Artificial Intelligence

Taro Sekiyama

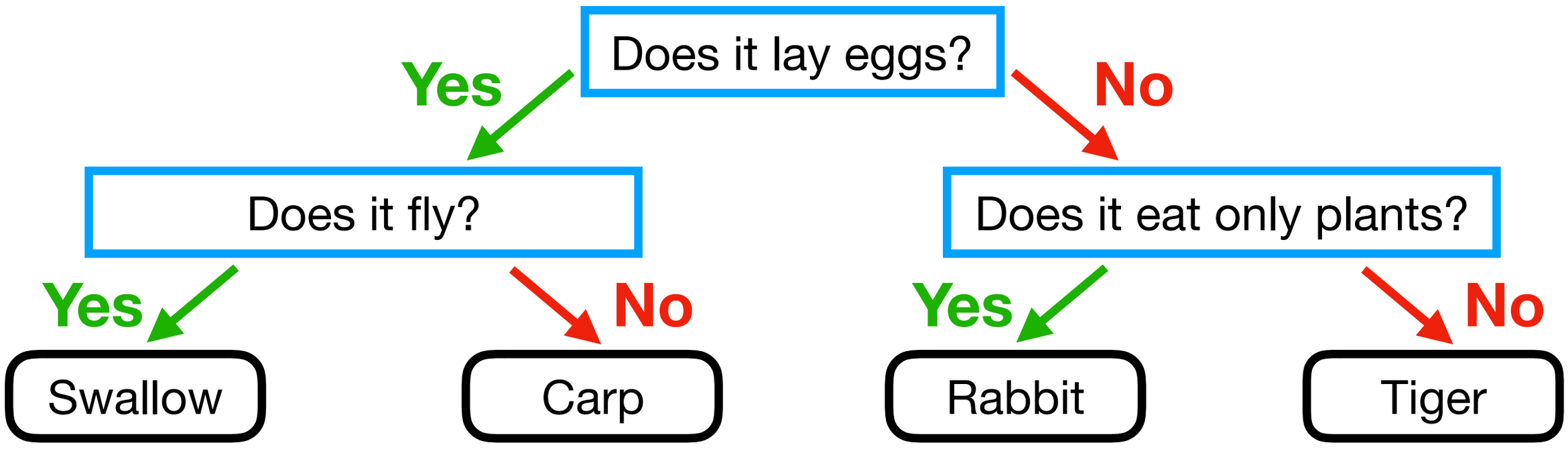
National Institute of Informatics (NII)
sekiyama@nii.ac.jp

Agenda

1. Decision trees
2. Random forests

Decision trees

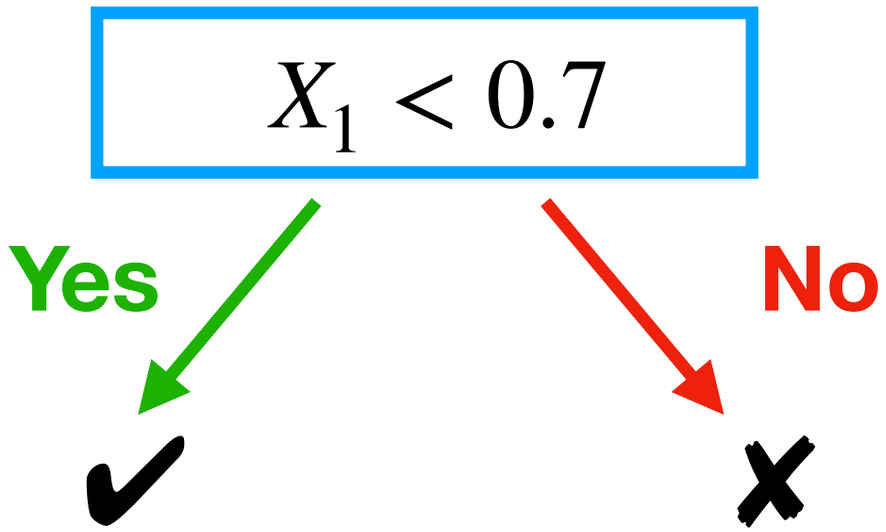
- Making ***decision*** by answering yes/no questions
- The process is drawn by ***trees***
- Ex: classification of animals: Rabbit, Tiger, Swallow, and Carp

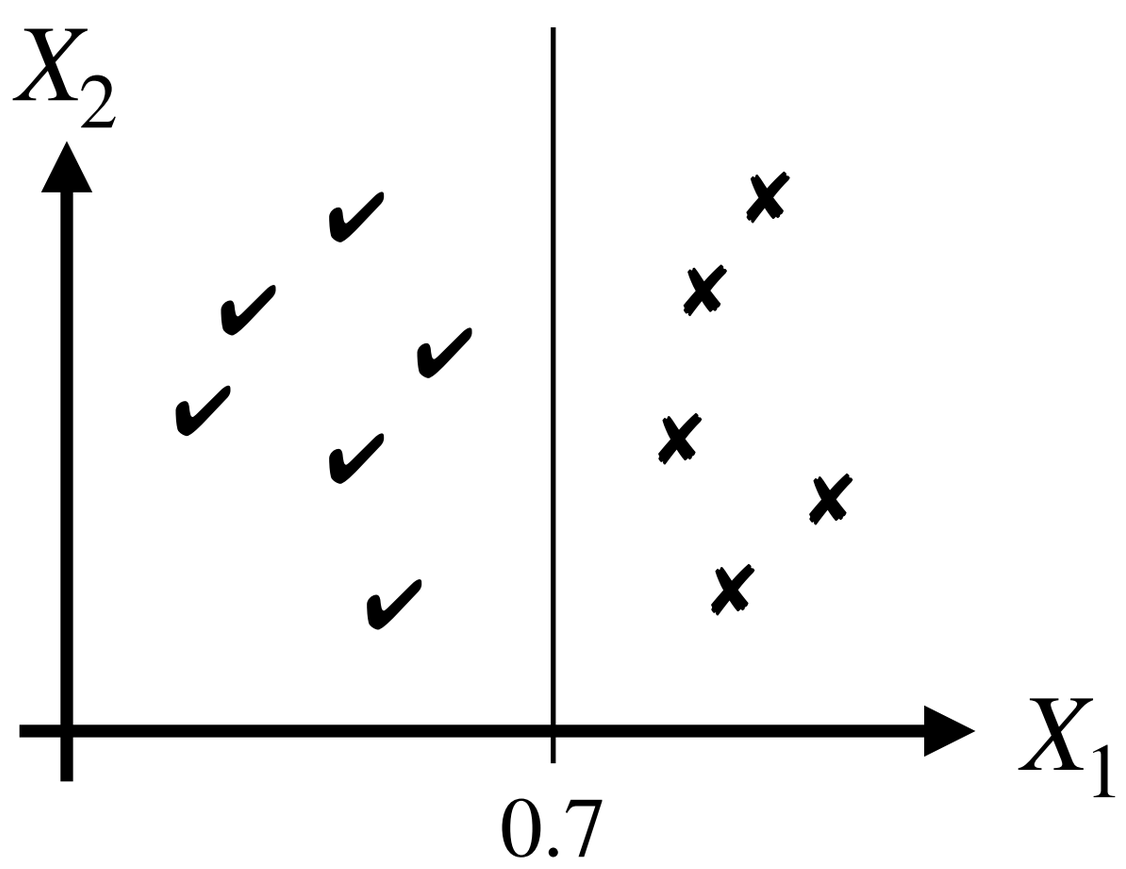


Decision on features

- Decision on real-valued features is made by answering:

  “feature X is less than real value R?”

