Machine Learning
Lecture 4:
Decision trees and Ensembles

Young && Yandex

Radoslav Neychev

girafe ai

Outline

01 Decision tree:
intuition

02 Decision tree construction procedure.

03 Information criteria

04 Pruning

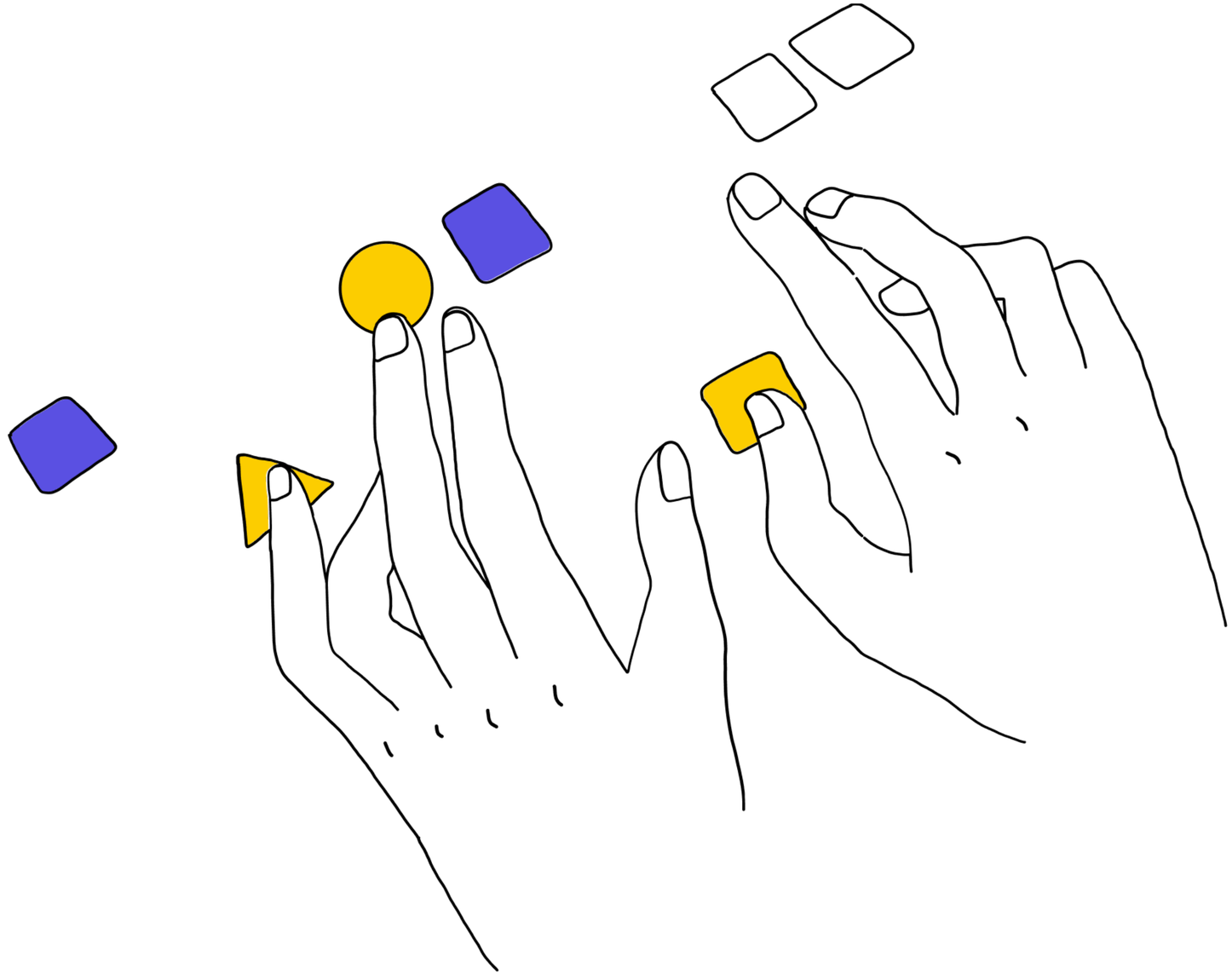
05 Decision trees special highlights

- Decision tree as linear model
- Dealing with missing data
- Categorical features

06 Bootstrap and Bagging

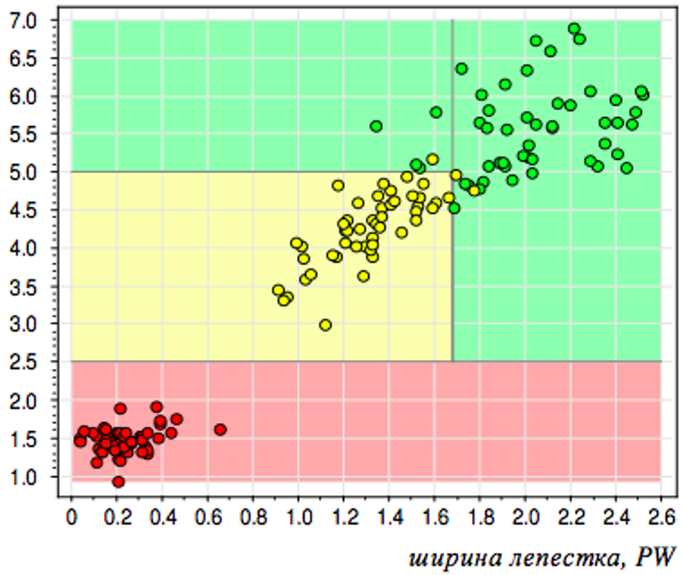
07 Random Forest

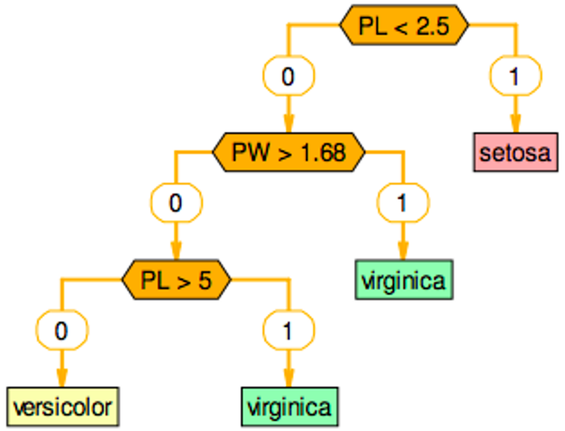
Decision tree: intuition



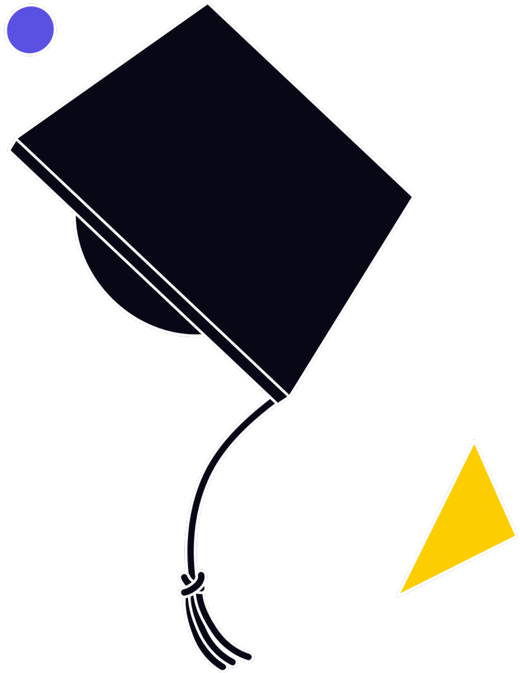
01

Decision tree for Iris data set

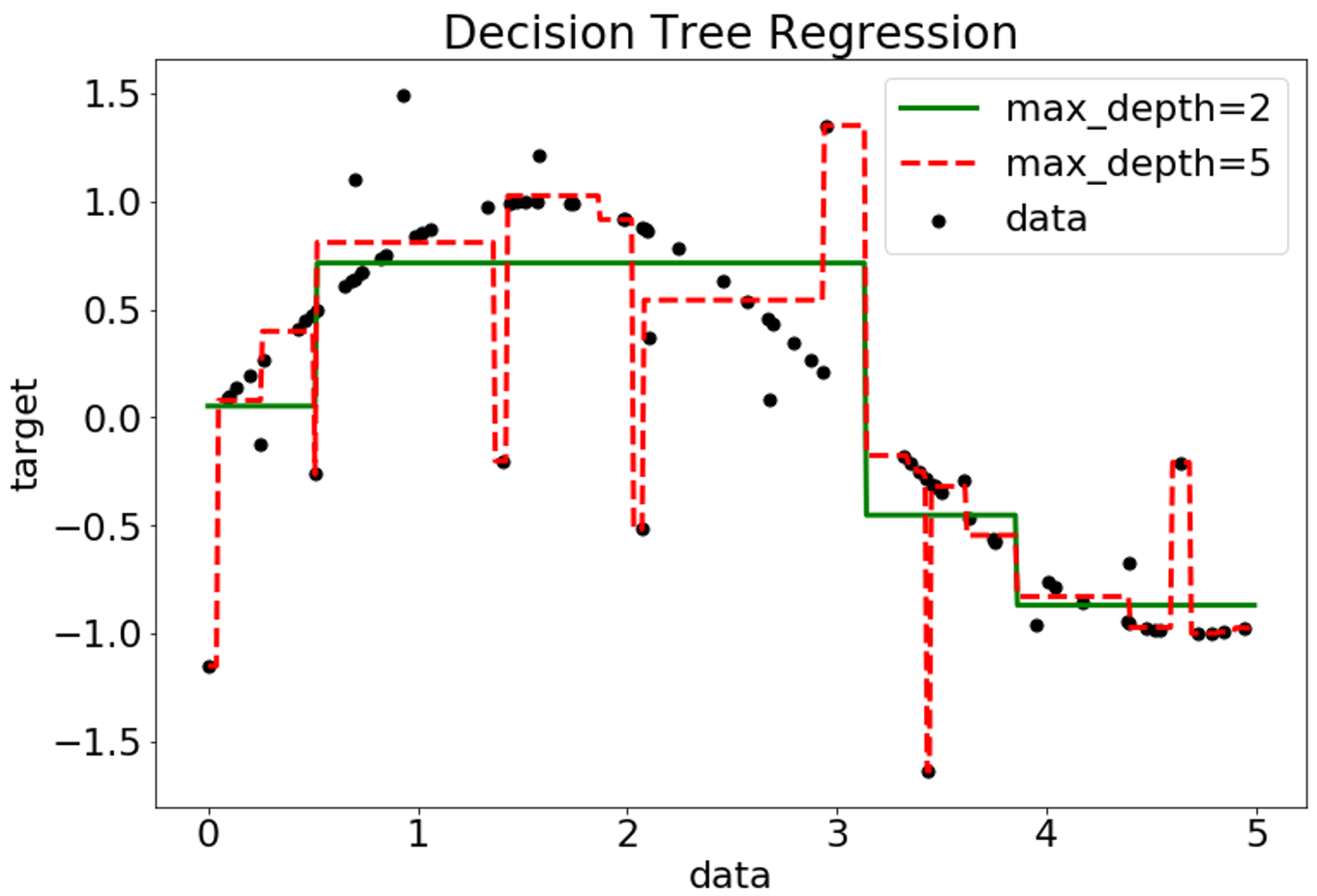




setosa	$r_1(x) = [PL \leqslant 2.5]$
virginica	$r_2(x) = [PL > 2.5] \wedge [PW > 1.68]$
virginica	$r_3(x) = [PL > 5] \wedge [PW \leqslant 1.68]$
versicolor	$r_4(x) = [PL > 2.5] \wedge [PL \leqslant 5] \wedge [PW < 1.68]$



