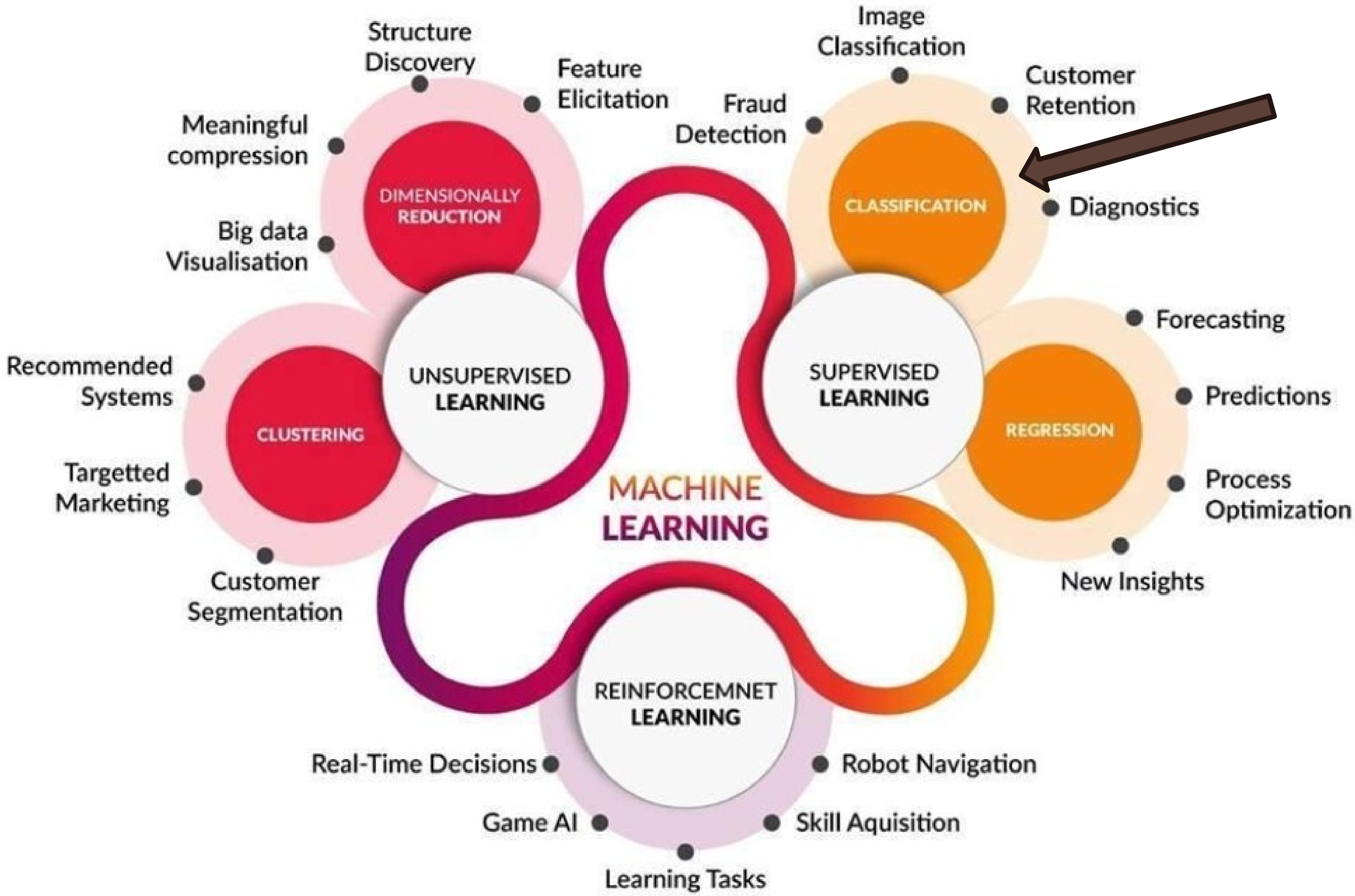
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

CPGEI

Prof. Heitor Silvério Lopes
Prof. Thiago H. Silva

Data Mining & Knowledge Discovery

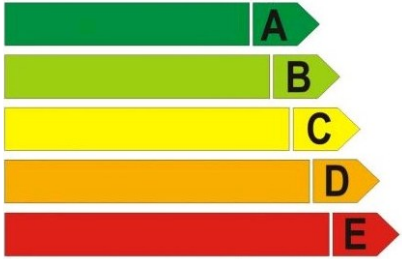
Class 2a – Data Classification (Trees)
2025


Structure Discovery
Feature Elicitation
Meaningful compression
DIMENSIONALLY REDUCTION
Big data Visualisation
Recommended Systems
CLUSTERING
Targetted Marketing
Customer Segmentation
UNSUPERVISED LEARNING
Image Classification
Customer Retention
Fraud Detection
CLASSIFICATION
Diagnostics
SUPERVISED LEARNING
Forecasting
Predictions
REGRESSION
Process Optimization
New Insights
MACHINE LEARNING
REINFORCEMNET LEARNING
Real-Time Decisions
Robot Navigation
Game AI
Skill Aquisition
Learning Tasks

What is data classification ?