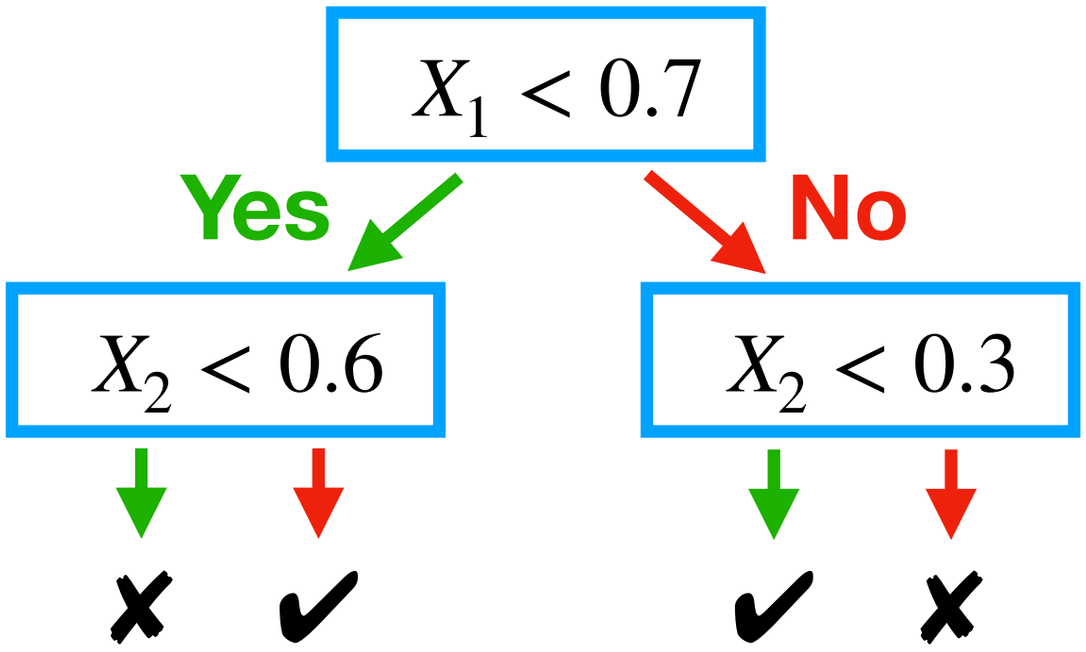


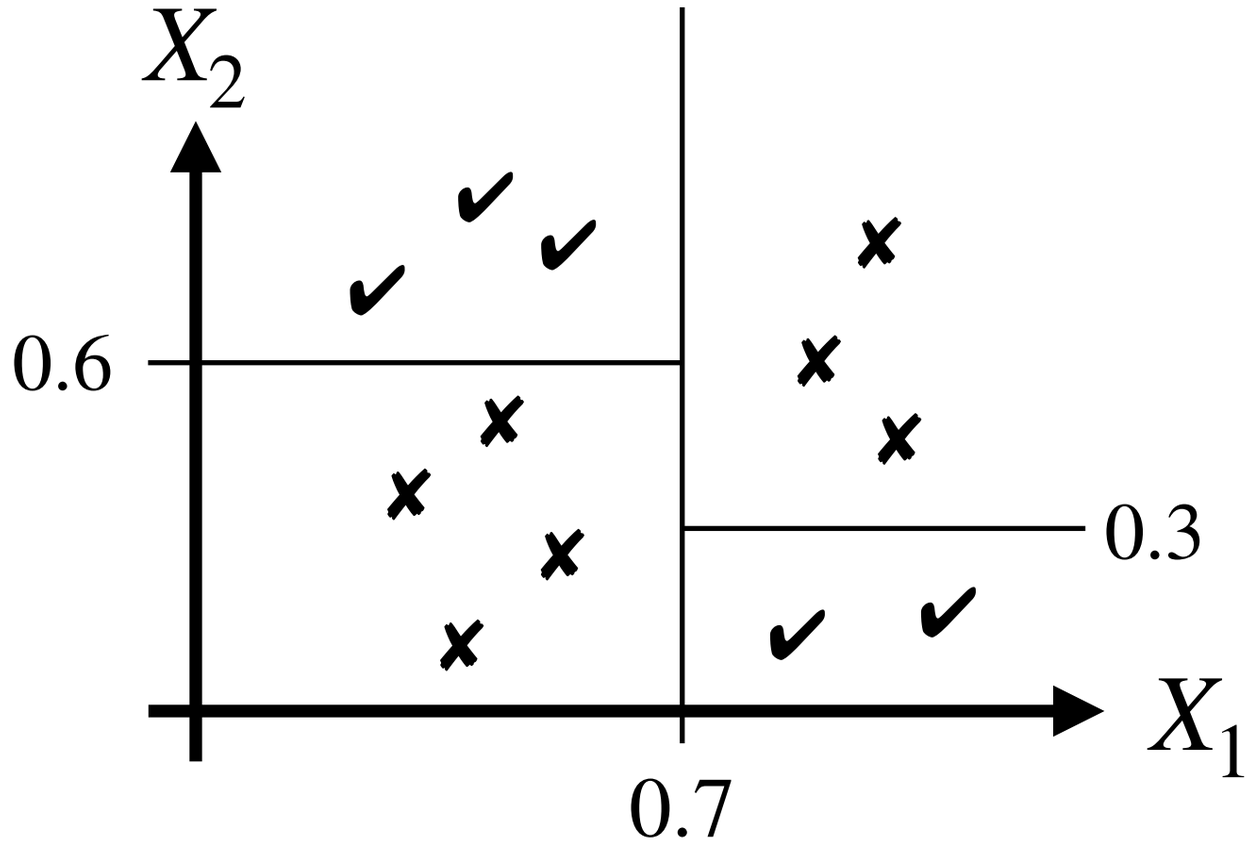


Decision on features

- Decision on real-valued features is made by answering:

  “feature X is less than real value R?”