Decision tree in regression



Green — decision tree of depth 2

Red — decision tree of depth 5

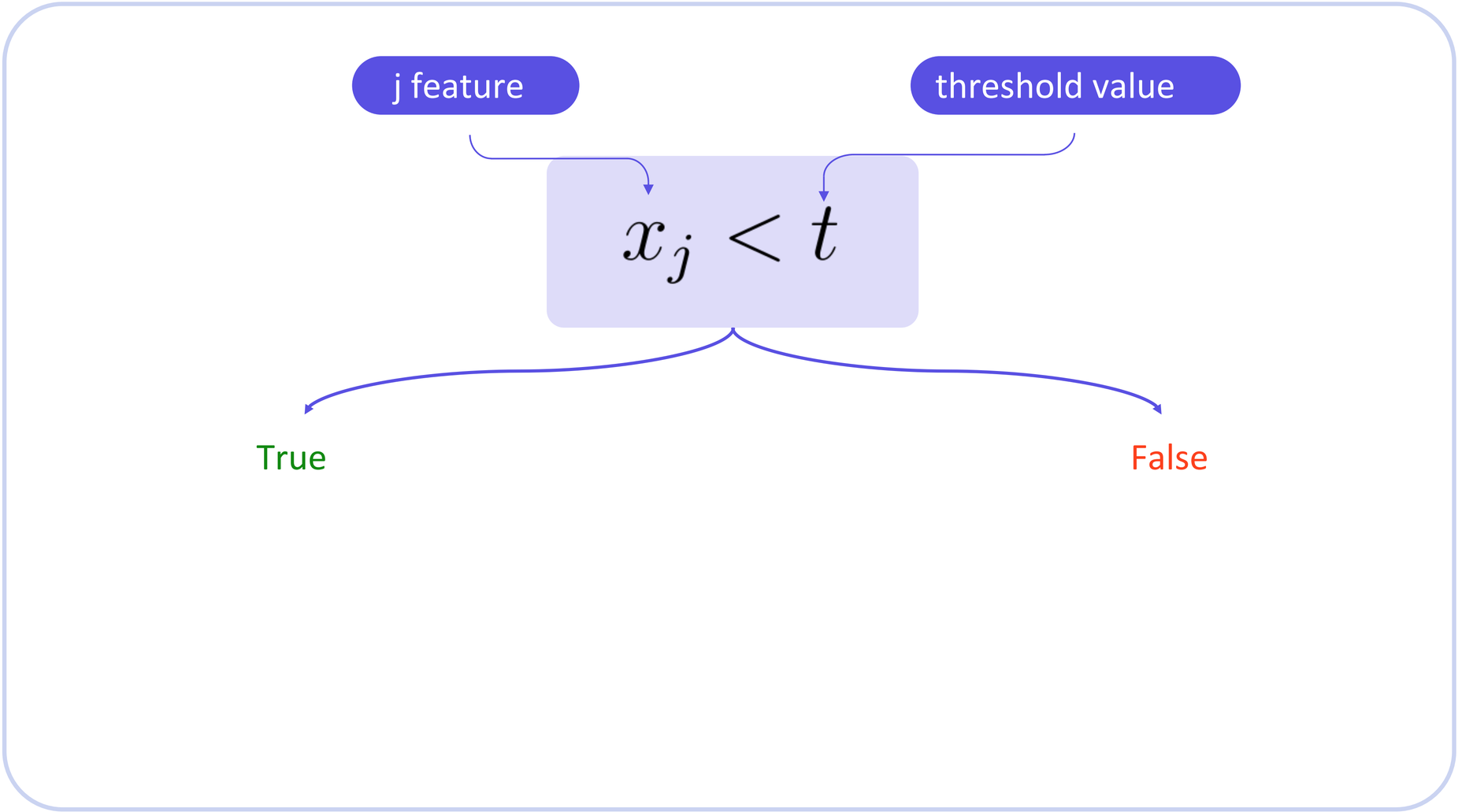
Every leaf corresponds to some constant

Decision Tree construction procedure

02

Constructing decision trees

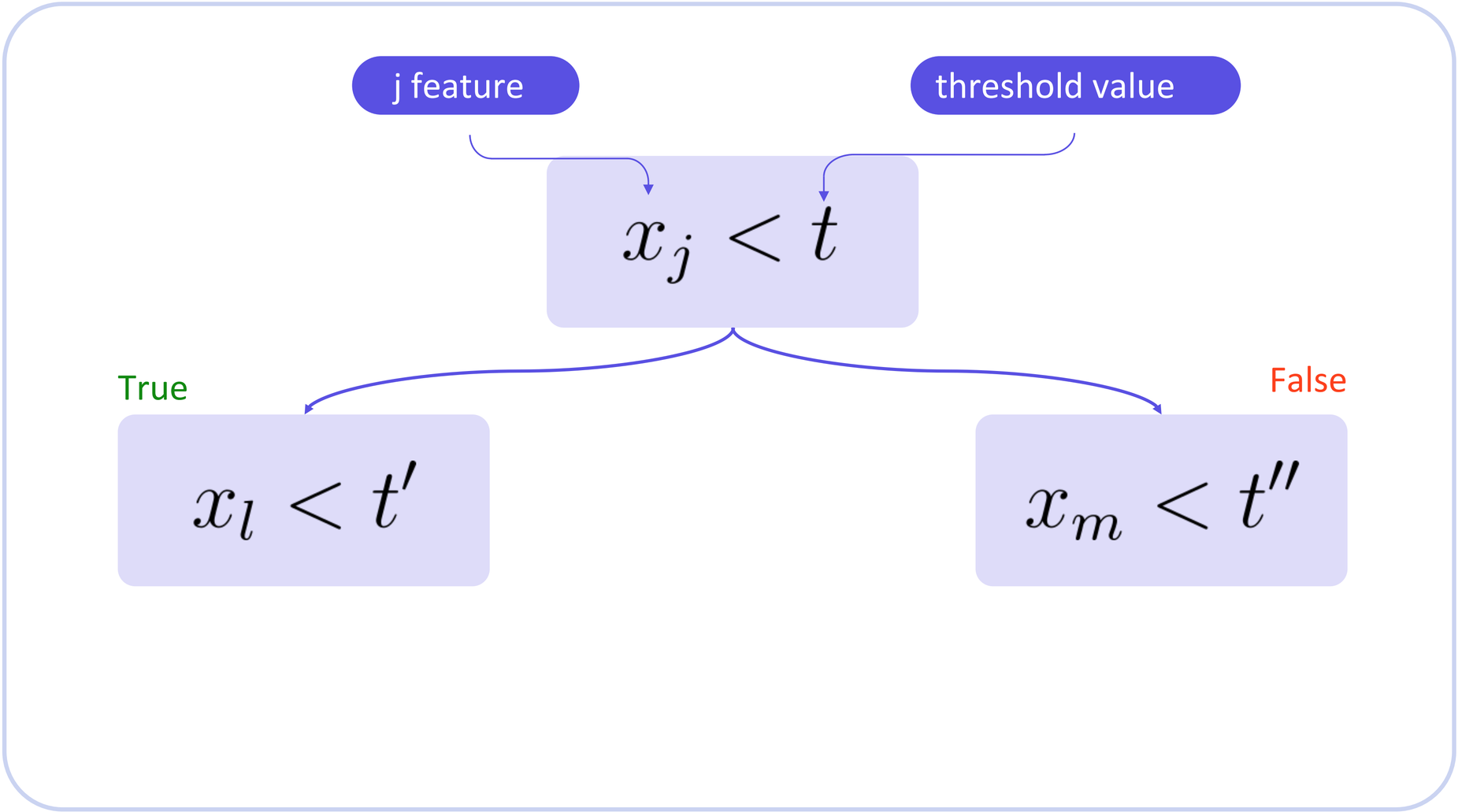
01 Make a split



Constructing decision trees

01 Make a split

02 Repeat

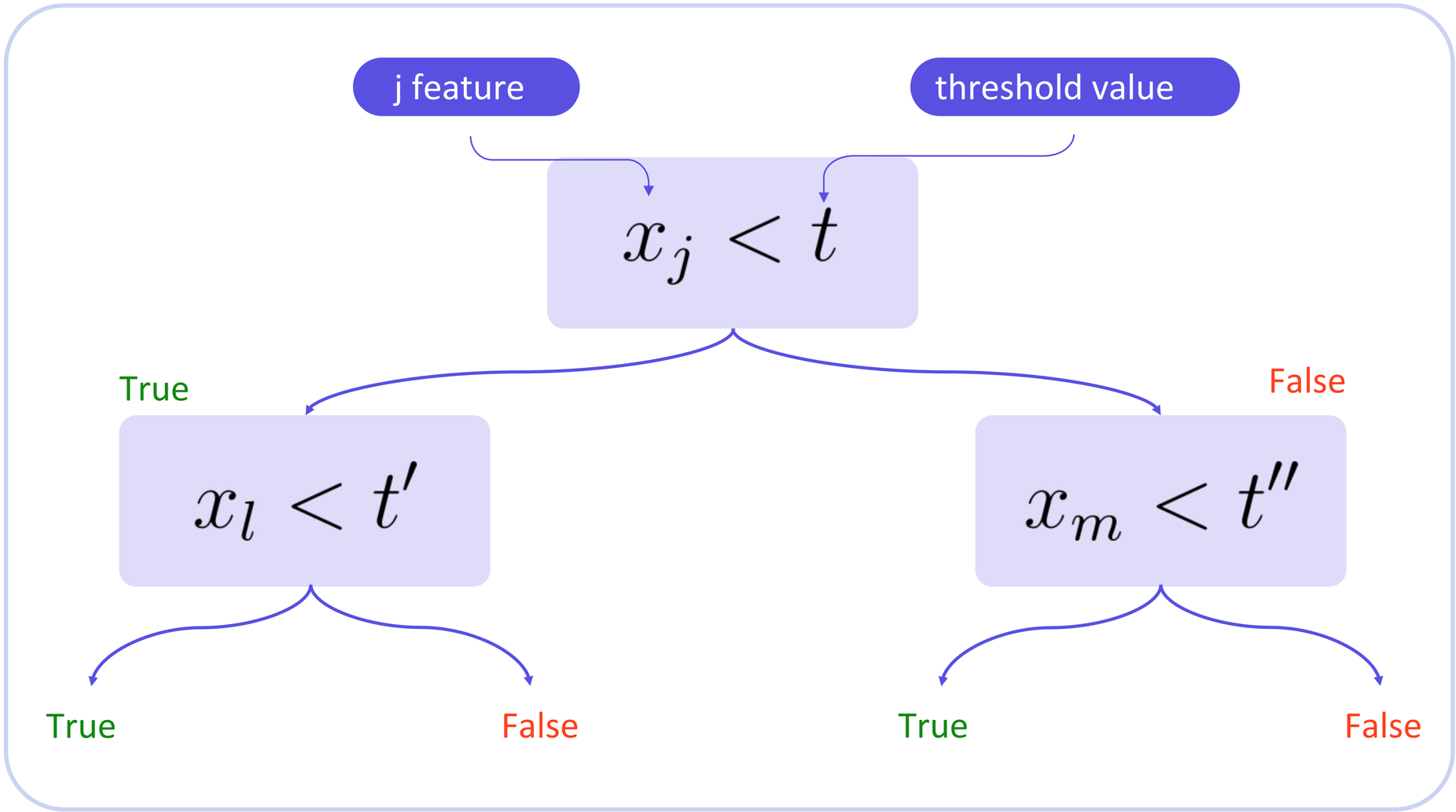


Constructing decision trees

01 Make a split

02 Repeat

03 Repeat



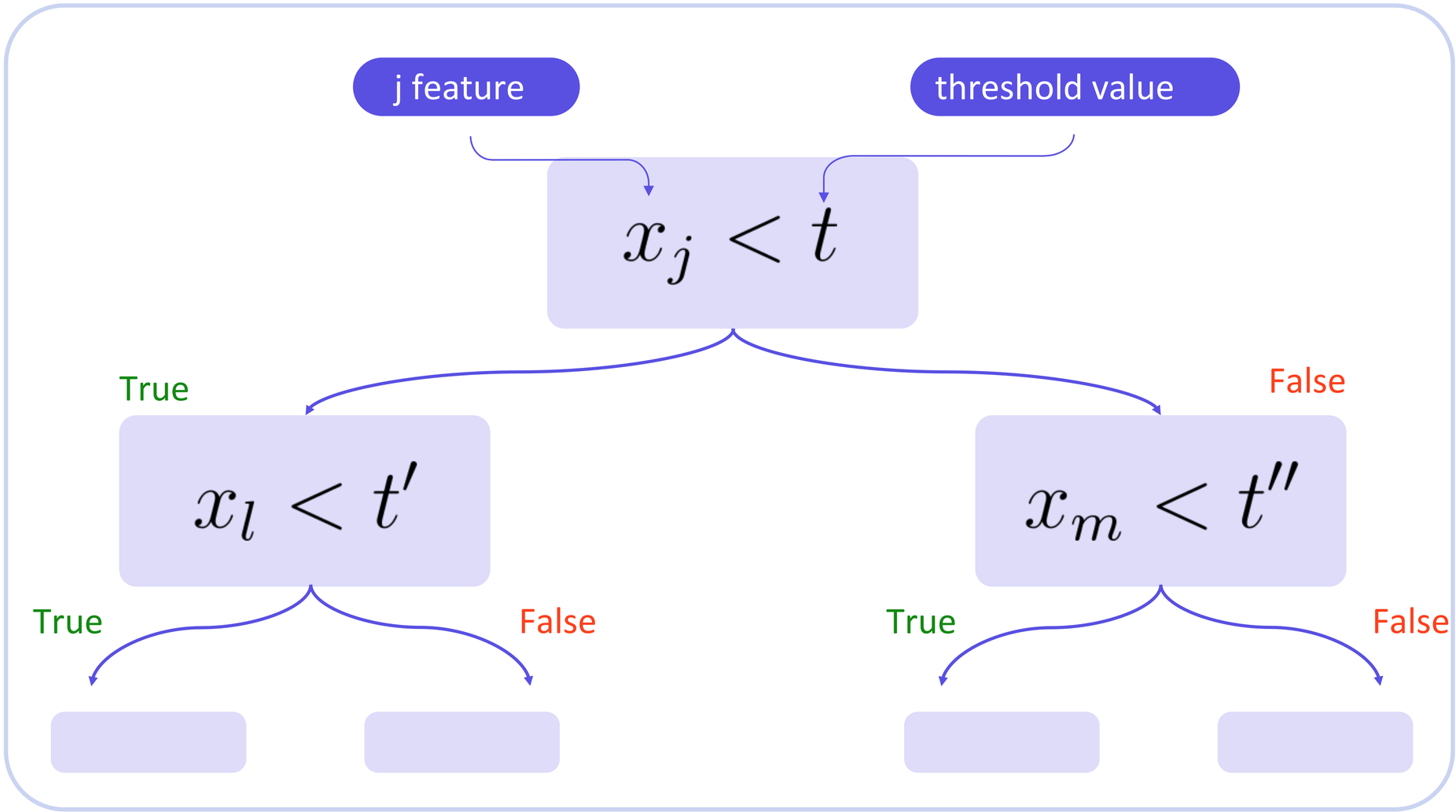
Constructing decision trees

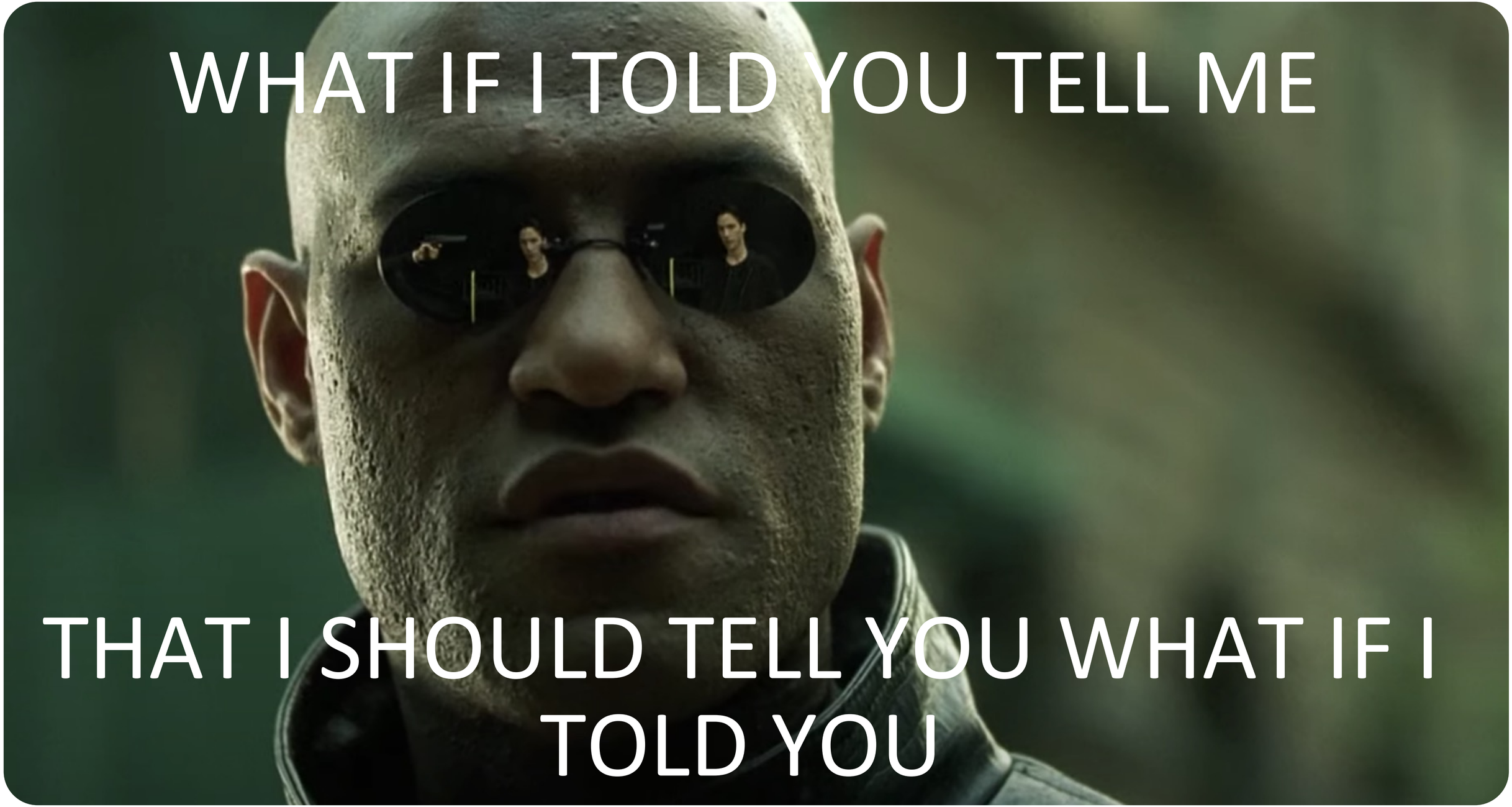
01 Make a split

02 Repeat

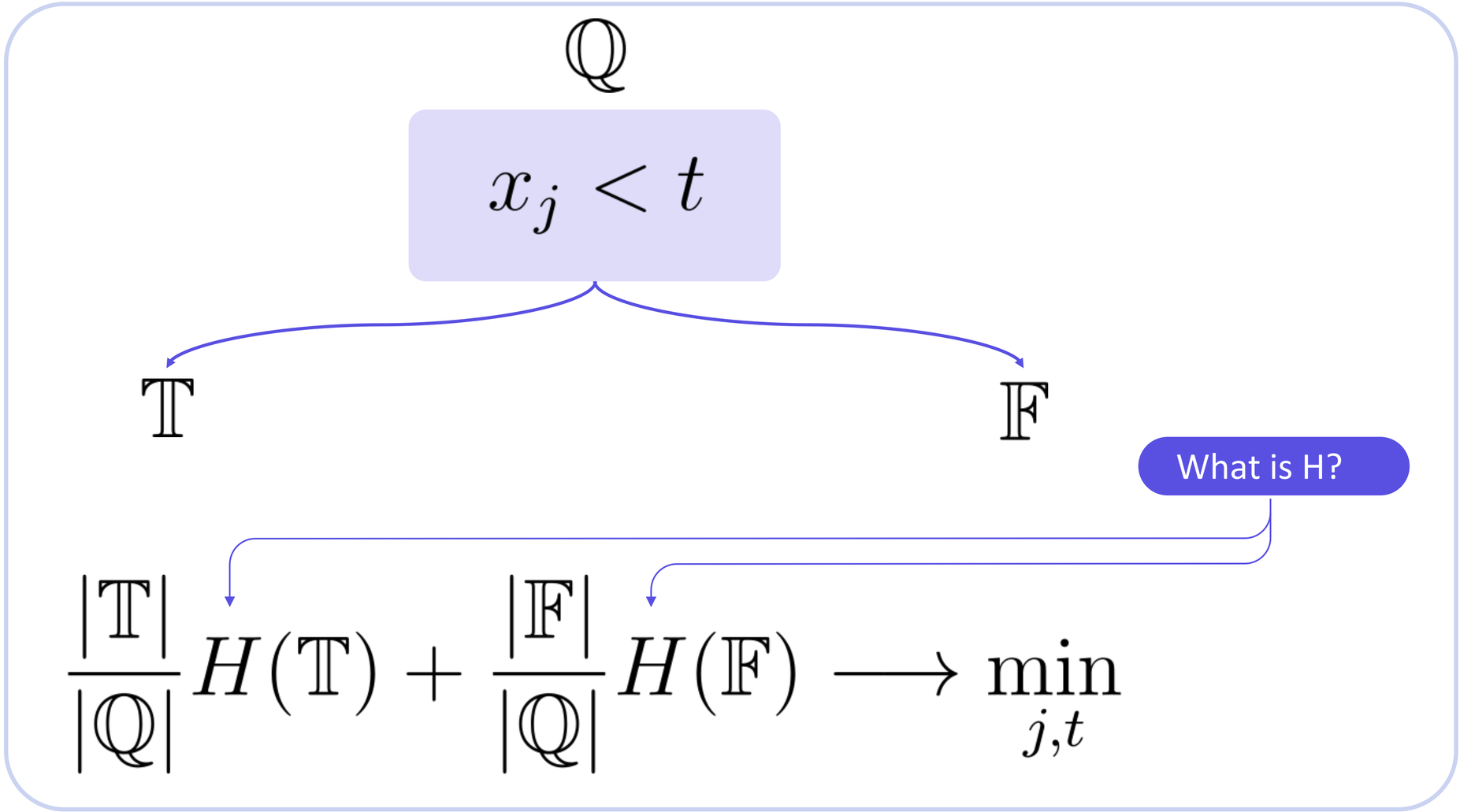
03 Repeat

04 …

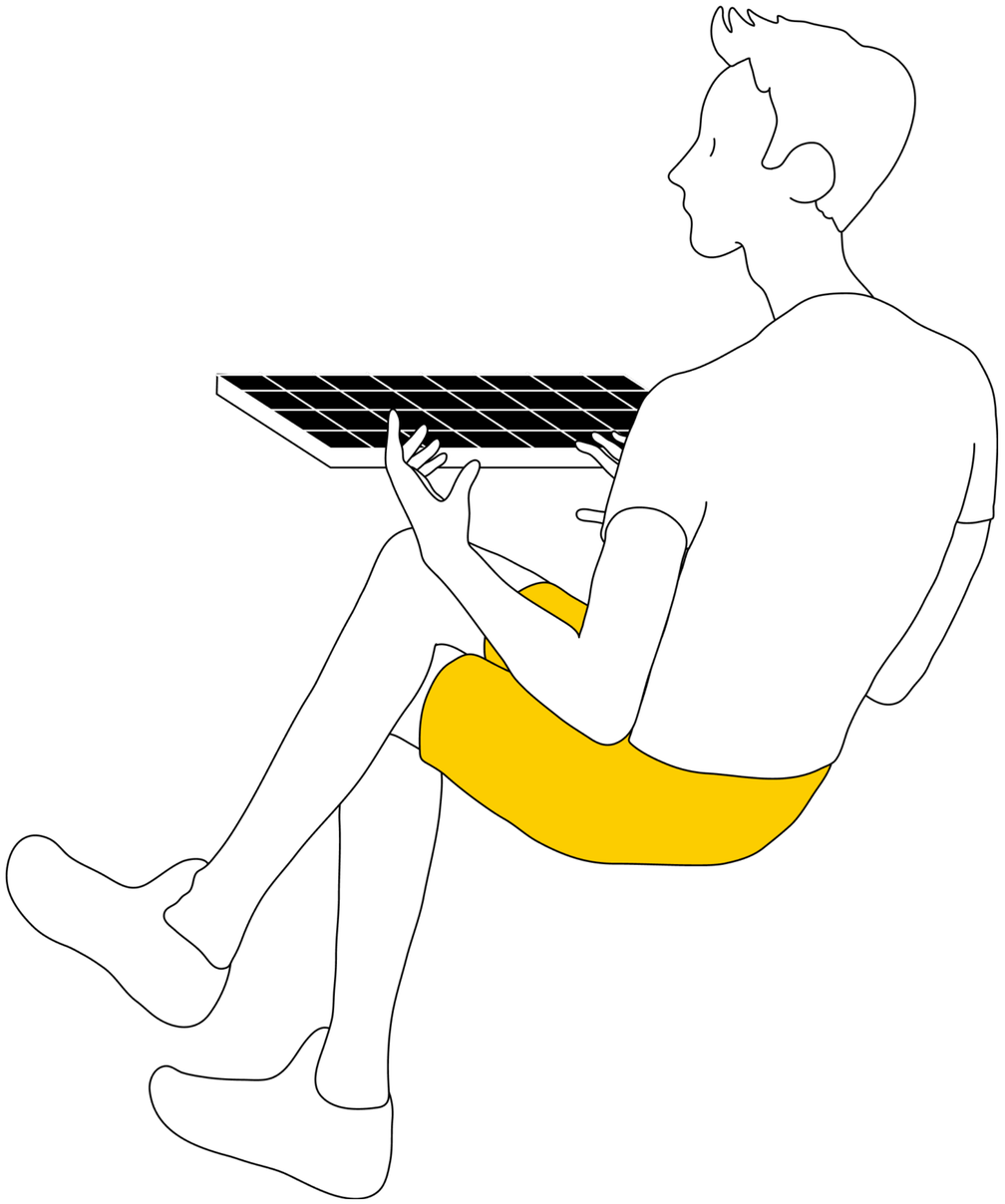