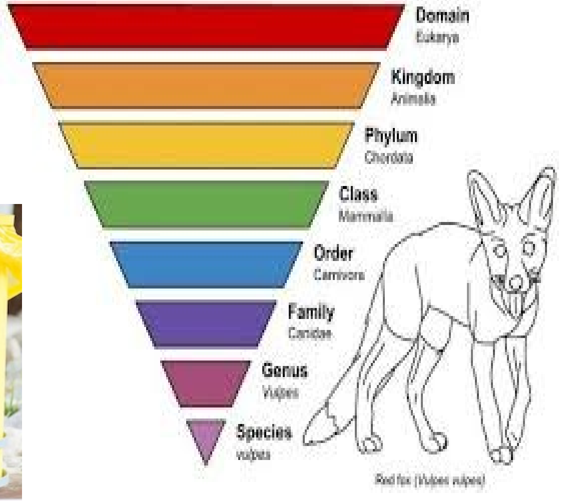
- It is a basic task that humans do almost intuitively
- It is present in our daily tasks, Science and Technology

Power consumption classes



Selective garbage collection

Taxonomy of living beings



Tasks x Methods in Data Mining

Tasks	Methods
Classification	Decision trees (C4.5), Cassification rules, k-nearest-neighboors, Random forest, Support vector machine, Bayesian classifier, Neural network, Adaboost
Association Rules	Apriori, FP-growth, Eclat, Zigzag
Regression	Linear Regression, Polynomial regression, Logistic regression
Feature Selection & Dimensionality Reduction	Principal component analysis (PCA), Chi-square, Entropy, Information gain
Clustering	K-means, Kohonen’s self-organized map, Density-based scan, Hierarchical grouping, t-SNE
Data visualization *	Silhouette plot, scatter plot, heatmap, box plot, clusters, t-SNE

Important definitions

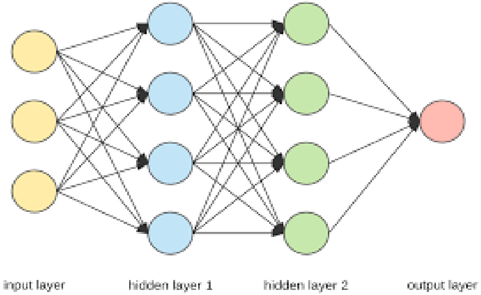
- Classification is the task of finding a computational model:
  - Predictive model
  - Descriptive model
- The construction of these models require a training dataset
- The quality of the model is checked by means of a testing dataset
- Each instance of the training/testing datasets is composed of atributes (features):
  - Predictive atributes
  - Target atribute (or “class”)

Descriptive X Predictive models for classification

- Descriptive model: it is aimed to sumarize, in human-understandable way, the relevant features of a dataset to distinguish possible classes of objects

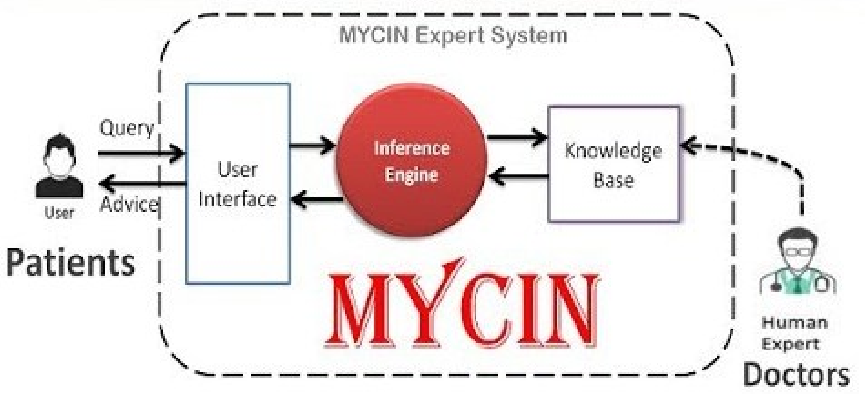
> Extraction of structured knowledge which is useful, previously unknown, non-trivial, **humanly comprehensible**, from large amounts of data (Fayyad et al., 1996)

- Predictive model: it is aimed to create a (not necessarily understandable) model that can be used to classify unknown data

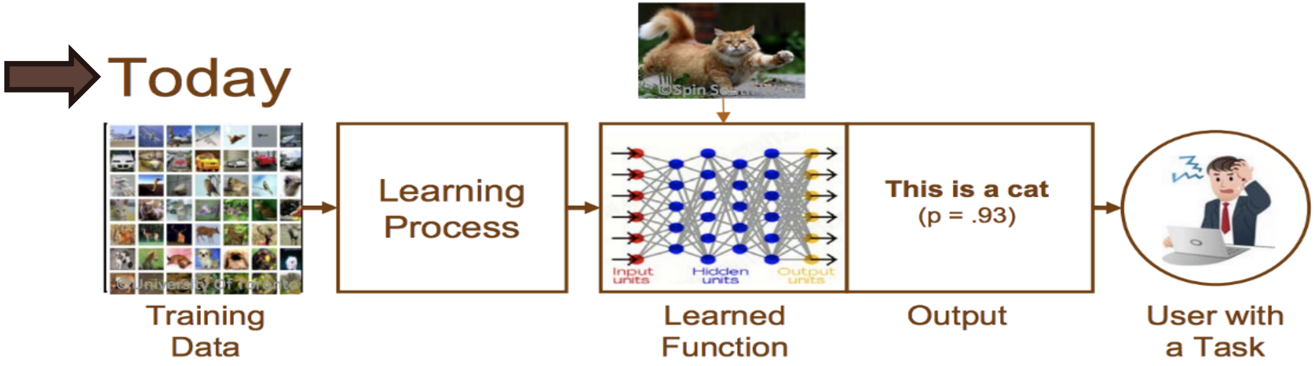


The importance of descriptive models

- In the early days of A.I. (~1970), rule-based methods were developed to support medical diagnosis.
  - Mycin was developed at the Stanford University to identify bacteria, recommend antibiotics adjusted to patients’ conditions
- Their acceptance was very limited due to the low capacity to really “explain” their decisions