Prediction

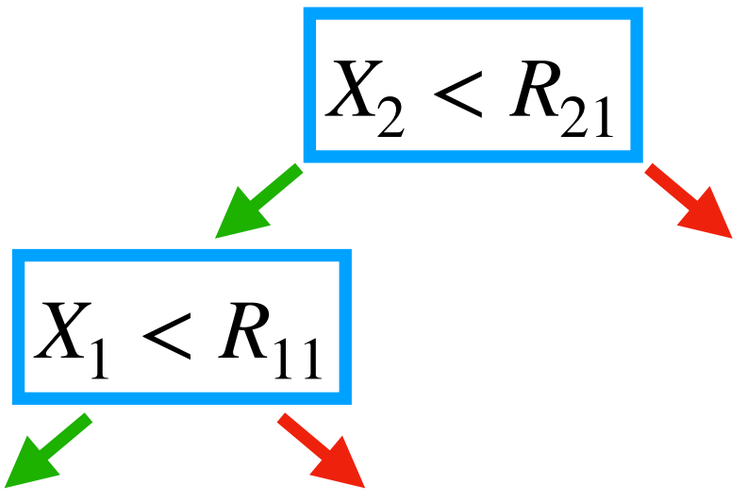
- Estimate an output using the data points in the leaf node found by answering questions
- Classification
  - The most frequent label among the data points
- Regression
  - The average of the outputs of the data points
  - Linear regression with the outputs

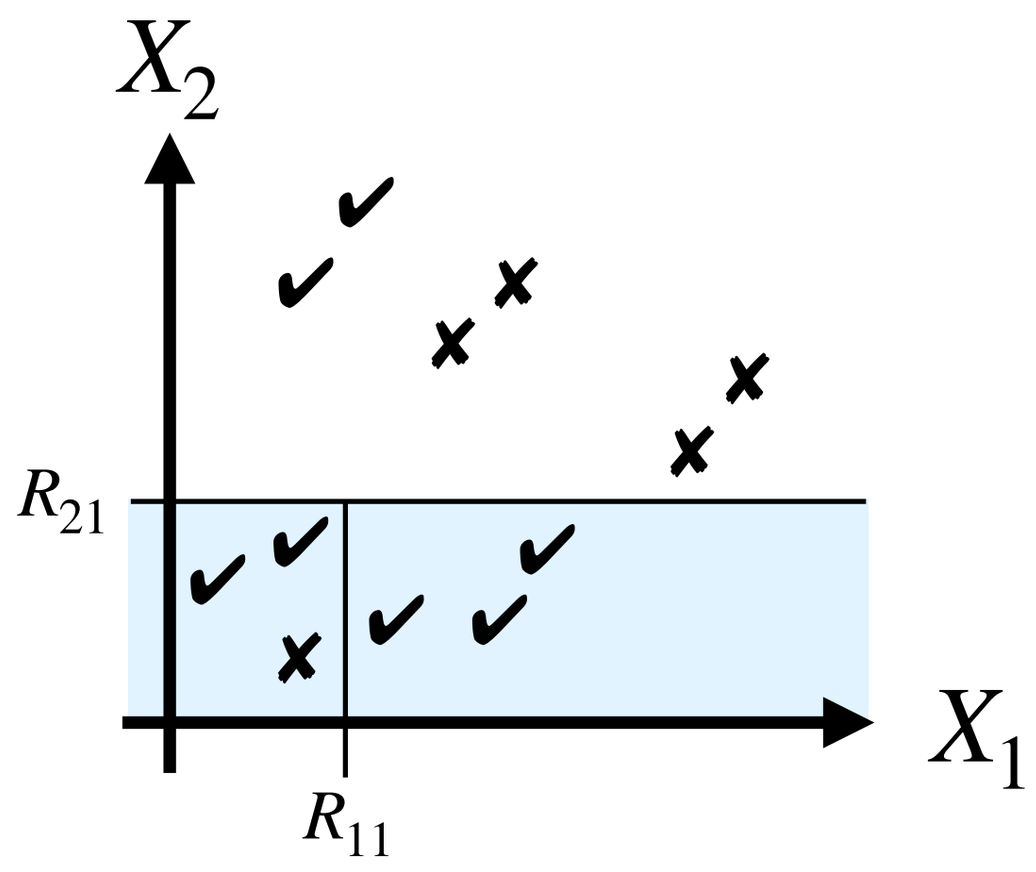
Pros and cons

- 👍 Interpretable
  - Easy to understand why and how the output is predicted
- 👎 Inaccurate prediction (compared with other approaches)

Expressivity

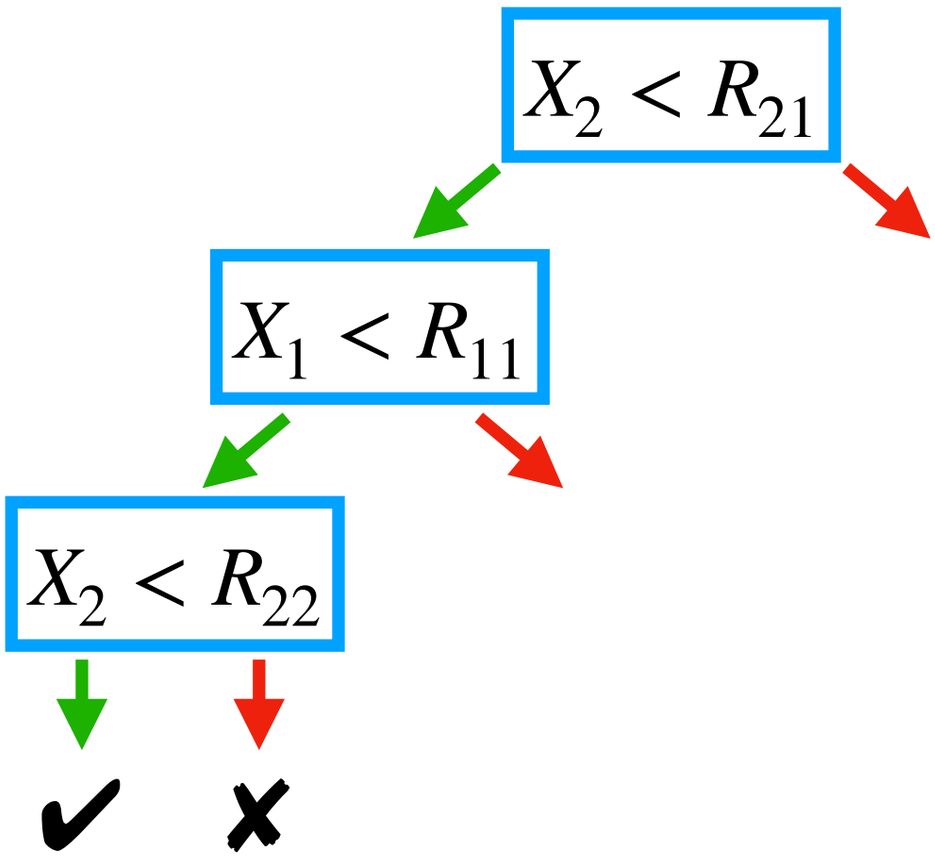
- For any dataset, it is possible to build a model that makes perfectly correct predictions