WHAT IF I TOLD YOU TELL ME
THAT I SHOULD TELL YOU WHAT IF I TOLD YOU

How to split data properly?



Information criteria

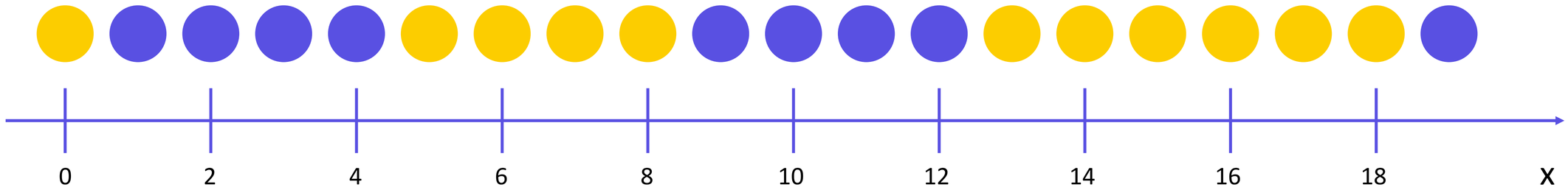


03

Information criteria

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

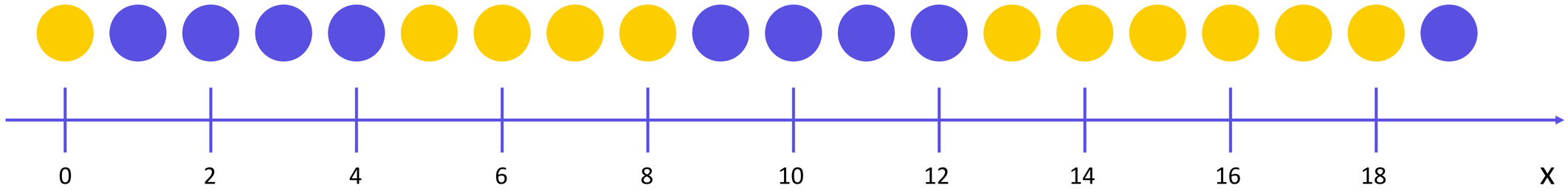
Consider binary classification problem:

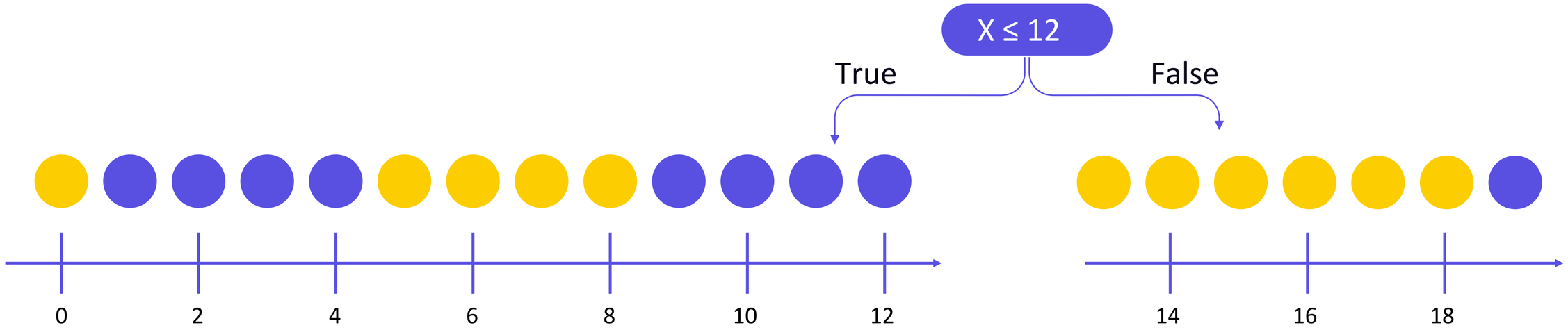


Information criteria

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

Consider binary classification problem:





Information criteria

$H(\mathbb{T})$ measures the “heterogeneity” of our data.

Consider binary classification problem:

Obvious way:
Misclassification criteria:

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

1. Entropy criteria:

$$H(\mathbb{T}) = -p_0 \log p_0 - p_1 \log p_1$$

2. Gini impurity:

$$H(\mathbb{T}) = 1 - p_0^2 - p_1^2$$

Information criteria

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

Consider multiclass classification problem:

Obvious way:
Misclassification criteria:

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

1. Entropy criteria:

$$H(\mathbb{T}) = -\sum_k p_k \log p_k$$

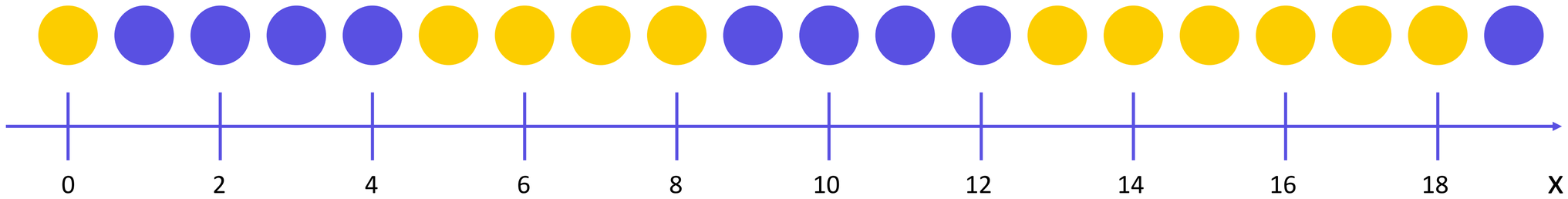
2. Gini impurity:

$$H(\mathbb{T}) = 1 - \sum_k p_k^2$$

Information criteria

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

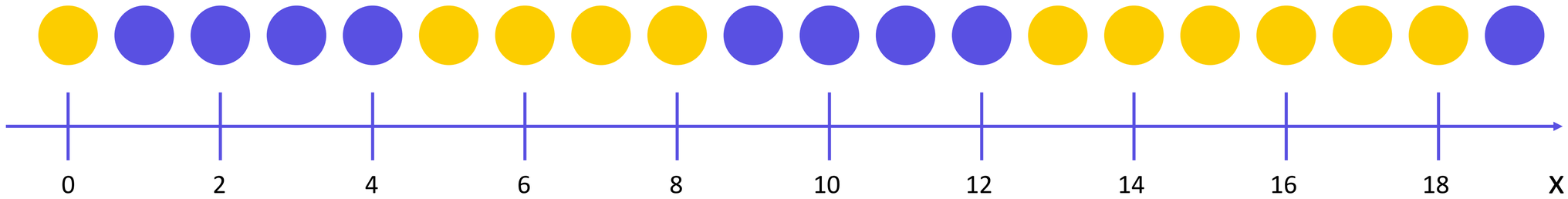
Consider binary classification problem:

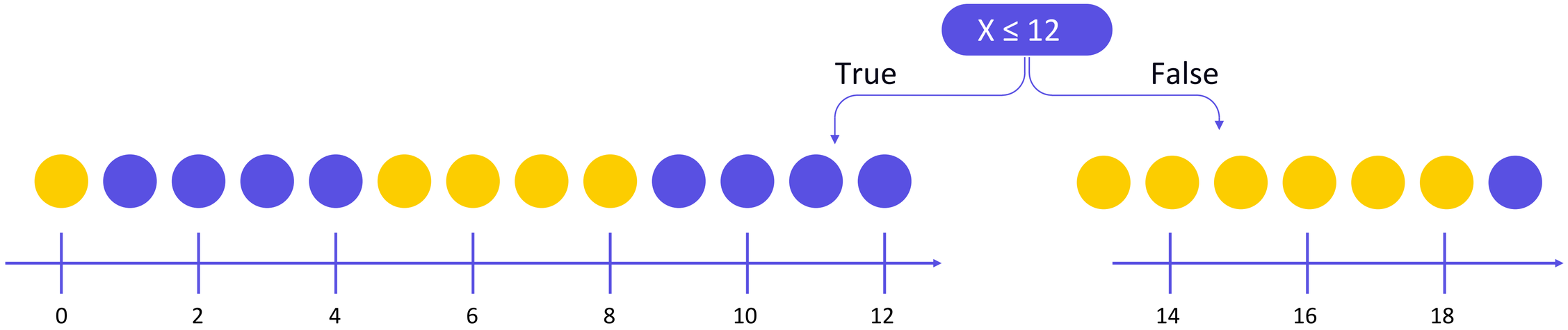


Information criteria

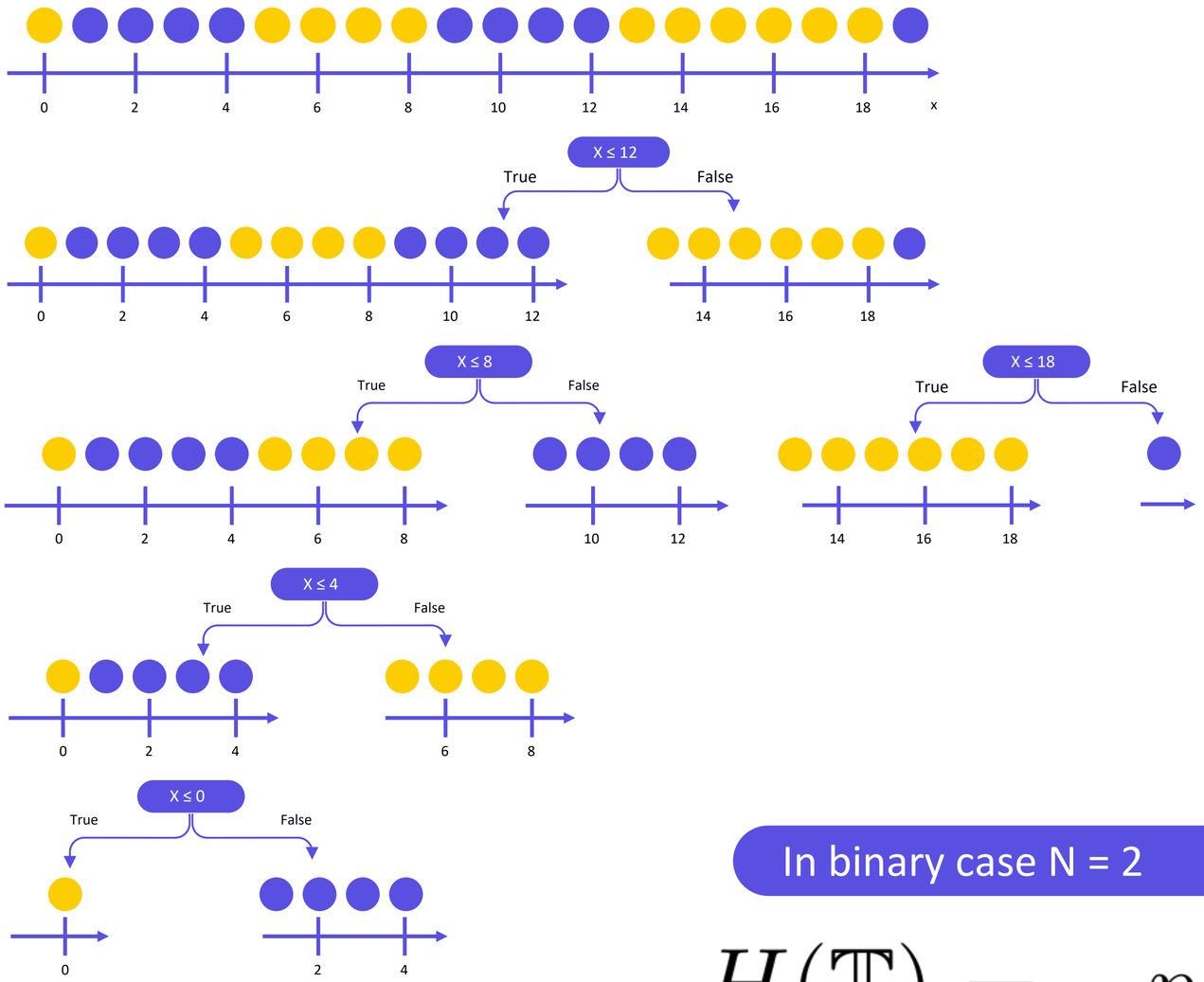
H(R) is measure of “heterogeneity” of our data.

Consider binary classification problem:





Information criteria: Entropy



In binary case N = 2

$$H(\mathbb{T}) = -p_0 \log p_0 - p_1 \log p_1$$

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

Information criteria: Gini impurity

$$H(\mathbb{T}) = 1 - \sum_k p_k^2$$

In binary case N = 2

$$H(\mathbb{T}) = 1 - p_0^2 - p_1^2 = 2p_0p_1$$

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

Information criteria

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

Consider multiclass classification problem:

Obvious way:
Misclassification criteria:

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

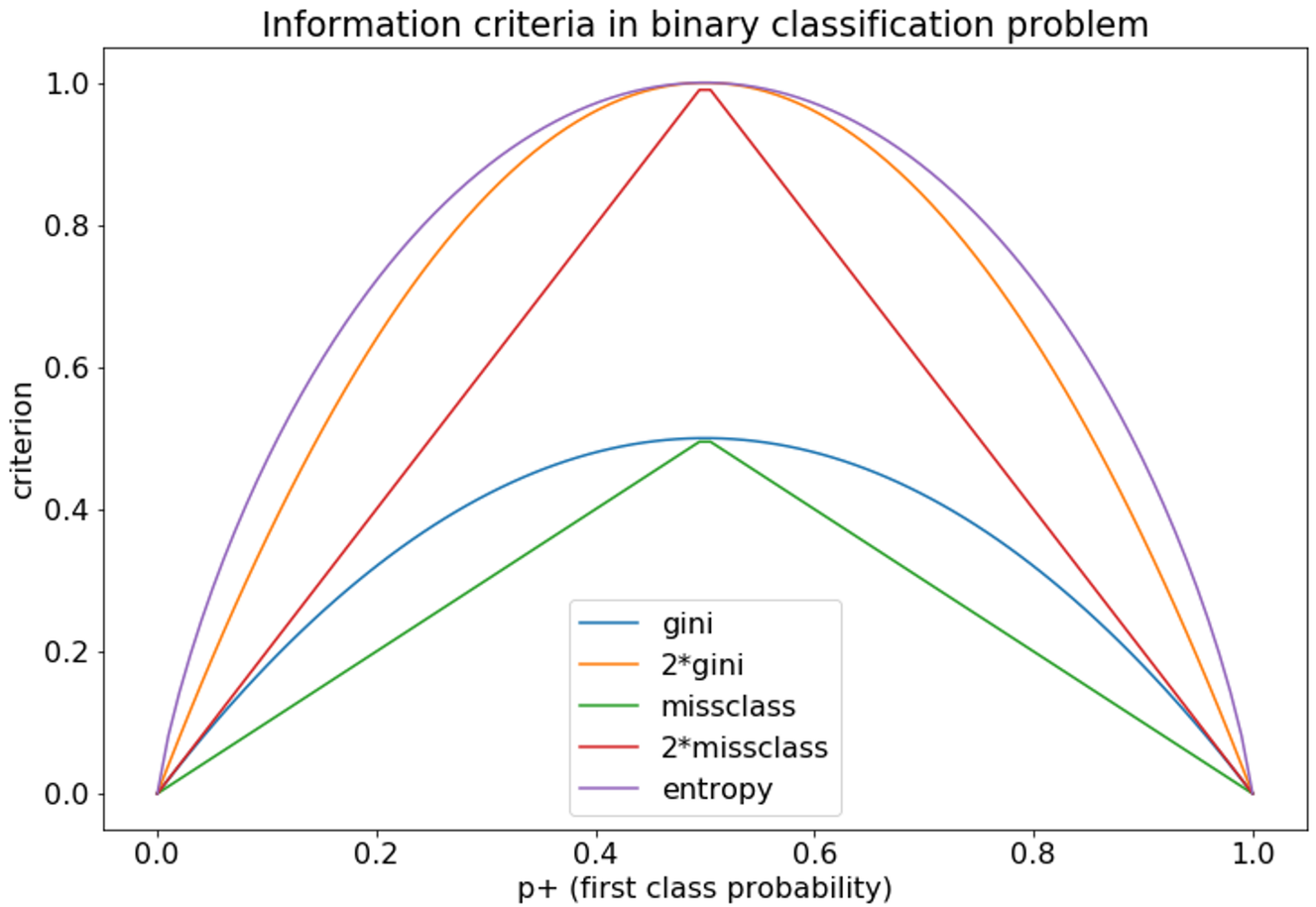
1. Entropy criteria:

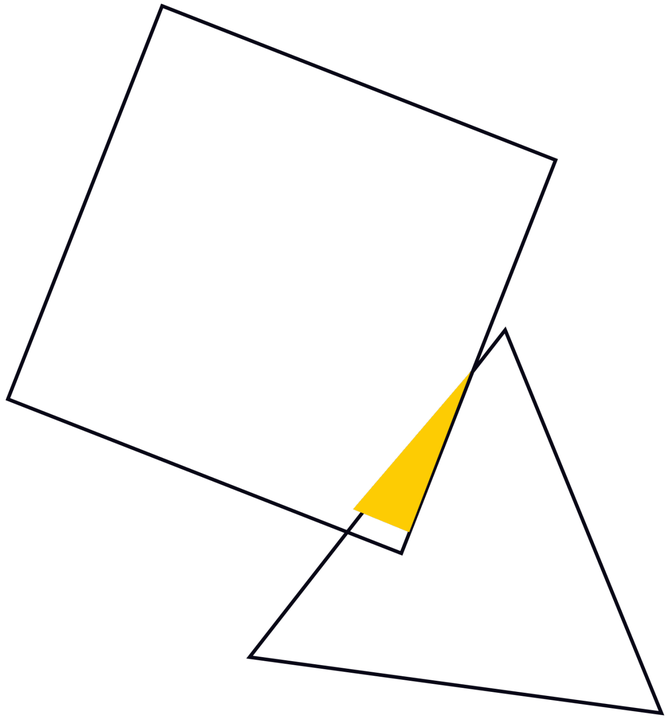
$$H(\mathbb{T}) = -\sum_k p_k \log p_k$$

2. Gini impurity:

$$H(\mathbb{T}) = 1 - \sum_k p_k^2$$

Information criteria





Information criteria

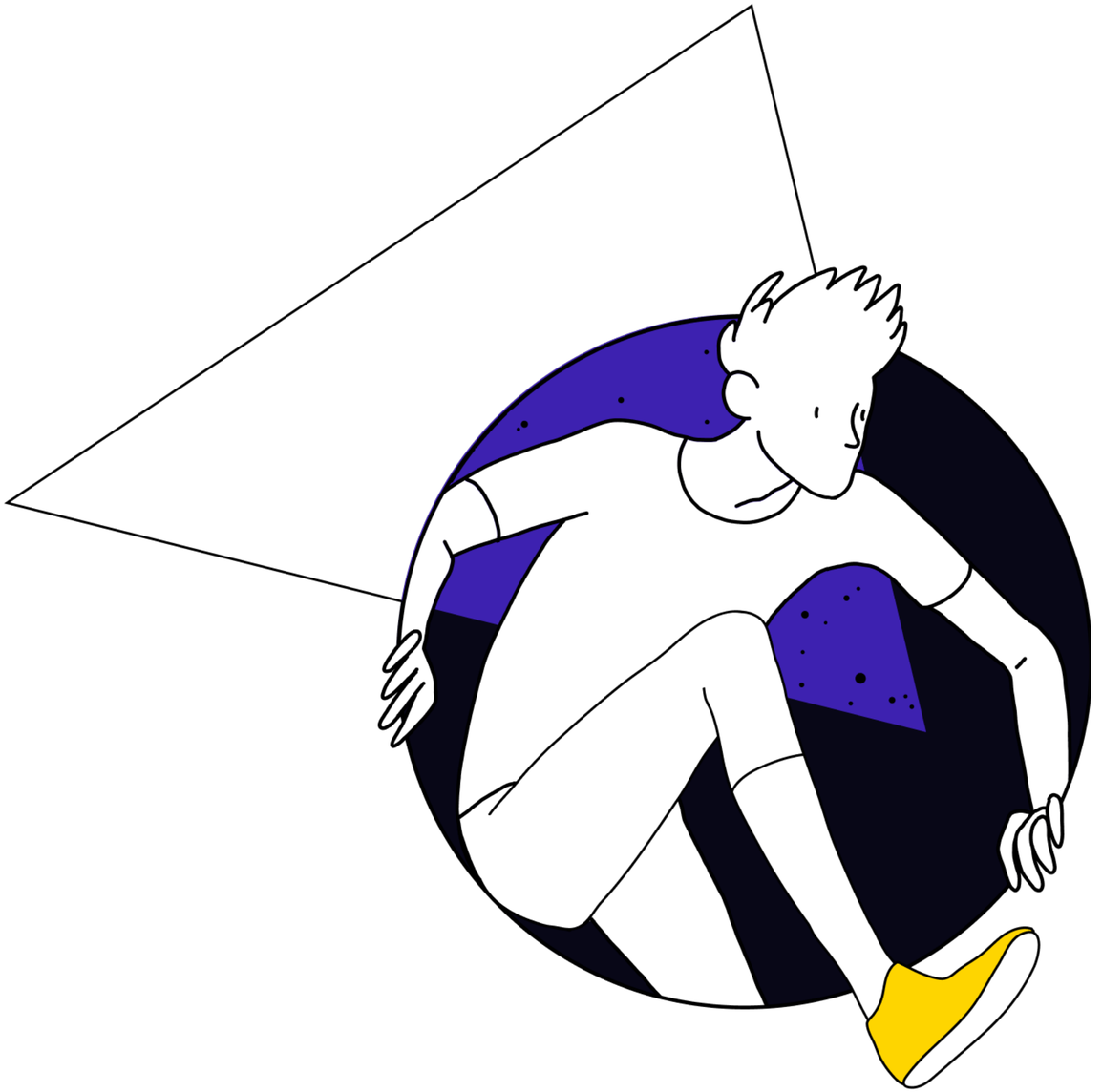
H(R) is measure of “heterogeneity” of our data.

