TECHNISCHE UNIVERSITÄT DRESDEN

Dresden Database Systems Group

Classification

Programming for Data Science

Classification Problem

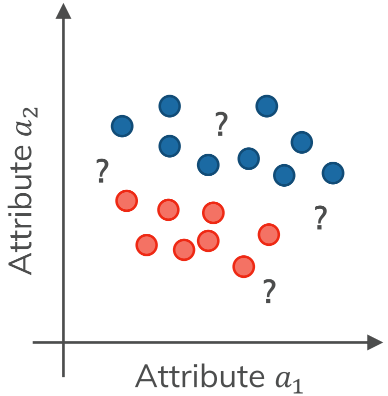
Dresden Database Systems Group

Given

- A d-dimensional space D with attributes a_i, $(i = 1, \dots, d)$
- A set $C = \{c_1, \dots, c_k\}$ of k different class labels c_j, $(j = 1, \dots, k)$
- A set $X \subseteq D$ of n observations $X = \{x_1, \dots, x_n\}$ with known class labels where $x_l = (a_1, \dots, a_d), (l = 1, \dots, n)$

Goal

- Labeling all observations $D \setminus X$ whose class is unknown
- Better understand the data

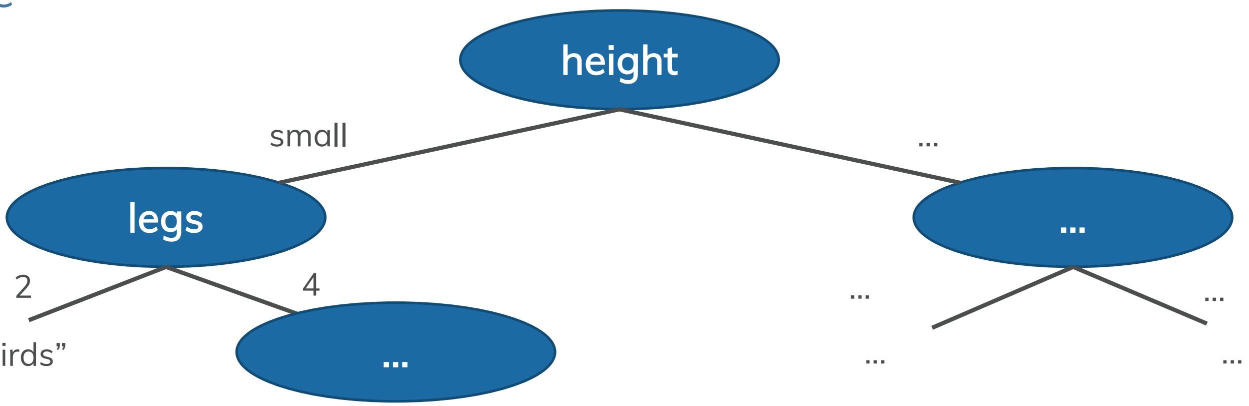


Decision Tree Classifier

Training set

Index	1	2	3	4	…
Height *h*	small	small	tall	small	…
Legs *l*	0	2	2	4	…
Class C	Fish	Bird	Human	Cat	…

Decision Tree



Decision Tree Classifier (2)

Decision Tree

- Flowchart-like tree structure
- Inner nodes are test attributes
- Leaf nodes represent class label and frequency
- Different paths (different attributes and values) to different class labels

Construction of Decision Tree

- Construction
  - The training set is linked with the root node
  - Partitioning of training set with respect to test attributes
- Pruning
  - Identification and pruning of noise and outliers

Prediction of Decision Tree

- New items are classified by tree traversal
- Class label is determined by leaf node

Decision Tree Classifier (3)

Partition Algorithm (Greedy-like)

- Construction of decision tree: top-down, recursive, divide-and-conquer
- Supports categorical and continuous attributes
- Initially, training set is linked to root node
- Recursive partitioning of training set on each node
  - Passing disjoint subsets of training set to child node
- Selection of test attributes and split points per inner node
  - Usage of heuristics or statistical measures, e. g., information gain