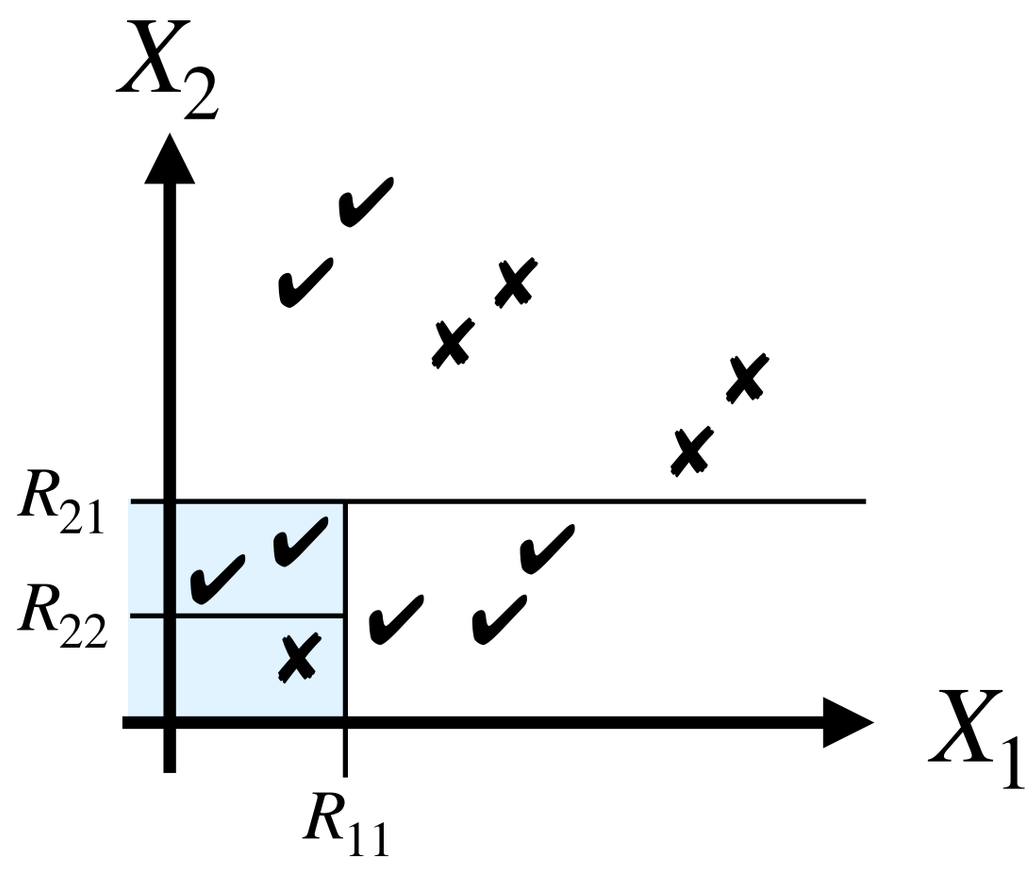




Expressivity

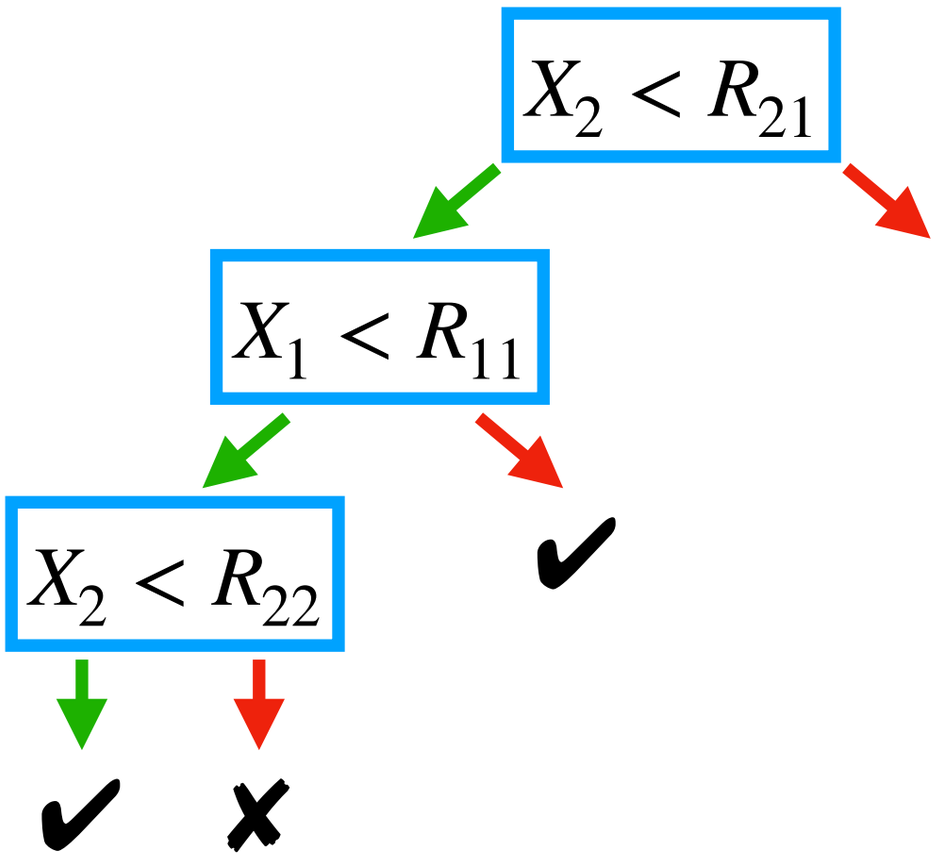
- For any dataset, it is possible to build a model that makes perfectly correct predictions