Explainable Artificial Intelligence



Today

Training Data

Learning Process

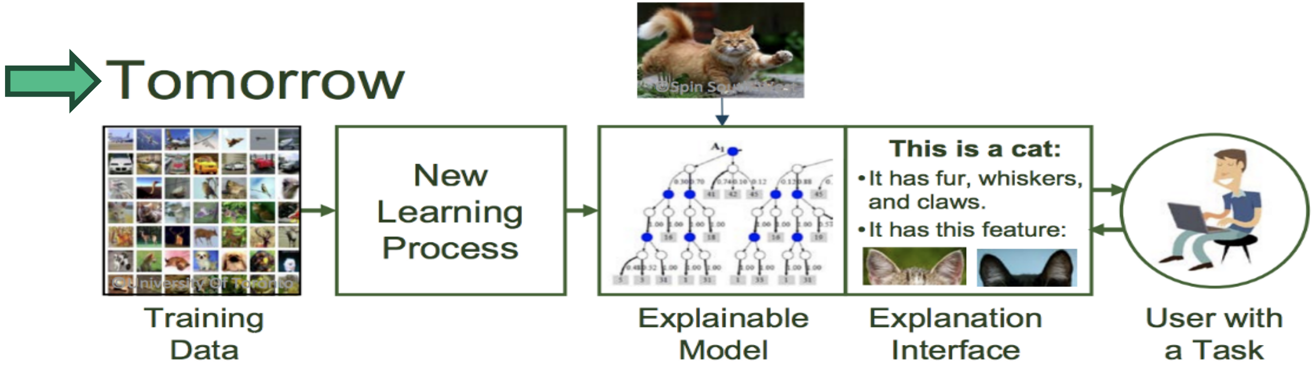
Learned Function

This is a cat
(p = .93)

Output

User with a Task

- Why did you do that?
- Why not something else?
- When do you succeed?
- When do you fail?
- When can I trust you?
- How do I correct an error?



Tomorrow

Training Data

New Learning Process

Explainable Model

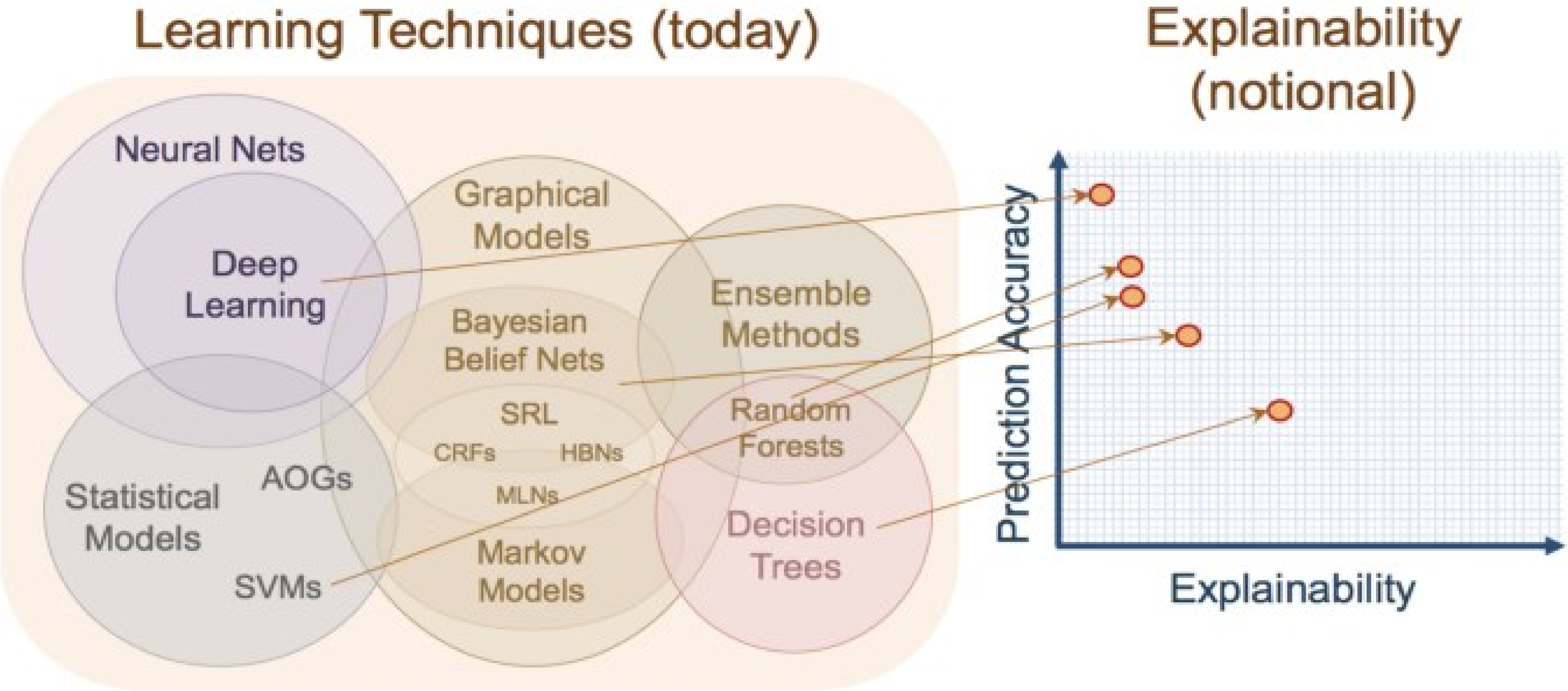
This is a cat:
- It has fur, whiskers, and claws.
- It has this feature:

Explanation Interface

User with a Task

- I understand why
- I understand why not
- I know when you'll succeed
- I know when you'll fail
- I know when to trust you
- I know why you erred