Consider regression problem:

1. Mean squared error

$$H(\mathbb{T}) = \min_c \frac{1}{|\mathbb{T}|} \sum_k (y^{(k)} - c)^2$$

Pruning



04

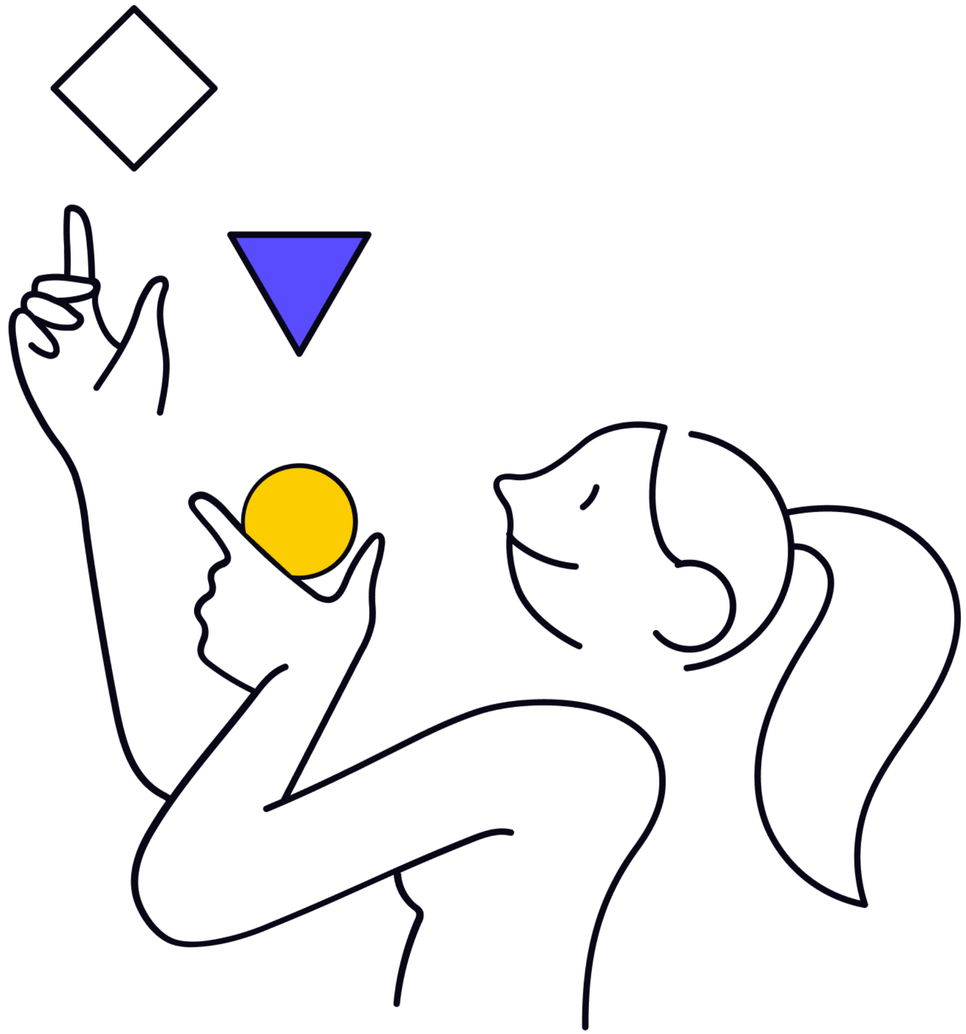
Pruning

01 Pre-pruning:

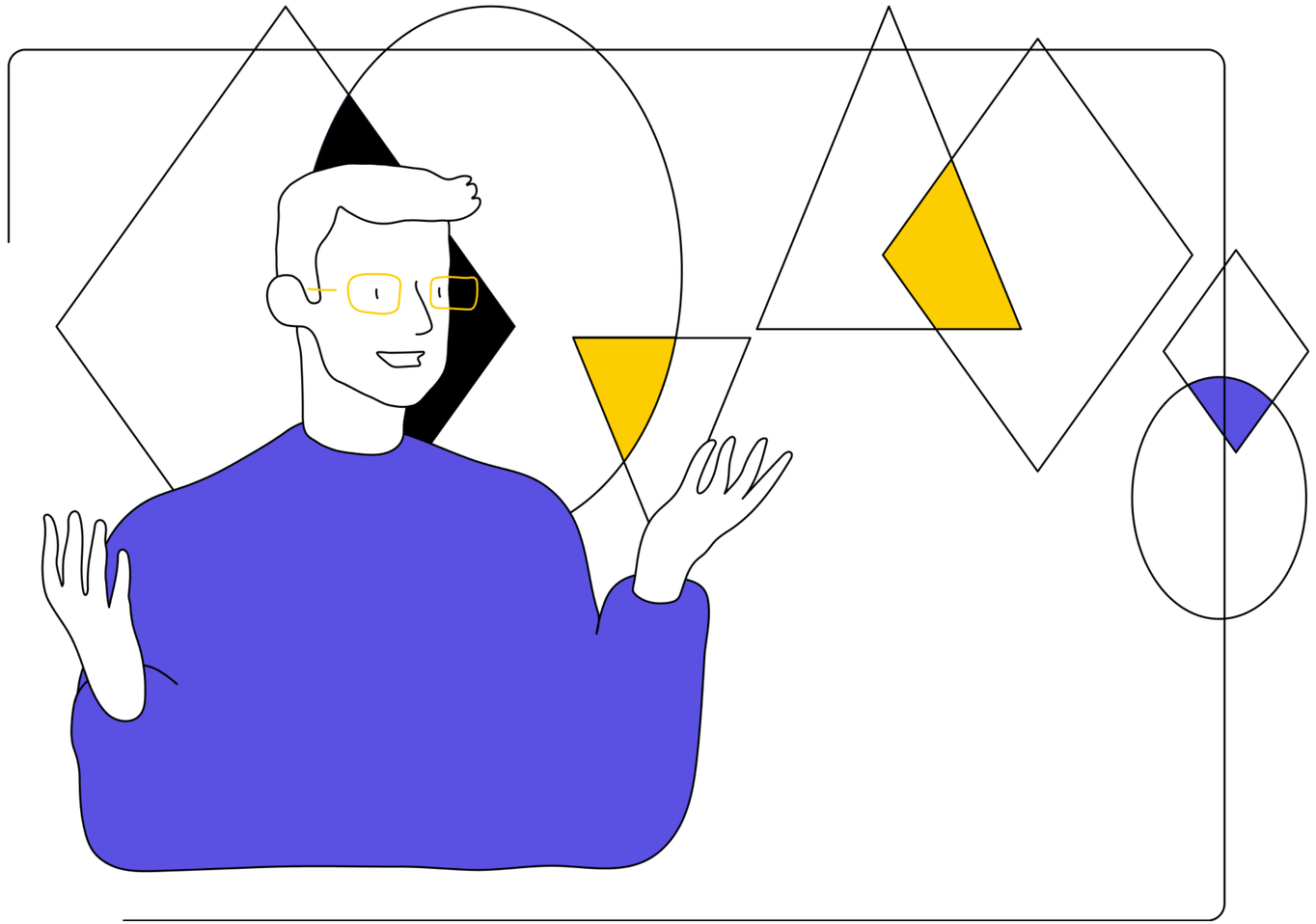
Constrain the tree before construction

02 Post-pruning:

Simplify constructed tree



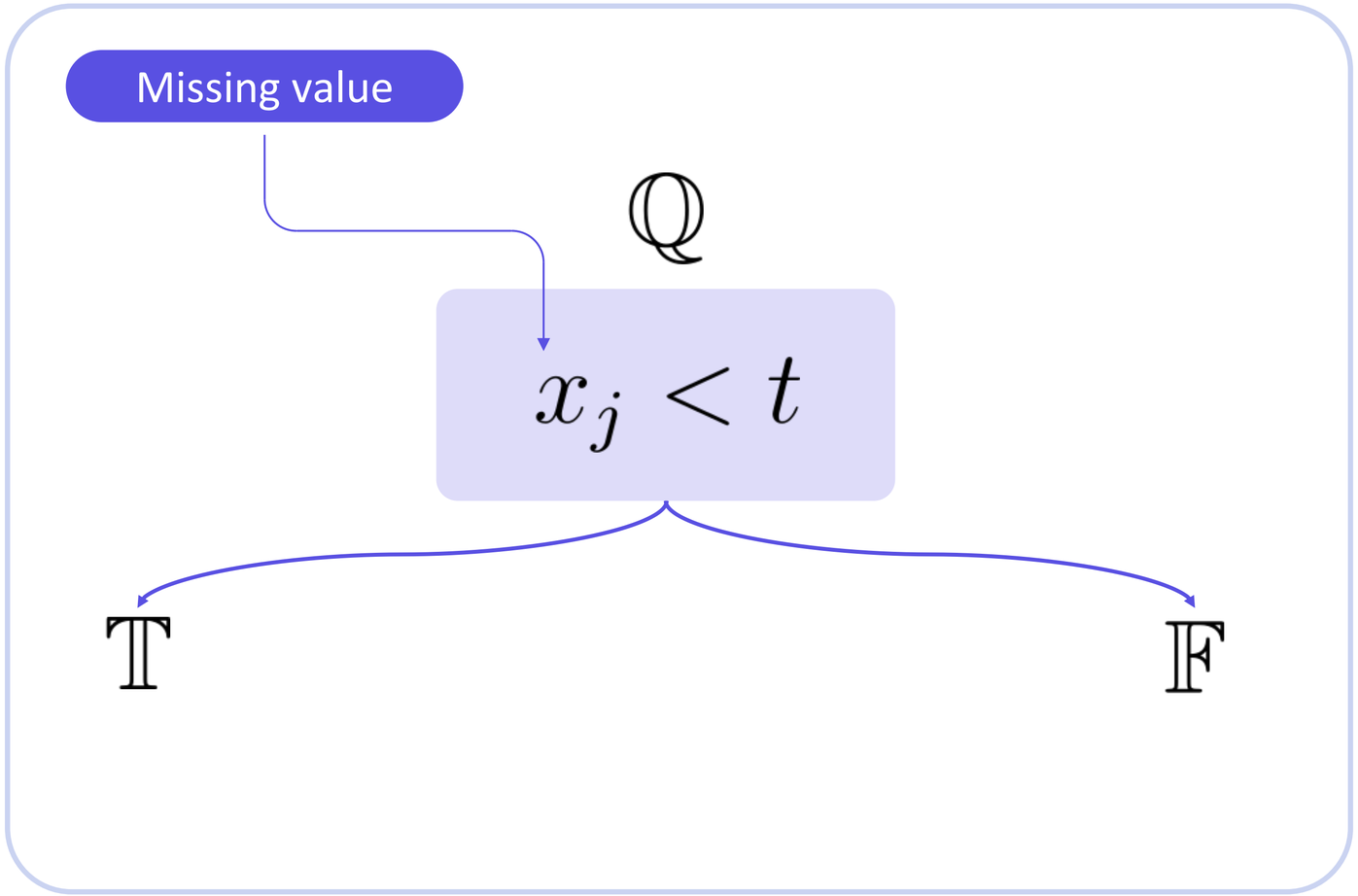
Special highlights



05

Missing values in Decision Trees

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



$$\hat{y} = \frac{|\mathbb{T}|}{|\mathbb{Q}|}\hat{y}_{\mathbb{T}} + \frac{|\mathbb{F}|}{|\mathbb{Q}|}\hat{y}_{\mathbb{F}}$$

Decision Trees as Linear models

Let $\mathbb{Q}$ be the subspace of the original feature space, corresponding to the leaf of the tree

Prediction takes form

$$\hat{y} = \sum_j c_j [\boldsymbol{x} \in \mathbb{Q}_j]$$

Construction algorithms: overview

01 **ID-3**
Entropy criteria; Stops when no more gain available

02 **C4.5**
Normalised entropy criteria; Stops depending on leaf size; Incorporates pruning

03 **C5.0**
Some updates on C4.5

04 **CART**
Gini criteria; Cost-complexity Pruning; Surrogate predicates for missing data

05 etc.