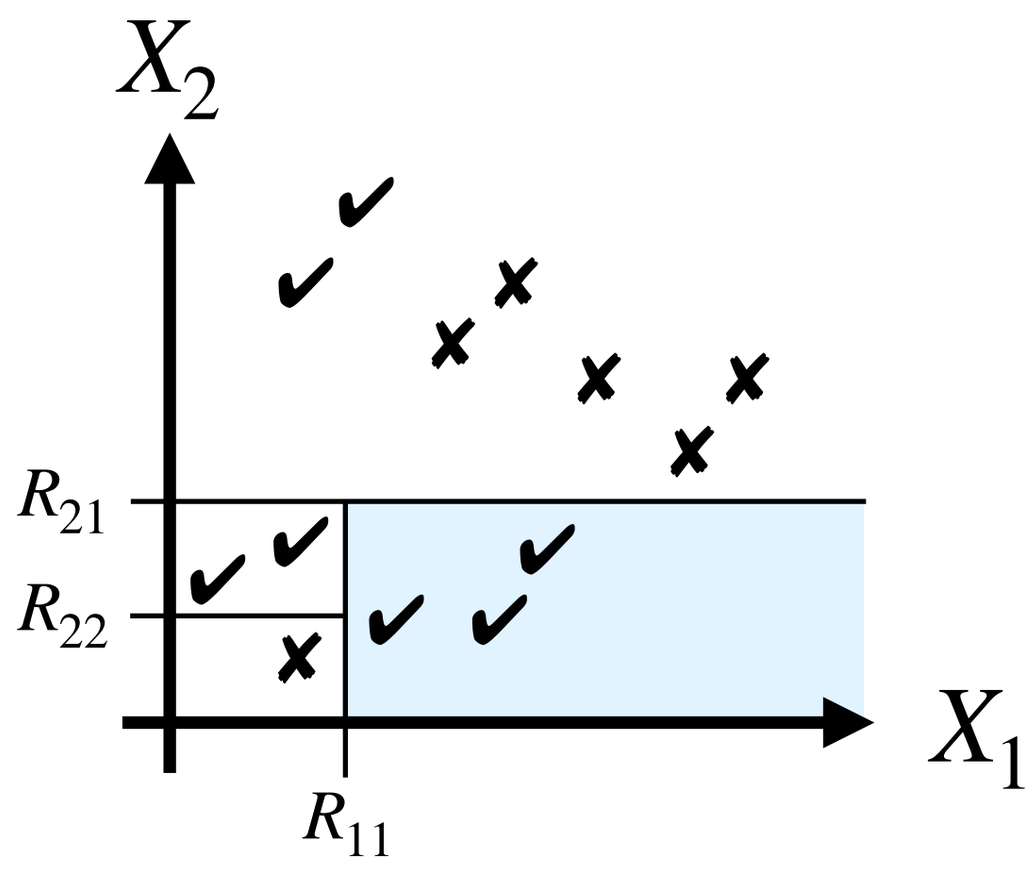




Expressivity

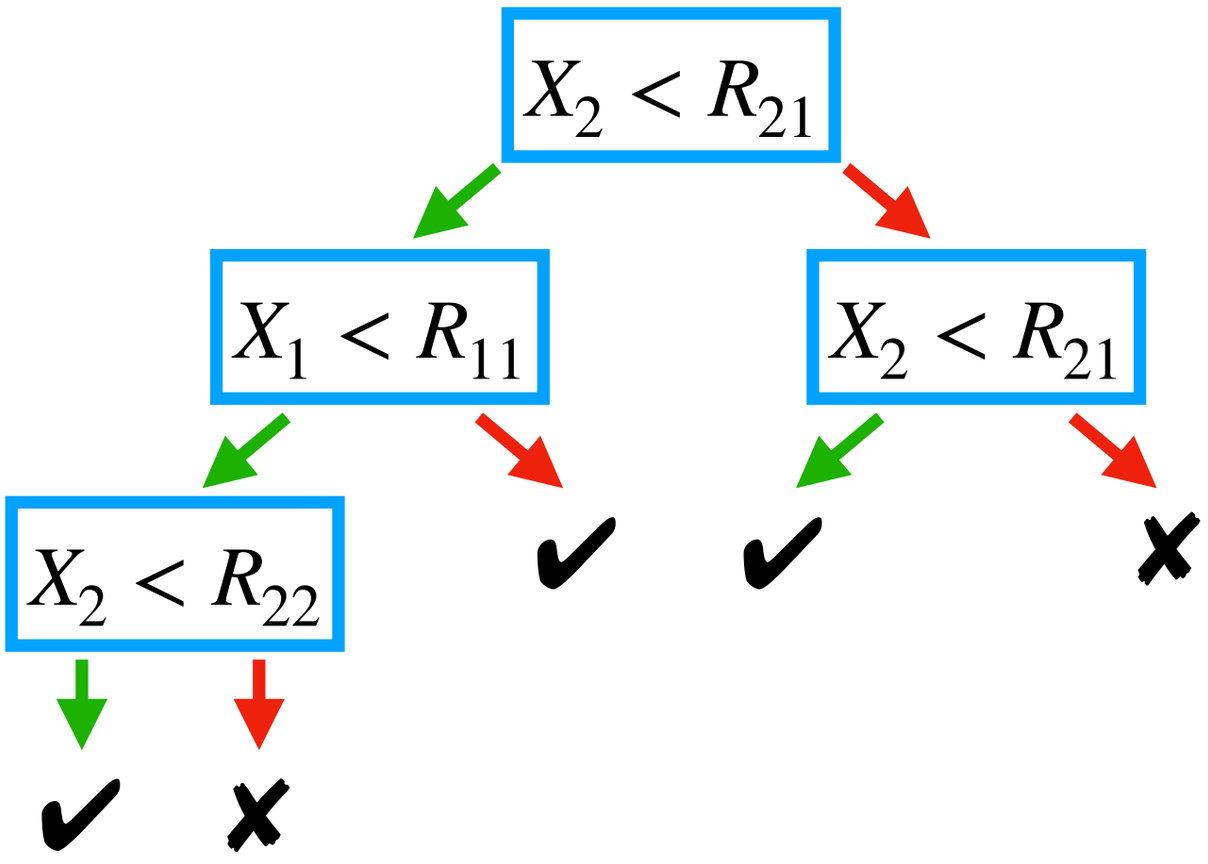
- For any dataset, it is possible to build a model that makes perfectly correct predictions