“Explainability” of the main data classification methods



First steps to explain anomalies in videos

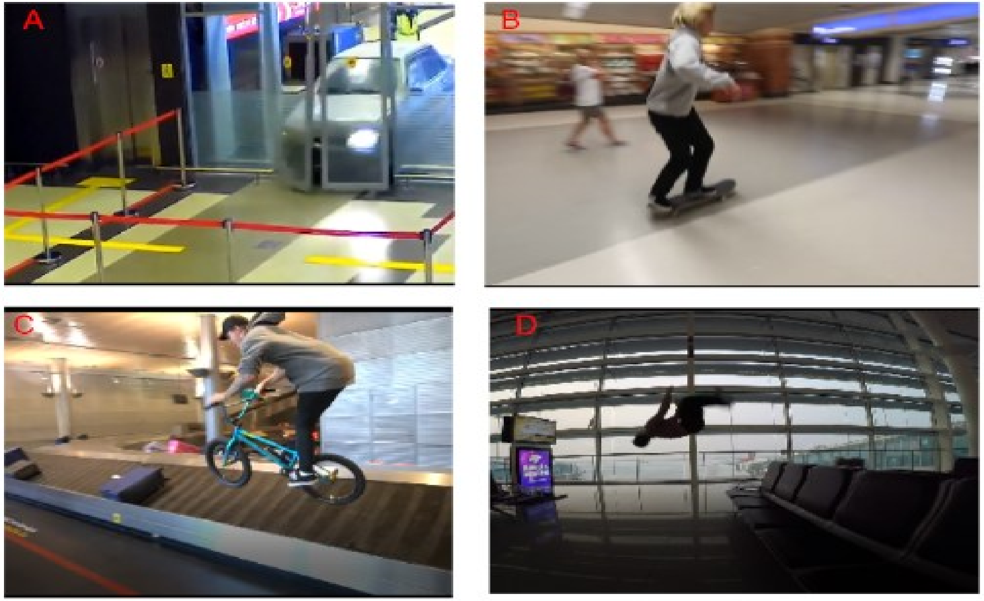


Figure 3. Examples of anomalies in the dataset. A) a person is driving a car. B) A man is skateboarding. C) A man is riding a bike. D) A person is jumping and vaulting over the seats.

- First, detect an anomaly
- Then, describe it

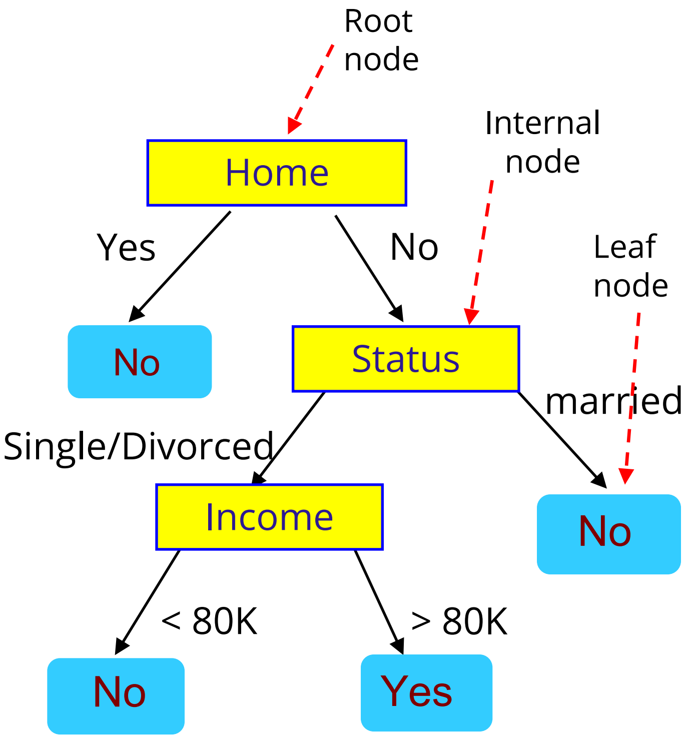
Inácio, A.S., Teixeira, R.M., Lopes, H.S., Explainable anomaly detection in videos based on the description of atomic actions. *Proc. XV Brazilian Congress on Computational Intelligence (CBIC)*, 2021

Some methods for data classification

- Descriptive modeling:
  - Methods based on decision trees: **C4.5**, Random Forest, Hoeffding Tree...
  - Methods based on rules: ZeroRule, OneRule, Ripper, Ridor...
- Predictive modeling:
  - Instance-based or Bayesian network-based: Ibk, Naïve Bayes, HMM
  - Other: Neural Networks (MLP, RBF), Support-Vector Machines (SVM), cluster-based classification

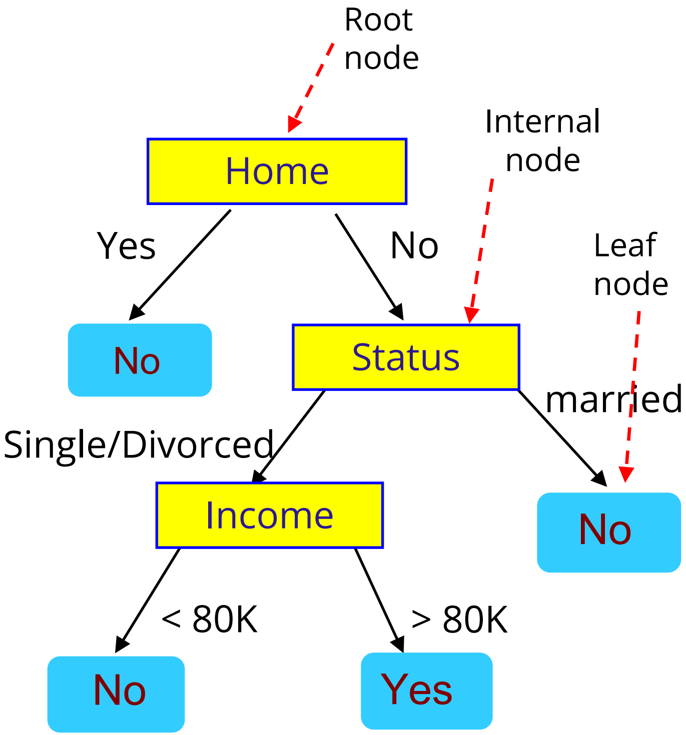
Induction of decision trees from data

- A decision tree is a directed graph with:
    - Nodes
        - Each node representa an atribute
        - The initial node is the “root”
        - There are internal nodes and leaf nodes (terminals)
    - Arcs:
        - Arcs connect the root node to internal nodes until the leaf nodes
        - Each node represent a specific value for an atribute
- A learning algorithm induces a decision tree using a training dataset
- The generated tree is used to classify unknown instances