Bootstrap and Bagging

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 Xj: $$\varepsilon_j(\boldsymbol{x}) = b_j(\boldsymbol{x}) - y(\boldsymbol{x}), \quad j = 1, \ldots, N,$$

Then $$\mathbb{E}_{\boldsymbol{x}} \left[b_j(\boldsymbol{x}) - y(\boldsymbol{x})\right]^2 = \mathbb{E}_{\boldsymbol{x}} \varepsilon_j^2(\boldsymbol{x}).$$

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

Bootstrap

Consider the errors unbiased and uncorrelated:

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

The final model averages all predictions:

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

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

Error decreased by N times!

Bootstrap

This is a lie

Consider the errors ~~unbiased and uncorrelated:~~

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

The final model averages all predictions:

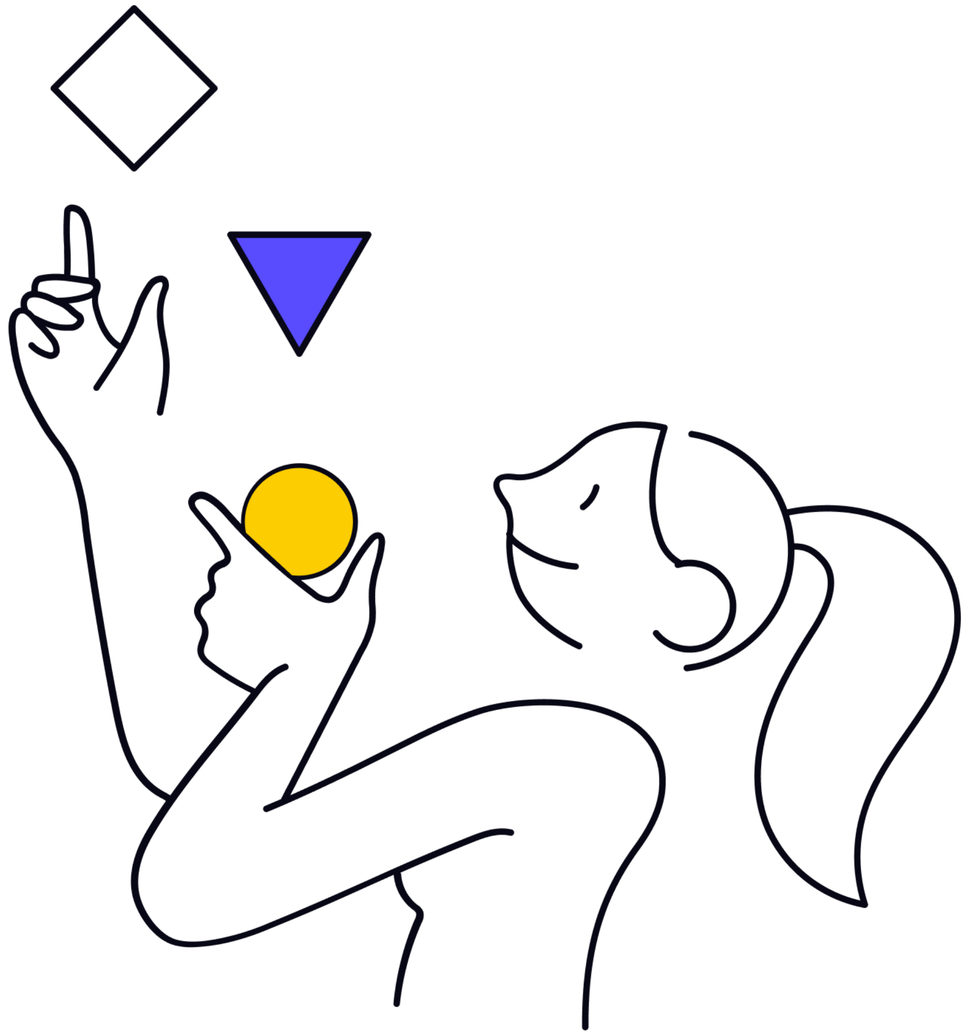
$$a(\boldsymbol{x}) = \frac{1}{N}\sum_{j=1}^{N} b_j(\boldsymbol{x}).$$

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

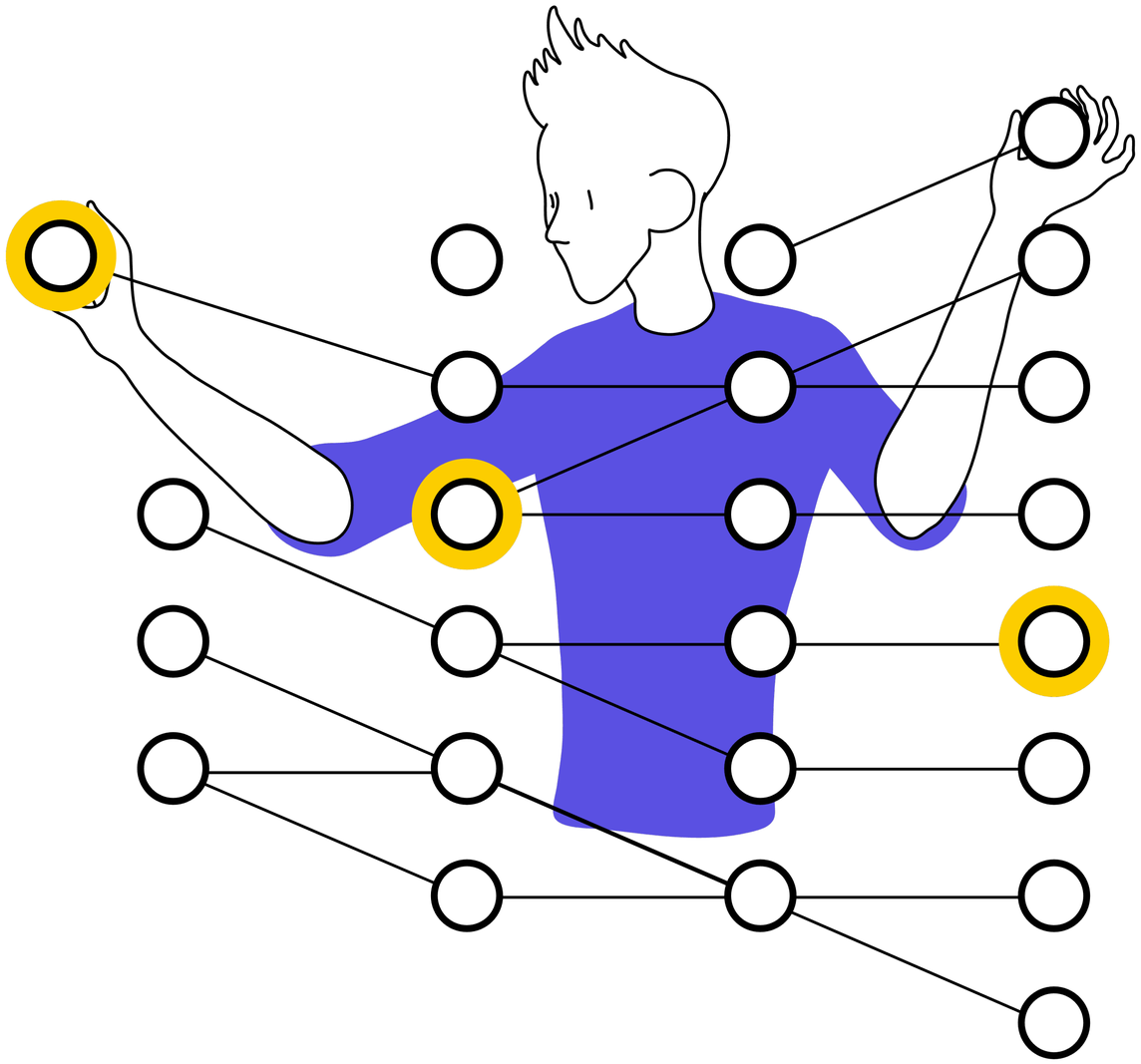
Error decreased by N times!

