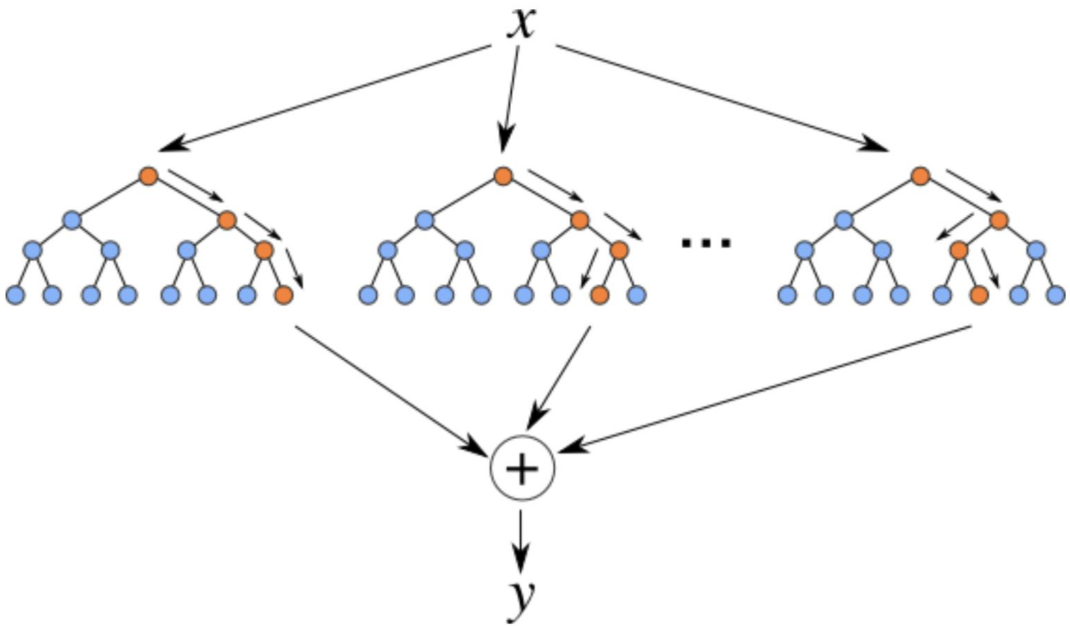
Bagging = Bootstrap aggregating

Decreases the variance if the basic algorithms are not correlated.

RSM - Random Subspace Method

Same approach, but with features.

Bagging + RSM = Random Forest



Random Forest

- One of the greatest “universal” models.

Random Forest

- One of the greatest “universal” models.
- There are some modifications: Extremely Randomized Trees, Isolation Forest, etc.
-

Random Forest

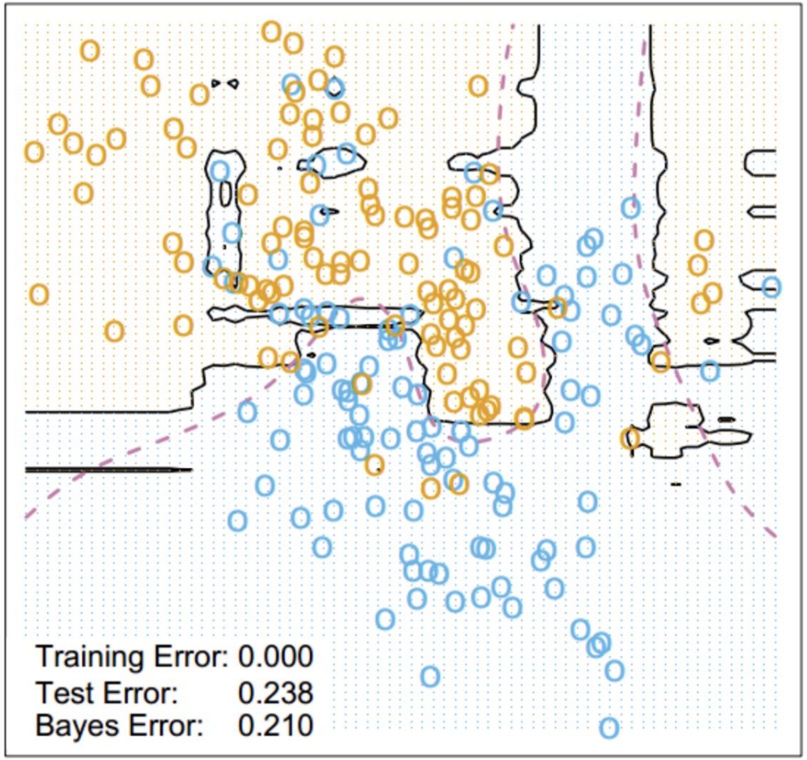
- One of the greatest “universal” models.
- There are some modifications: Extremely Randomized Trees, Isolation Forest, etc.
- Allows to use train data for validation: OOB

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

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

Random Forest Classifier



3−Nearest Neighbors