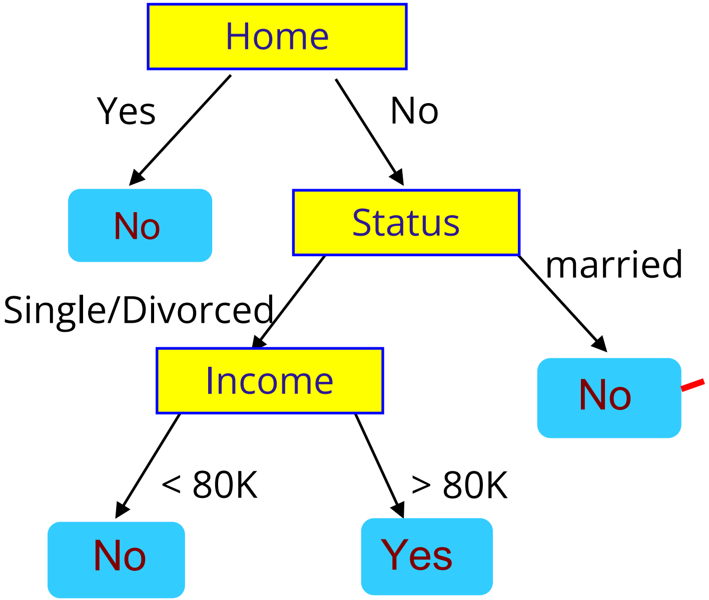


First step: model induction

	Categorical	Categorical	Continuous	Class
#	Own home?	Marital status	Year income (1000 R$)	Insolvent ?
1	Yes	Single	125	No
2	No	Married	100	No
3	No	Single	70	No
4	Yes	Married	120	No
5	No	Divorced	95	Yes
6	No	Married	60	No
7	Yes	Divorced	220	No
8	No	Single	85	Yes
9	No	Married	75	No
10	No	Single	90	Yes



Second step: applying the model to the test set



#	Own home?	Marital status	Year income (1000 R$)	Insolvent ?
1	No	Married	80	???

How the trees are induced

- Attributes should be divided into the tree's branches
  - Depends upon the type of the atribute: nominal (binary or multiple), continuous
  - The Division can be binary or multiple

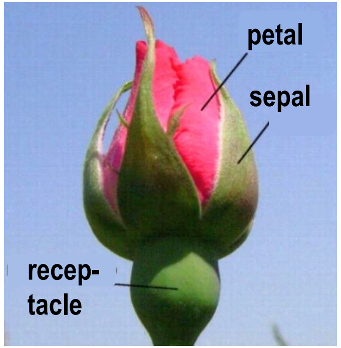
- How one can determine the best division at each branch?
  - Using a "measure of impurity" of the classes after each Branch
  - Such measure needs to be **minimized** at each step
  - Measures:
    - Gini index
    - Classification error
    - Entropy
    - Information gain

$$Entropy(t) = -\sum_{j} p(j|t) \log p\,(j|t)$$

$$InformationGaino_{split} = Entropy(p) - \left(\sum_{i=1}^{k} \frac{n_i}{n} Entropy(i)\right)$$

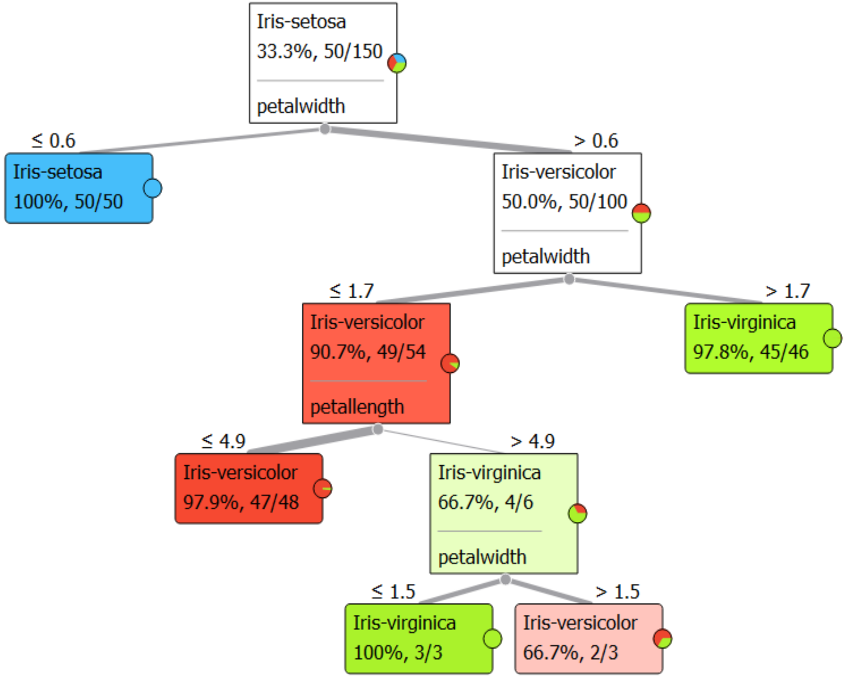
Case study: the iris dataset

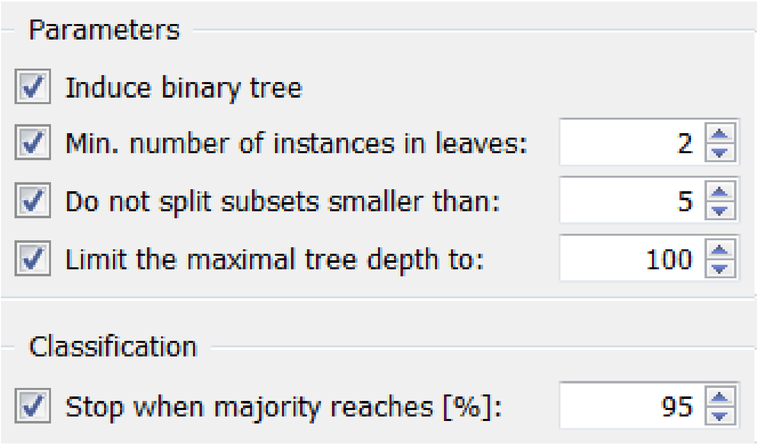
- Original data collected by R.A.Fisher. One of the most used dataset in the classification data literature
- Objective: find the class of each flower, based on measures of length and width of petals and sepals
- Details:
    - Instances: 150 (50 for each class)
    - Predictor atributes: 4 (petal width, petal length, sepal width, sepal length)
    - Meta atribute (Class): Iris setosa, Iris versicolor, Iris virginica