Termination

- All observations of training set belong to a class
- There are no attributes that can be used for partitioning (class label is chosen by majority vote)

Decision Tree Classifier (4)

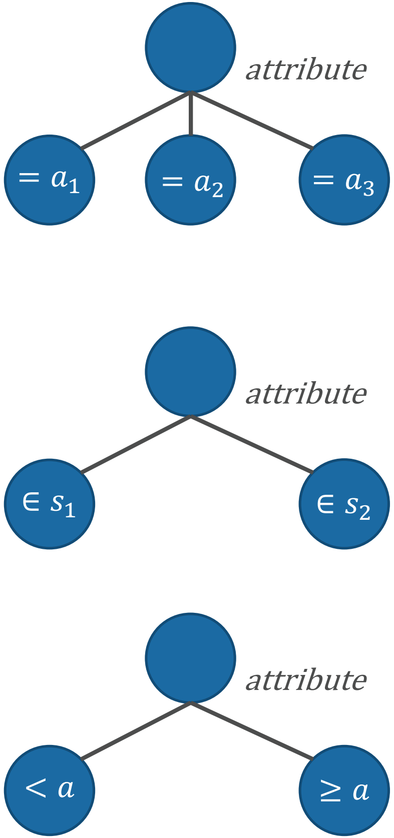
Dresden Database Systems Group

Categorical Attributes

- Split based on equality constraint
  $attribute = value$
- Split based on subset constraint
  $attribute \in set$
- Other alternatives

Continuous Attributes

- Split based on inequality constraint
  $attribute < value$
- Definition of interval with endpoints allows for many decision options
- Other alternatives



Decision Tree Classifier (5)

Accuracy of Splits for Prediction

- Let T be the training set
- Disjoint, complete partitioning of $T = T_1, T_2, \ldots, T_m$
- Relative frequency p_j of class c_j in T_i
- Goal
  - Find a measure that describes the heterogeneity of the test set with respect to their class attributes
  - A split of $T = T_1, T_2, \ldots, T_m$ shall minimize the heterogeneity of each partition T_i

Common Measures

- Information Gain
  - Used for categorical attributes
  - Modifications for continuous attributes exist
- Gini Index (IBM IntelligentMiner)
  - Measure of inequality
  - Used for continuous attributes
  - Modifications for categorical attributes exist

Information Gain

Basics

- *Self-information* represents a unit of information for a given event
- An event with probability p has the self-information I:

$$I(p) = -\log_2 p$$

- The entropy is the expected information of the set T with probability p_i of item i:

$$H(T) = \sum_{i=1}^{k} p_i \cdot I(p_i) = -\sum_{i=1}^{k} p_i \cdot \log_2 p_i$$

Application on Decision Trees

- There are k classes c_i with frequency p_i
- $H(T) = 1$ if all classes c_i have the same probability $p_i = 1/k$
- $H(T) = 0$ if one class c_i has $p_i = 1$
- Entropy refers to uncertainty

Information Gain (2)

Dresden Database Systems Group

Definition

- Attribute A realizes partitioning T in $T_1, T_2, \dots, T_m$
- The *information gain* of A with respect to T is

$$\text{informationgain}(T, A) := H(T) - \sum_{i=1}^{m} \frac{|T_i|}{|T|} \cdot H(T_i)$$

- The expected value of the information gain is the reduction in the entropy of T by learning from attribute A

Algorithms

- Iterative Dichotomiser 3 (ID3)
- C4.5 as an extension of ID3

Decision Trees (ID3)

Generate the decision tree for the following classification problem.