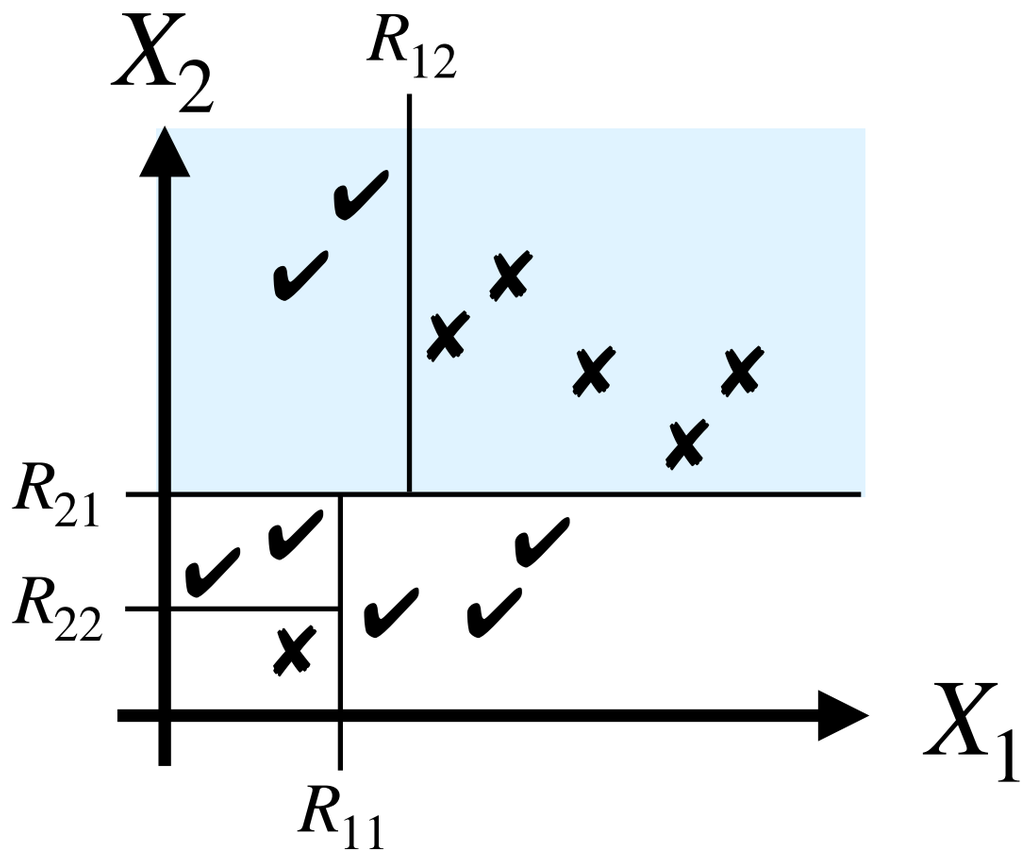




Expressivity

- For any dataset, it is possible to build a model that makes perfectly correct predictions

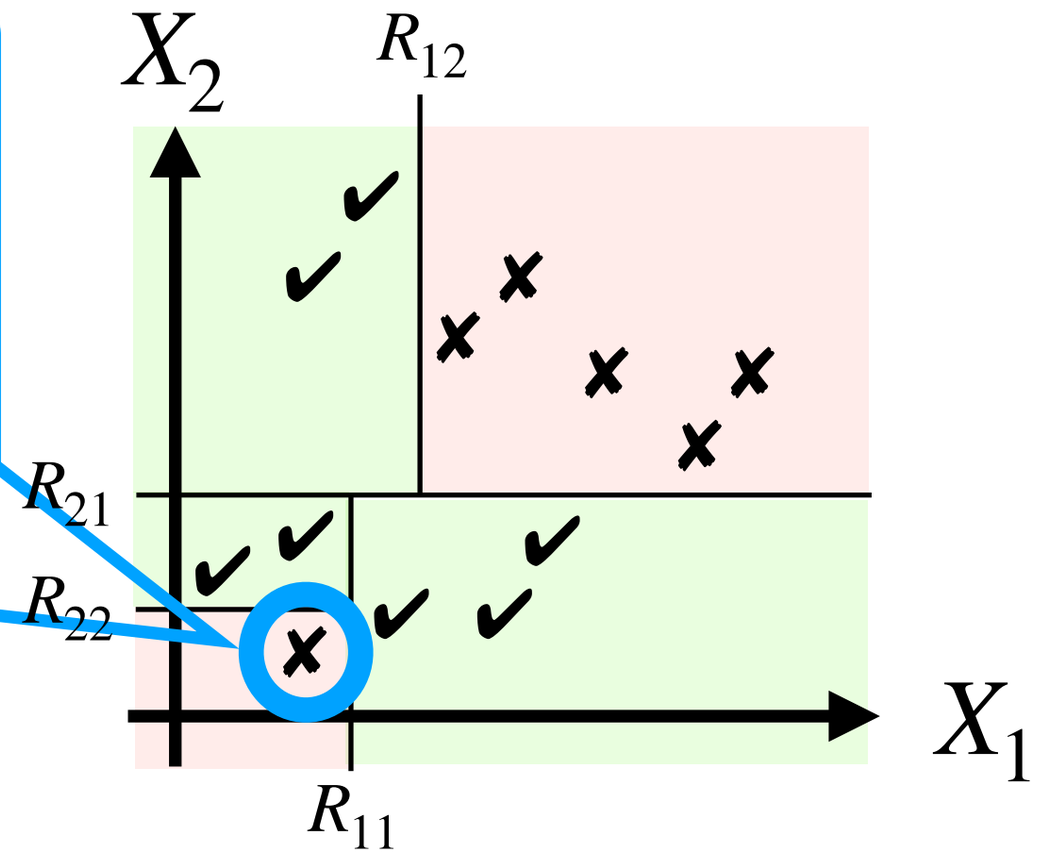




Problem with "perfect" model

- The model is usually ***overfitted***
  - Sensitive to individual training data points
  - Not accurate for unknown data points

If this data point is an outlier, the model missclassifies all the unknown data points of features $X_1 < R_{11}$ and $X_2 < R_{22}$



Overfitting

- Common problem in ML
- Related to the complexity of ML models
  - Complex models are more likely to be overfitted
  - Simple models are more likely to be ***underfitted***
    - Not expressive enough to capture the characteristics of data points

Complexity of decision trees

- Idea: Adjust the # of data points in a leaf node
- If the # is small, the model tends to be sensitive to individual data points
  - Extreme case: each leaf node has a single data point
- If the # is large, the model tends to respect the majority
  - Extreme case: a single leaf node to which all the data points belong