Case study #1: the iris dataset

- Induced decision tree:





Parameters

☑ Induce binary tree

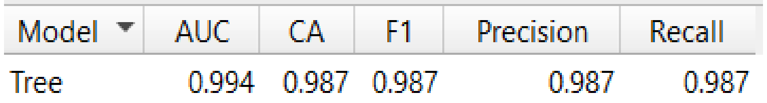
☑ Min. number of instances in leaves: 2

☑ Do not split subsets smaller than: 5

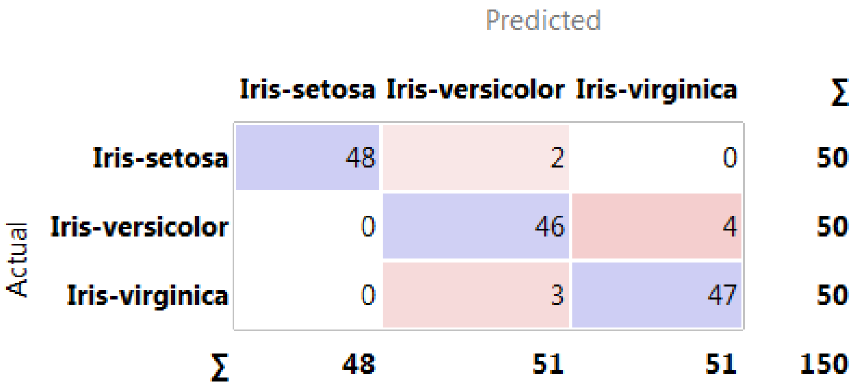
☑ Limit the maximal tree depth to: 100

Classification

☑ Stop when majority reaches [%]: 95



Model	AUC	CA	F1	Precision	Recall
Tree	0.994	0.987	0.987	0.987	0.987



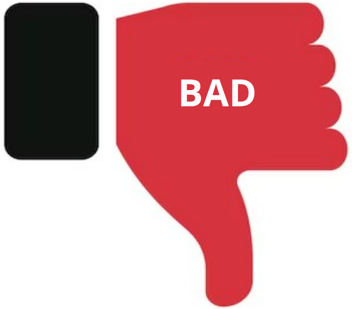
Predicted

		Iris-setosa	Iris-versicolor	Iris-virginica	∑
Actual	Iris-setosa	48	2	0	50
	Iris-versicolor	0	46	4	50
	Iris-virginica	0	3	47	50
	∑	48	51	51	150

Advantages and drawback sof decision trees

- Advantages:
  - Visual representation
  - Compact representation of a rule set
  - Fast classification of new instances
  - It can deal with continuous of discrete atributes

- Drawbacks:
  - Irrelevant atributes can negativelly affect the construction of the tree and its understanding
  - Small variations in the data can result in significantly different trees
  - A subtree can be replicated several times
  - Decision trees are not adequate when having many classes



Overfitting and underfitting in decision trees

- Overfitting:
  - The model has a low training error (learns well on the training data)
  - But the model has a high generalization error (on the test data)
  - Overfitting may be due to an insuficiente number of training instances

- Underfitting:
  - When the training and generalization errors are both high
  - The model does not fit the data
  - The model is wrong or too simple

Performance evaluation methods

IMPORTANT

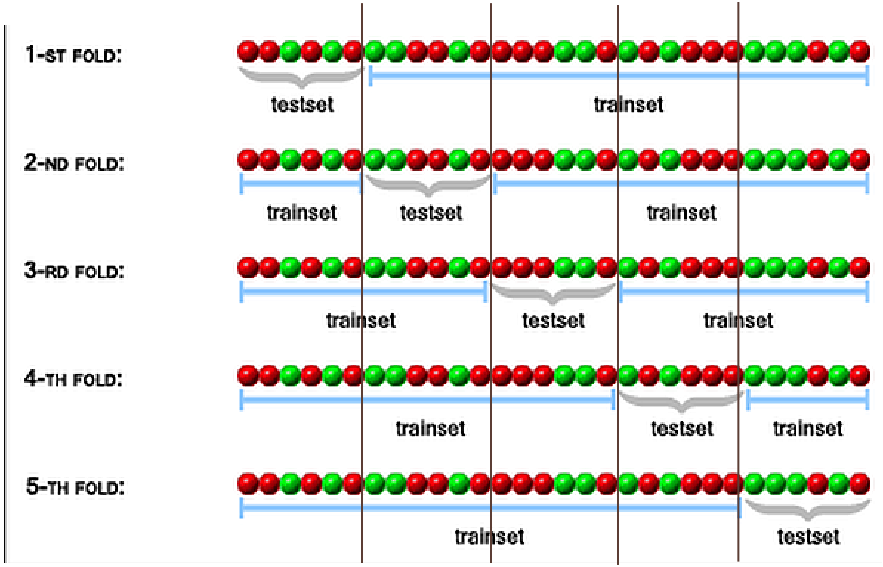
1. Hold-out
    - Data is randomly partitioned into 2 disjoint subsets
    - One of the subsets (2/3 of the tdata) is used for **training** the model
    - The other subset (1/3) is used for **testing** the model
    - This method is suitable When there is a **large** amount of data
    - The quality metric is reported on the test dataset

Performance evaluation methods

IMPORTANT

2. Cross-validation:

- Data is randomly partitioned into ***k*** disjoint subsets
- *K*-fold: training is done with *k-1* partitions and tested with the remanining one
- Repeat with the next partition until all *k* partitions have been tested
- Usually, *k=10* or *k=5* (for small number of instances)
- Report the **mean value** of the quality metric
- Report the **best classifier** as the model



How to evaluate the quality of the predictive model

- Confusion matrix:

		Predicted class	
		yes	no
Real class	yes	TP	FN
	no	FP	TN

TP (true positive)