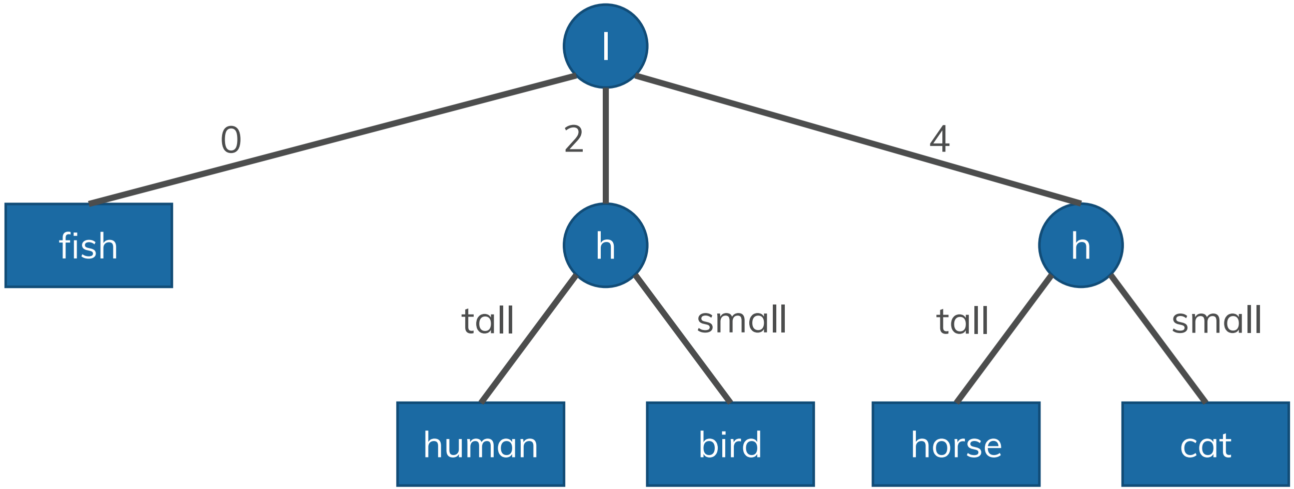
Index	1	2	3	4	5	6	7	8
Height *h*	small	small	tall	small	tall	tall	small	tall
Legs *l*	0	2	2	4	4	2	4	2
Class C	Fish	Bird	Human	Cat	Horse	Human	Cat	Human

- Attributes: $a_1 = h =$ height, $a_2 = l =$ legs
- Classes creatures $C = \{c_{fish}, c_{bird}, c_{human}, c_{cat}, c_{horse}\}$

$$\text{informationgain}(T, A) \coloneqq H(T) - \sum_{i=1}^{m} \frac{|T_i|}{|T|} \cdot H(T_i)$$

$$H(T) = -\sum_{i=1}^{k} p_i \cdot \log_2 p_i$$

Decision Tree (ID3)



Task

Dresden Database Systems Group

Step 0

- You will get a csv file from us. Load it in your language/environment.
- Explore the data in it. Identify the input data X and the labels.

Step 1

- Implement an ID3 decision tree*.

Step 2

- Use your decision tree to classify: rainy forecast, hot temperature, high humidity, strong wind

*use your own implementation

Package suggestions

R

- (data.table)

python3

- numpy
- pandas

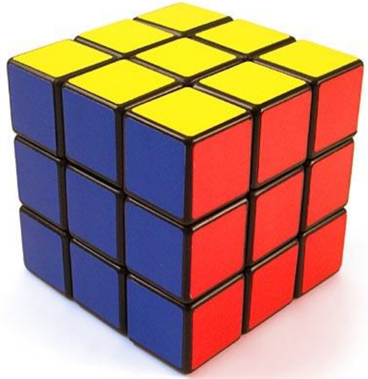
Exercise Appointment

We compare and discuss the results

- Tuesday, 19.11.2019,
- Consultation: 14.11.2019,
- Please prepare your solutions! Send us your code!

If you have questions, please mail us:

claudio.hartmann@tu-dresden.de Orga + Code
lucas.woltmann@tu-dresden.de Tasks + Python
lars.kegel@tu-dresden.de Tasks + R



Decision Tree (ID3)