Complexity metrics

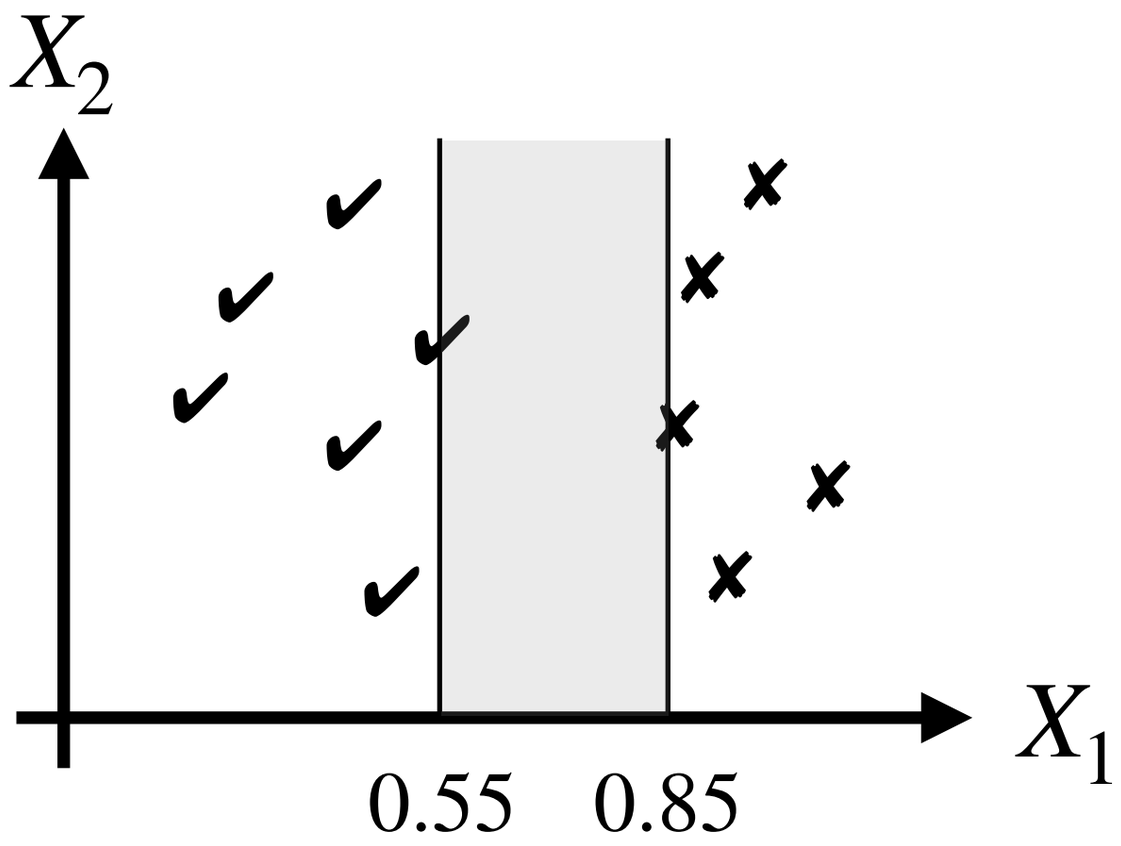
- Multiple metrics to control the complexity
  - Tree depth
  - The # of leave nodes
  - The minimum # of data points belonging to a leaf node

Learning

- Finding feature X and threshold R to construct a question "X is less than R?"
- Criterion: Better to distinguish as many differently labeled data points as possible
  - Less questions make a simpler model
  - Simpler models tend not to be overfitted

Learning

- The simplest tree has a single question that asks about X_1 with $R \in [0.55, 0.85]$



Need a quantitative means to measure how many data points are distinguished to choose best (X, R)

Metric for classification

Entropy

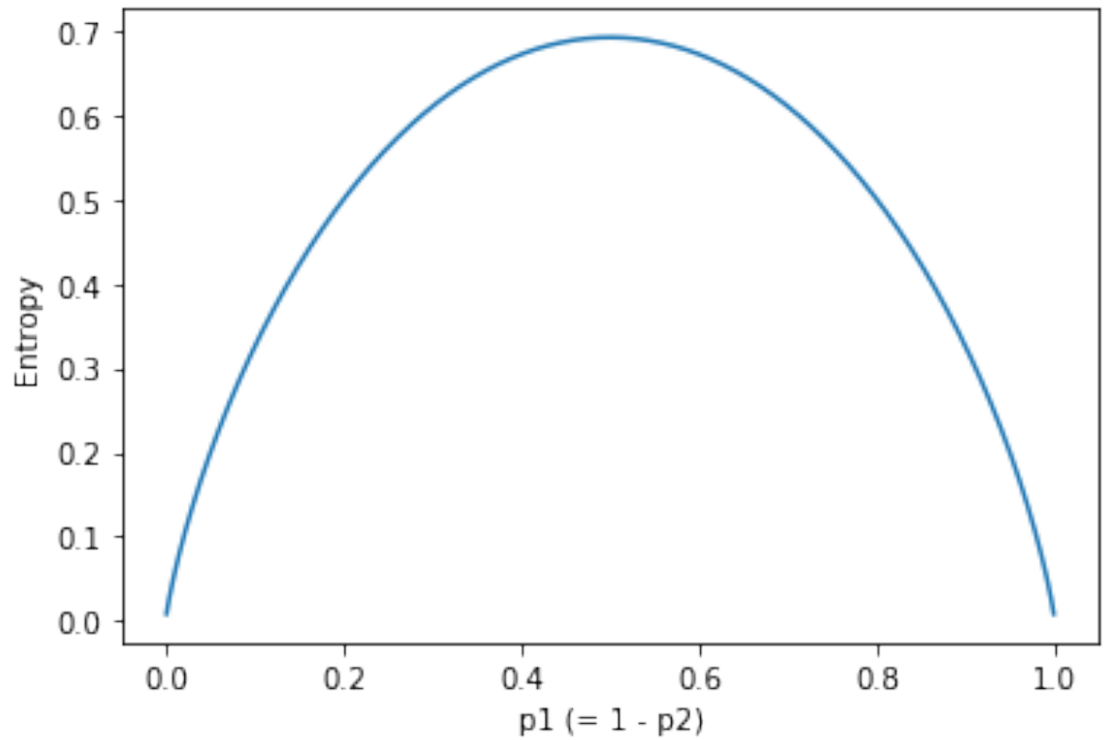
- A general metric to measure uncertainty
- In our setting, uncertainty means how many data points are ***NOT*** distinguished