FN (false negative)

FP (false positive)

TN (true negative)

- The most usual metric is Accuracy (or Hit Rate)

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

Predictive model quality assessment

- Example of Accuracy limitation:
    - Let’s say a problem with 2 classes: C1 has 9990 instances and C2 has 10 instances
    - If a classifier classifies all instances as being from C1, the accuracy is; 9990/10000 = 99,9%
    - The Accuracy is misleading because the model actually does not detect any instance of class C1
- Conclusion: Accuracy is NOT a good metrics when the classes do NOT have the same number of instances, i.e. they are **unbalanced classes**

Predictive model quality assessment

- Sensitivity (TPR – True Positive Rate or Recall), Specificity (True Negative Rate), Precision:

$$Sensitivity = \frac{TP}{TP+FN} \qquad Specificity = \frac{TN}{TN+FP} \qquad Precision = \frac{TP}{TP+FP}$$

- Precision: is the propostion of examples classified as positive that are actually positive.
- Recall (Sensitivity): is the proportion of positive examples that were classified as such (out of all positives, how many were identified)
- When a model has a high recall and low precision, it classifies examples correctly, but includes many false positives (negative examples as positives)

Predictive model quality assessment

- The F1 score is a quality metric that finds the best compromisse between precision and recall
- It is very useful for datasets with imbalanced classes
- F1 is calculated as the harmonic mean between Precision and Recall

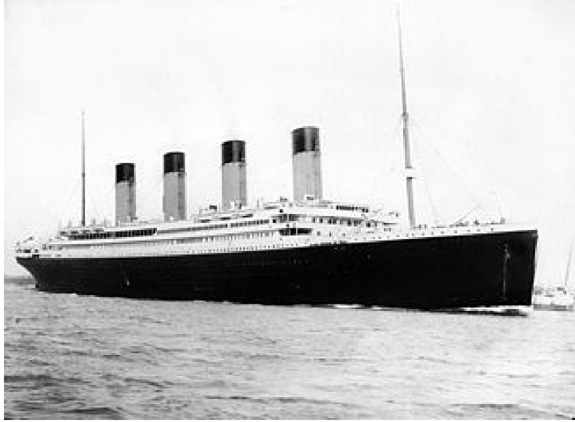
$$F1 = 2.\frac{Precision\,.\,Recall}{Precision + Recall} = \frac{TP}{TP + (FP + FN)/2}$$

Case study #2: Titanic dataset

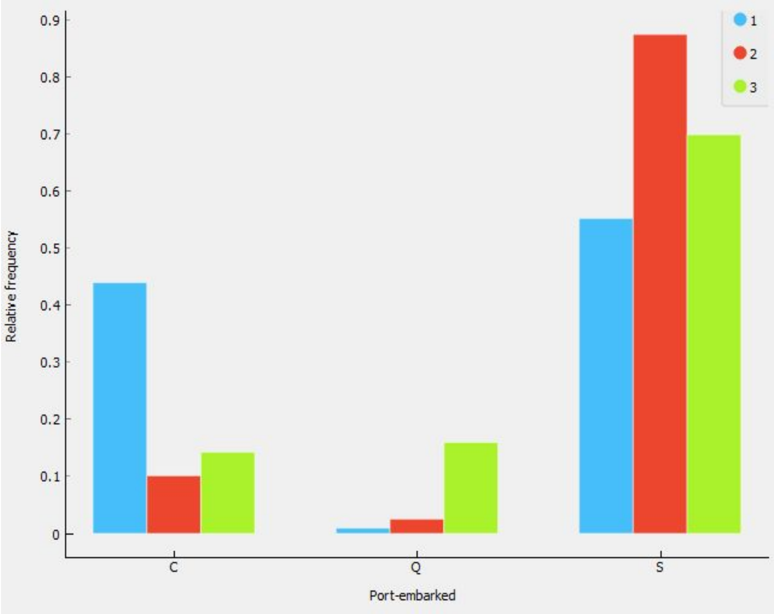
- There are several Titanic datasets, here the most complete one was used (**REAL** data)
- Objective: predict which passengers survived the sinking
  - Instances: 1309
  - Predicting atributes: 9 (+6 meta)
  - Passenger class: 1st, 2nd, 3rd
  - Sex: male, female
  - Age: 0..80 (avg 30 ± 14)
  - Siblings-spouses-aboard
  - Port-embarked: Southampton, Cheerbourg, Queenstown (910, 270, 1230)
  - Fare (?)
  - Body: (?)
  - Meta Attribute: survived (no, yes)
  - No: 809, yes: 500

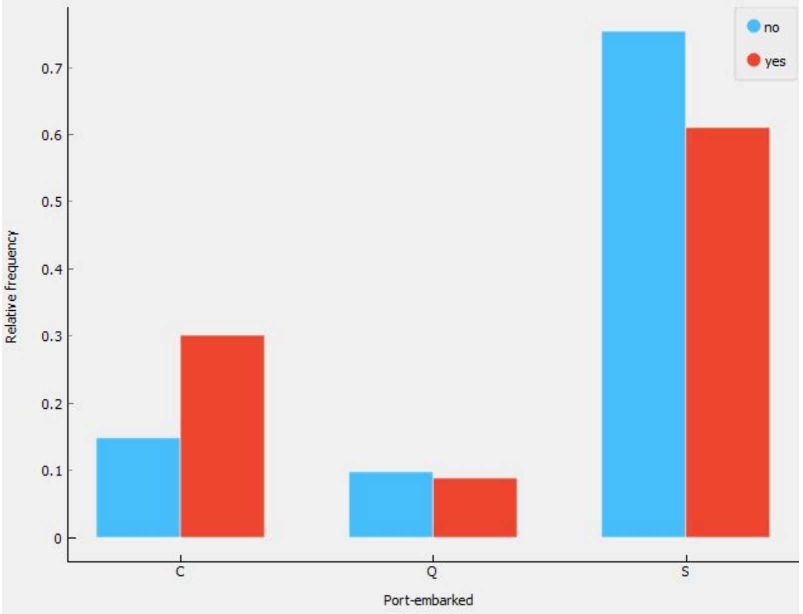
Queenstown
Southampton
Cherbourg
New York
Titanic