Bagging = Bootstrap aggregating

Decreases the variance if the basic algorithms are not correlated



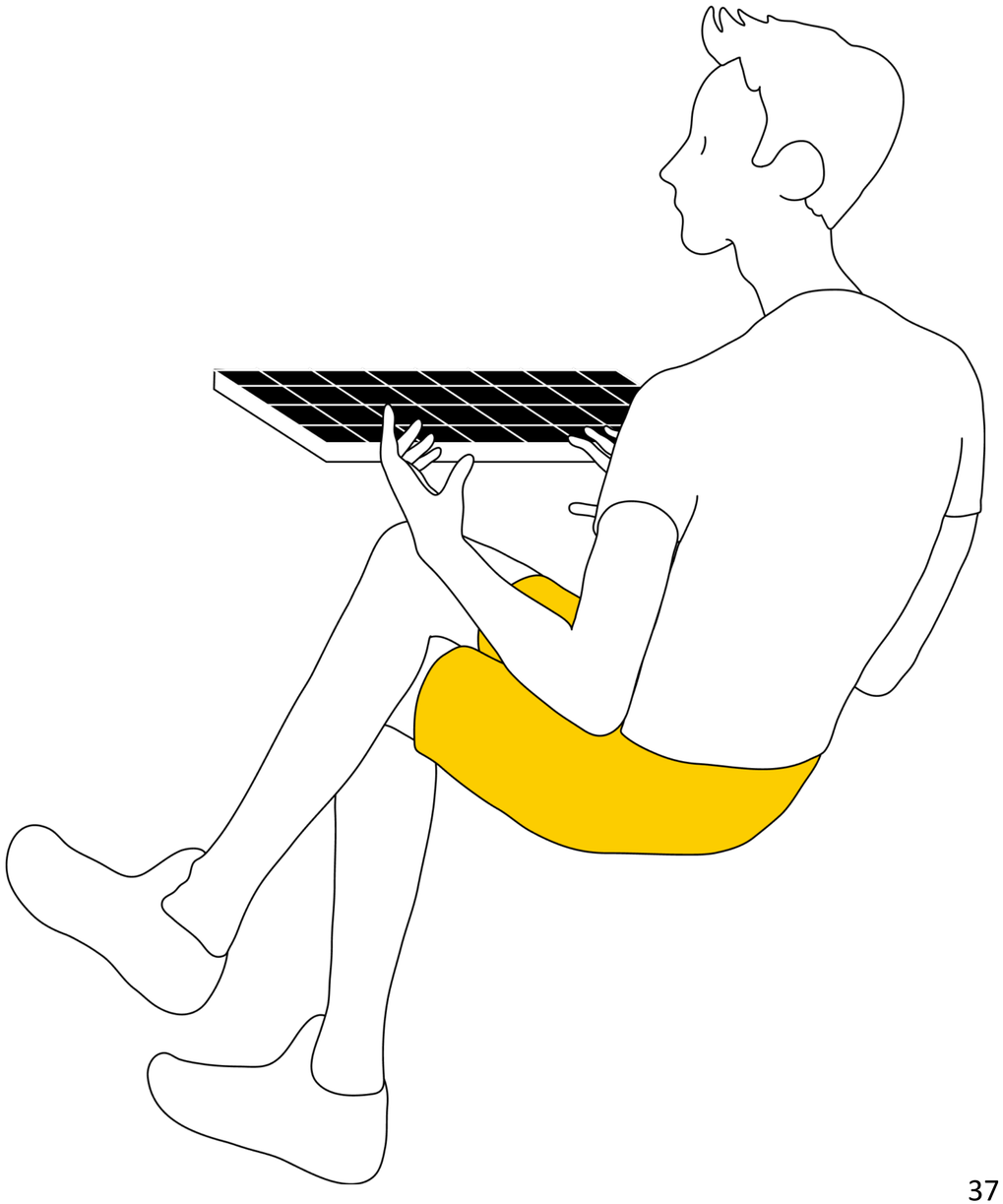
Random Forest



07

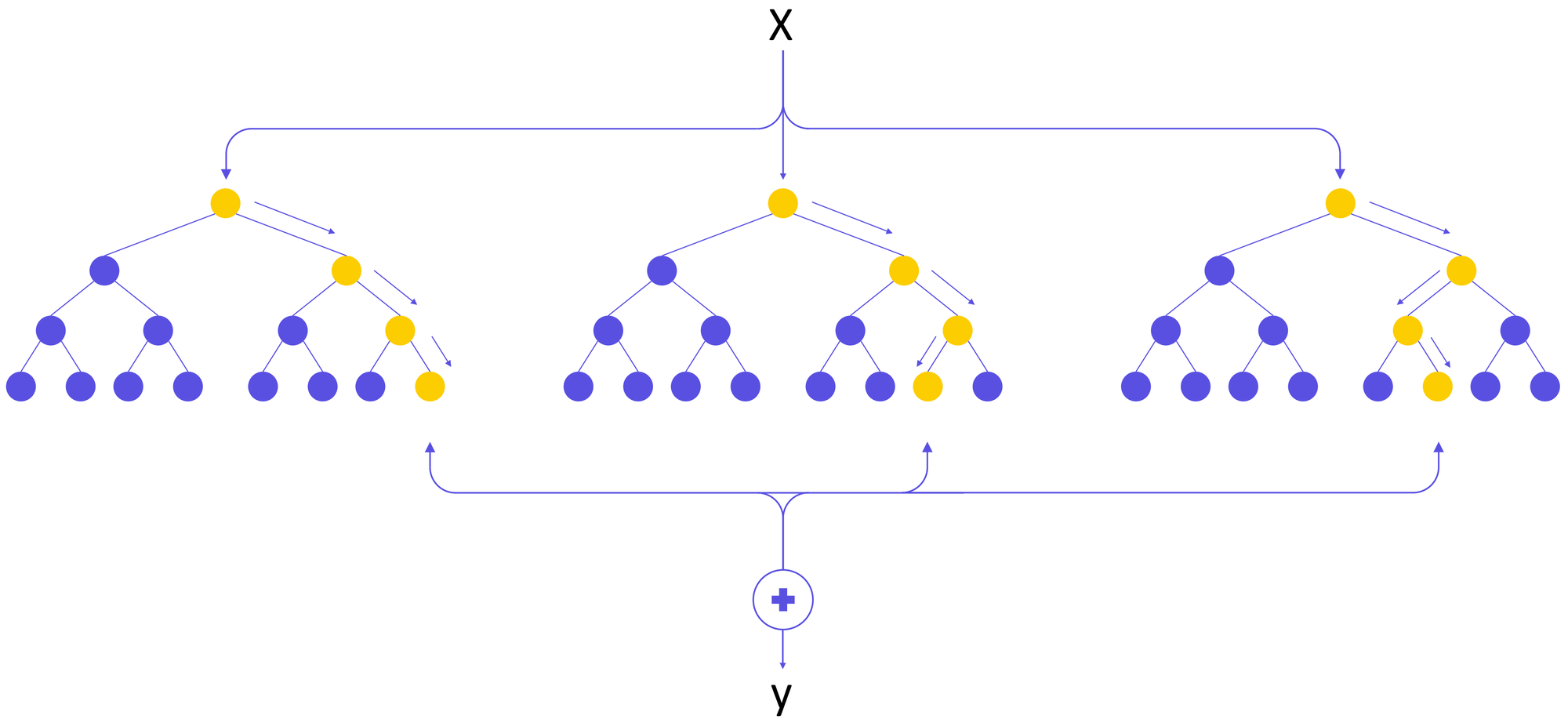
RSM — Random Subspace Method

Same approach,
but with features



Random Forest

Bagging + RSM = Random Forest



Random Forest

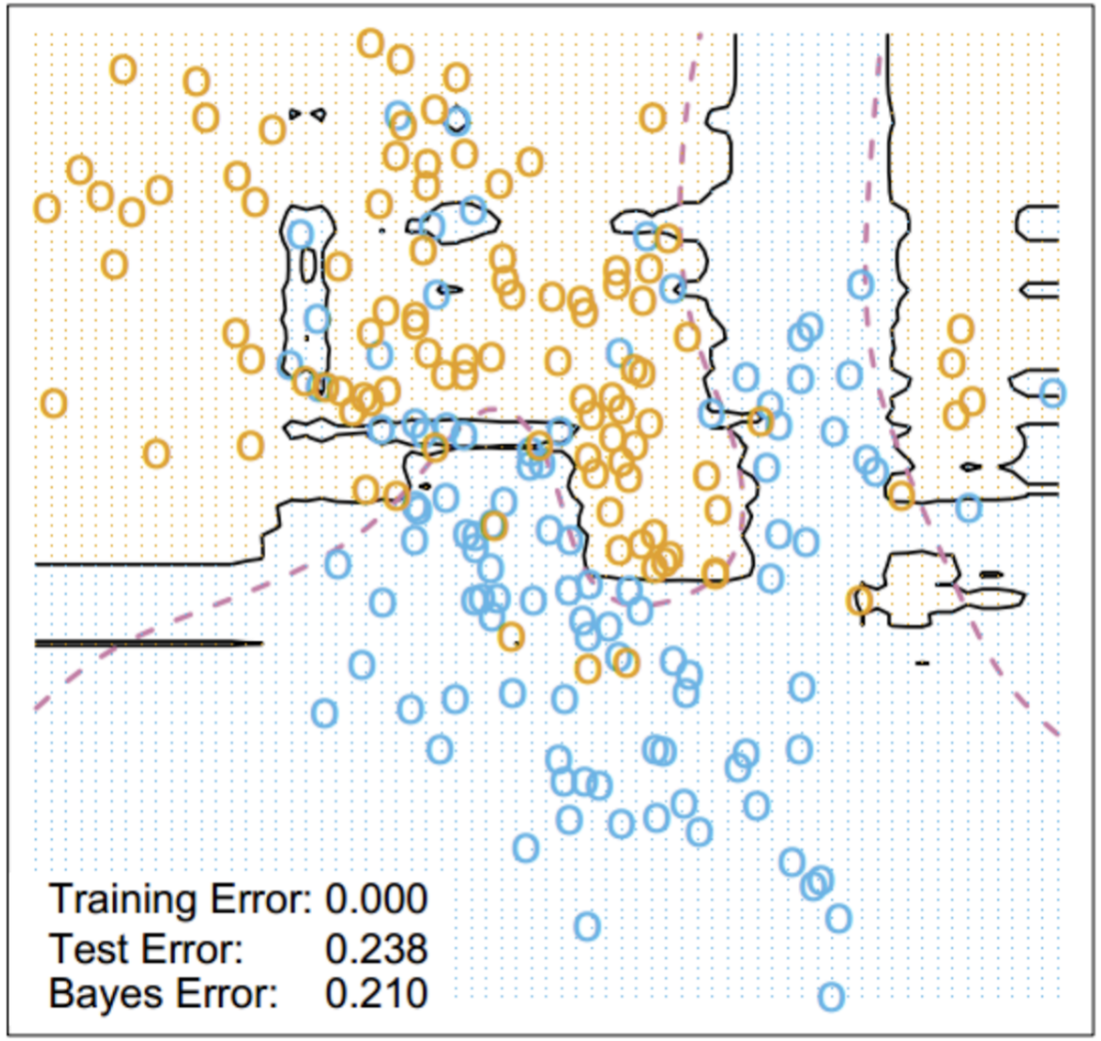
01 One of the greatest “universal” models

02 There are some modifications: Extremely Randomized Trees, Isolation Forest, etc.

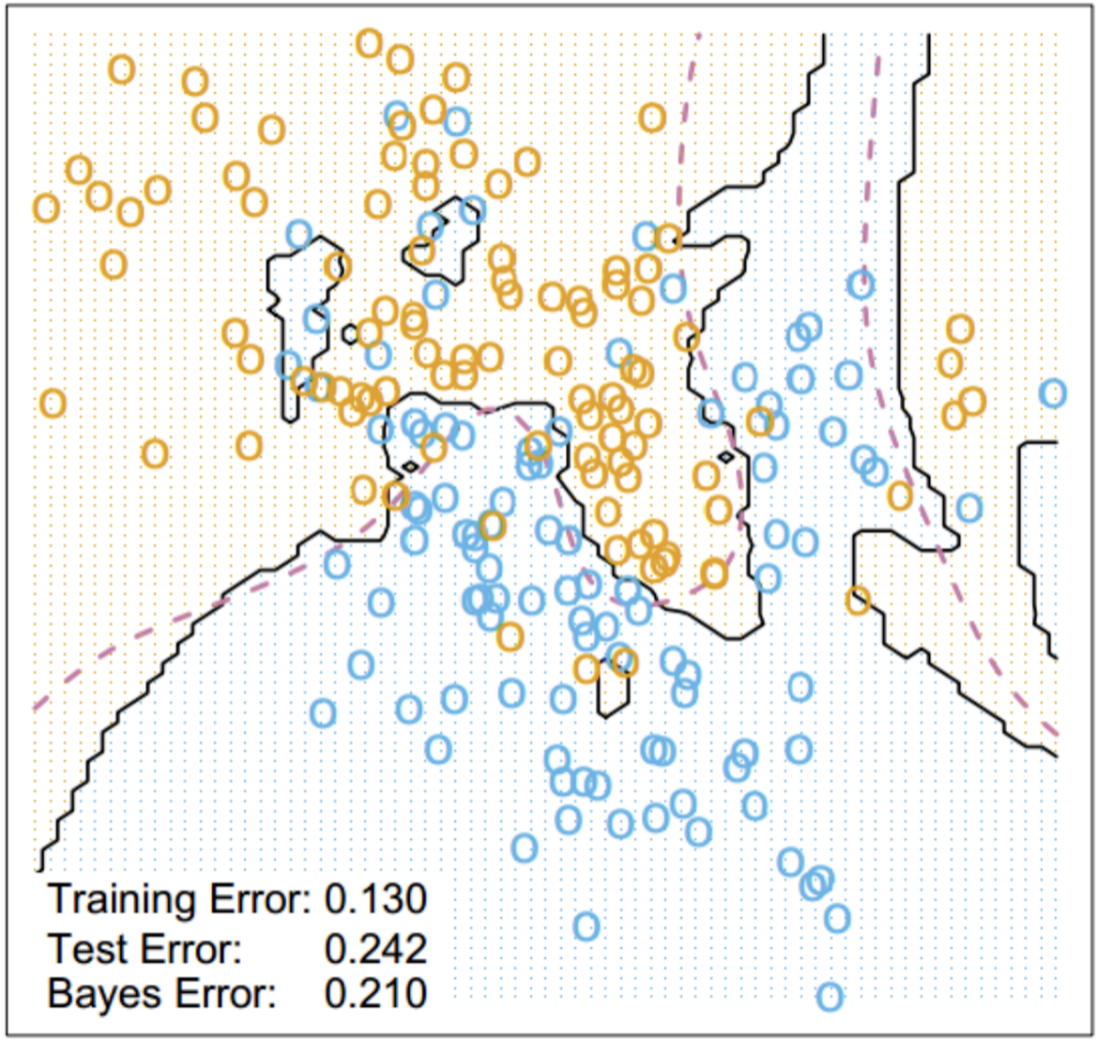
03 Allows to use train data for validation: OOB

$$\text{OOB} = \sum_{i=1}^{\ell} L\left(y^{(i)}, \frac{1}{\sum_{n=1}^{N}[\boldsymbol{x}^{(i)} \notin \boldsymbol{X}_n]} \sum_{n=1}^{N}[\boldsymbol{x}^{(i)} \notin \boldsymbol{X}_n] b_n(\boldsymbol{x}^{(i)})\right)$$

Random Forest Classifier



3-Nearest Neighbors



Revise

01 Decision tree: intuition

02 Decision tree construction procedure

03 Information criteria

04 Pruning

05 Decision trees special highlights

- Decision tree as linear model
- Dealing with missing data
- Categorical features

06 Bootstrap and Bagging

07 Random Forest

Thanks for attention!

Questions?

Radoslav Neychev