$$E(P) = -\sum_{p_i \in P} p_i \log p_i$$

P : a set of fractions p_i of data points with labels i

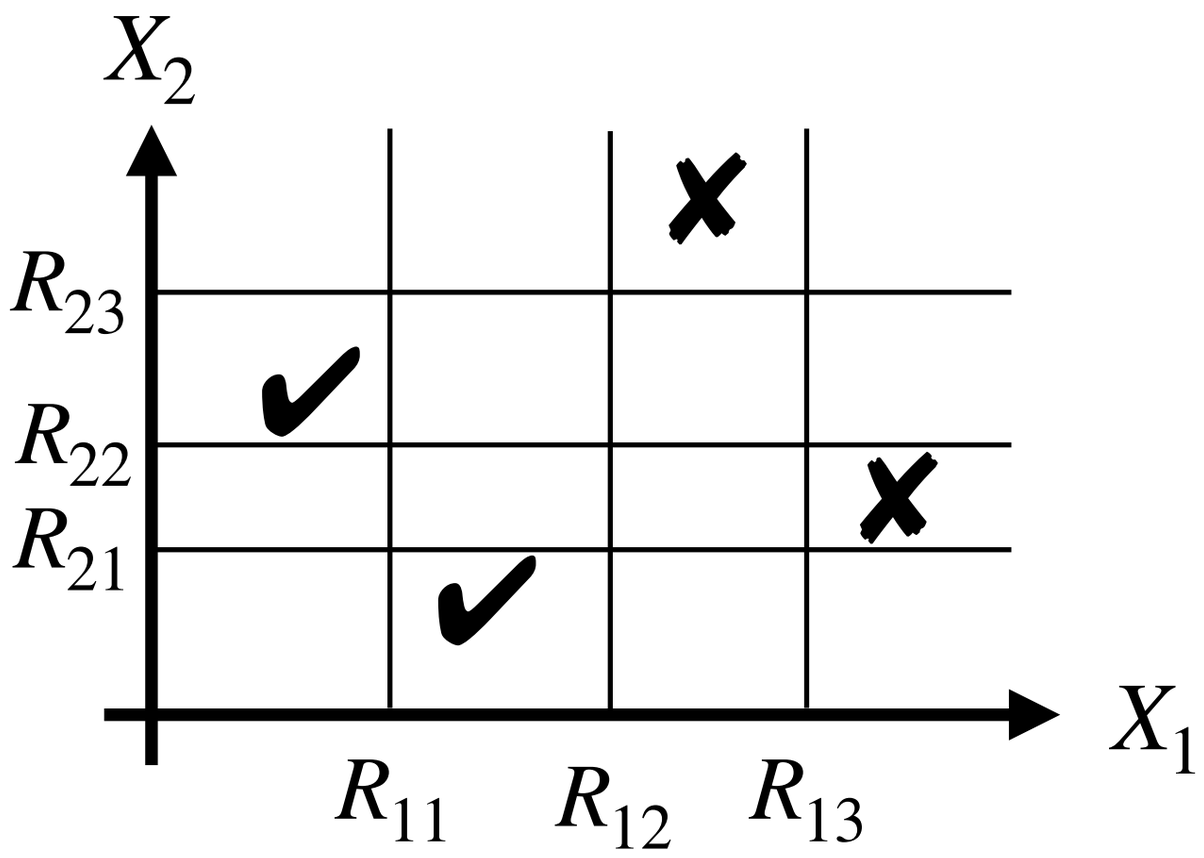
Two-labels entropy

- Highest when differently labeled data points are mixed up ($p_1 = 0.5$)
- Lowest when they are distinguished perfectly ($p_1 \in \{1.0, 0.0\}$)



Learning by example

- Finding $(X, R) \in \{(X_i, R_{ij}) \mid i \in \{1,2\}, j \in \{1,2,3\}\}$ that minimizes the entropy
  - R_{ij} is the mean of X_i-values of data points j & j+1



Problem with entropy

- Entropy is a numerical representation of uncertainty of a **subset** of a training dataset
- It misses the **fraction** of the subset
- We need a means to weight the entropies of subsets according to their fractions

Conditional entropy

$$CE(S, P) = -\frac{S}{T} E(P)$$

T : a training dataset

S : a subset of T

P : a set of fractions p_i of data points with labels i in S

Goal of learning: finding (X, R) that minimizes

$$CE(S_y, P_y) + CE(S_f, P_f)$$

S_y, S_f: subsets of data points that answer yes/no to the question

P_y, P_f: sets of fractions for S_y and S_f

Metric for regression

Variance reduction

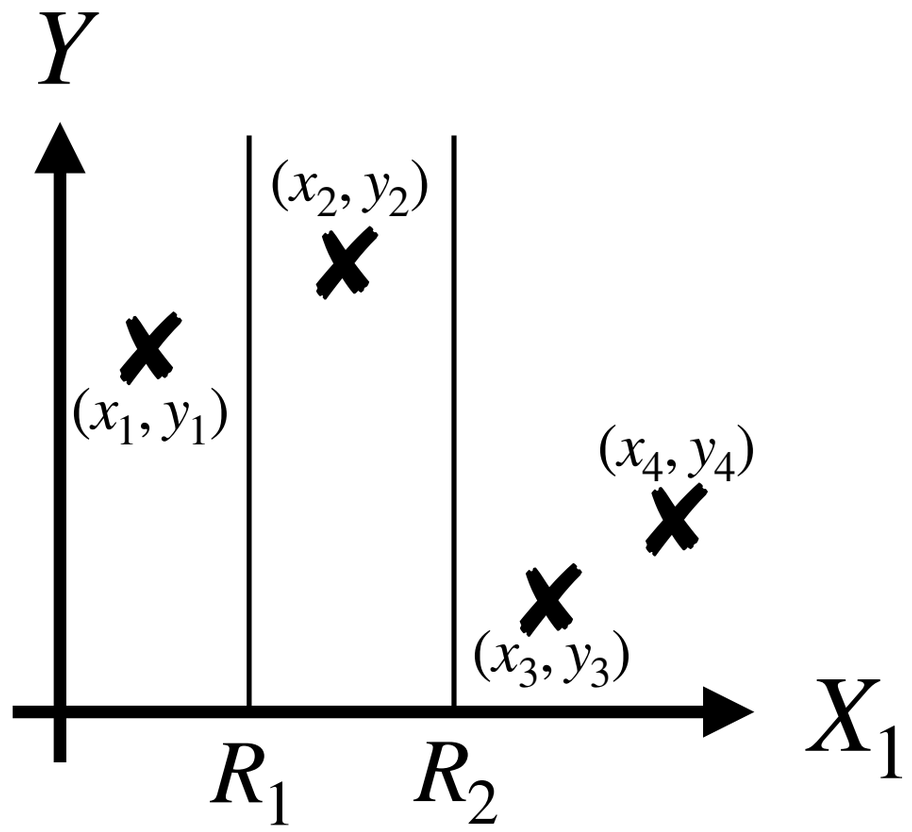
$$V(S_y, S_n) = \sum_{x \in S_y} \sum_{y \in S_y} (x - y)^2 + \sum_{x \in S_n} \sum_{y \in S_n} (x - y)^2$$

S_y, S_n are the sets of the outputs of data points that answer yes and no for a question

- Measures the degrees of “scatter” of the regions that are the splitting results by the question
- The goal of learning is to find (X, R) that minimizes the variance reduction

Metric by example

- The split by R_2 makes the variance reduction smaller than, e.g., R_1



Agenda

1. Decision trees
2. **Random forests**

Random forests

- Constructing a single ML model from a collection of decision trees
  - An ***ensemble*** learning algorithm
  - Improving the performance of a single decision tree

- Remark: other ensemble algorithms
  - Boosting, cascading, etc.

Prediction

- Classification
  - By voting
- Regression
  - The mean of the outputs of decision trees

Trees in a forest

- Decision trees in a random forest should be
  - Similar
    - A tree should produce a similar prediction to many of the other trees
  - (Slightly) different
    - A tree should produce a different prediction from some of the other trees
- The variation of trees are introduced by randomness

Randomness

- Two kinds of randomness in training
  - A dataset used to train each tree is sampled at random from a given entire dataset
  - For each question, asked feature candidates are selected at random
- Similar but slightly different trees are built by this randomness

Pros and cons

- 👍 Accurate prediction
- 👍 Scalability

- 👎 Interpretation of the model is more difficult than that of decision trees