Case study #2: Titanic dataset



- Case distribution by port of embarkation and by survival

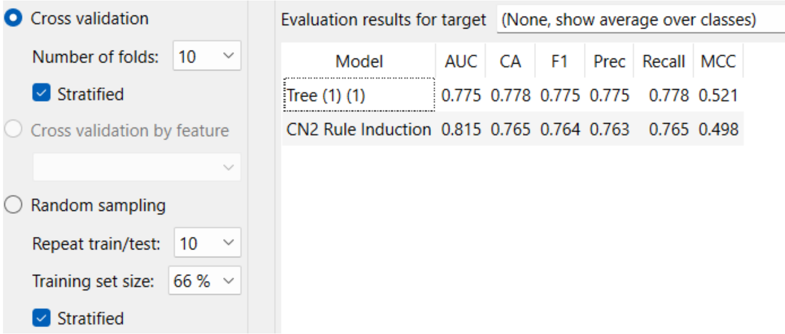




Southampton
Cheerbourg
Queenstown

Case study #2: Titanic dataset

- Pruned decision tree obtained with the full dataset



● Cross validation

Number of folds: 10

☑ Stratified

○ Cross validation by feature

○ Random sampling

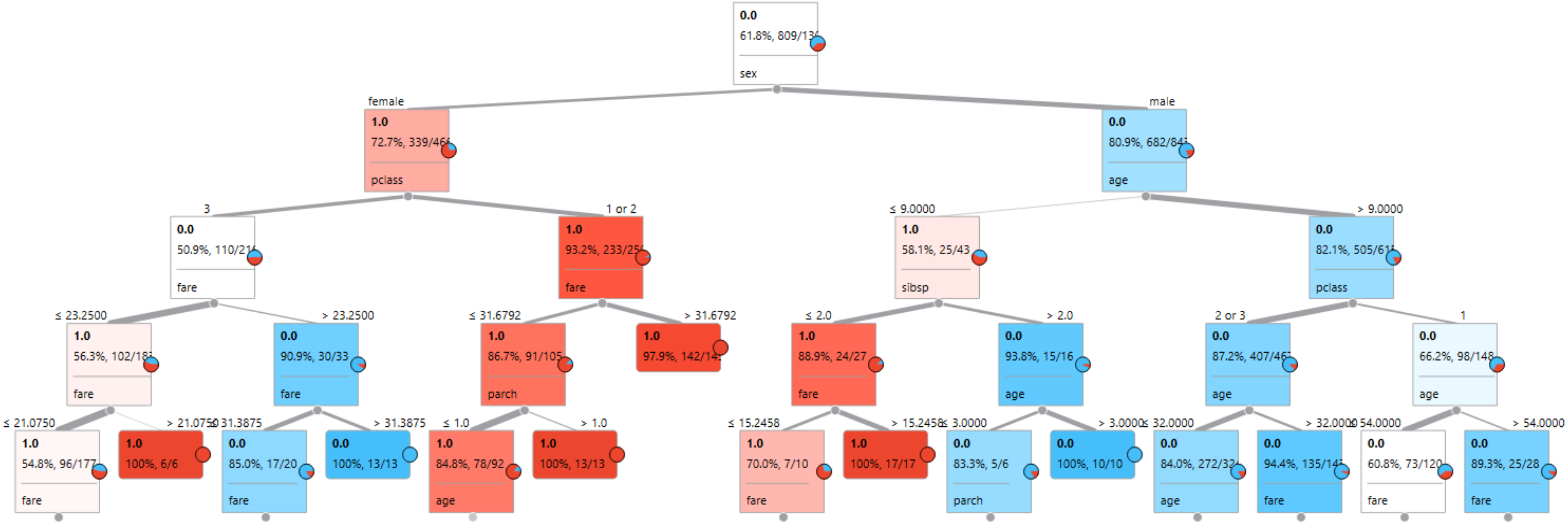
Repeat train/test: 10

Training set size: 66 %

☑ Stratified

Evaluation results for target (None, show average over classes)

Model	AUC	CA	F1	Prec	Recall	MCC
Tree (1) (1)	0.775	0.778	0.775	0.775	0.778	0.521
CN2 Rule Induction	0.815	0.765	0.764	0.763	0.765	0.498


0.0
61.8%, 809/13
sex
female
1.0
72.7%, 339/46
pclass
male
0.0
80.9%, 682/84
age
3
0.0
50.9%, 110/21
fare
1 or 2
1.0
93.2%, 233/25
fare
≤ 9.0000
1.0
58.1%, 25/43
sibsp
> 9.0000
0.0
82.1%, 505/61
pclass
≤ 23.2500
1.0
56.3%, 102/18
fare
> 23.2500
0.0
90.9%, 30/33
fare
≤ 31.6792
1.0
86.7%, 91/105
parch
> 31.6792
1.0
97.9%, 142/14
≤ 2.0
1.0
88.9%, 24/27
fare
> 2.0
0.0
93.8%, 15/16
age
2 or 3
0.0
87.2%, 407/46
age
1
0.0
66.2%, 98/148
age
≤ 21.0750
1.0
54.8%, 96/177
fare
> 21.0750
1.0
100%, 6/6
≤ 31.3875
0.0
85.0%, 17/20
fare
> 31.3875
0.0
100%, 13/13
≤ 1.0
1.0
84.8%, 78/92
age
> 1.0
1.0
100%, 13/13
≤ 15.2458
1.0
70.0%, 7/10
fare
> 15.2458
1.0
100%, 17/17
≤ 3.0000
0.0
83.3%, 5/6
parch
> 3.0000
0.0
100%, 10/10
≤ 32.0000
0.0
84.0%, 272/32
age
> 32.0000
0.0
94.4%, 135/14
fare
≤ 54.0000
0.0
60.8%, 73/120
fare
> 54.0000
0.0
89.3%, 25/28
fare