Prediction

- Label
  - A possible outcome of an event
  - Binary
    - Person can be “child” or “adult”
  - Nominal
    - Car can be “family”, “sport”, “terrain” or “truck”
- Ordinal
  - Movies can be rated “worst”, “bad”, “neutral”, “good” and “excellent”
- Quantitative
  - Houses have prices

Supervised Prediction

- Supervised predictive task
- The goal is to build a predictive model, from the labeled (train) instances in the data
- Which maps a vector of predictive attribute values to labels,
- In order to assign the correct labels for the unlabeled (test) instances in the data

Prediction

- Regression task
  - Labels are quantitative
- Classification task
  - Labels are binary, nominal or ordinal

Classification

- One of the most frequent task in analytics
  - Without paying attention, we are all the time classifying things
  - We perform a classification task when:
    - Deciding if we are going to stay at home, go out or visit a friend
    - Choosing a meal in a restaurant
    - Adding someone to our social network
    - Decide if someone is a friend

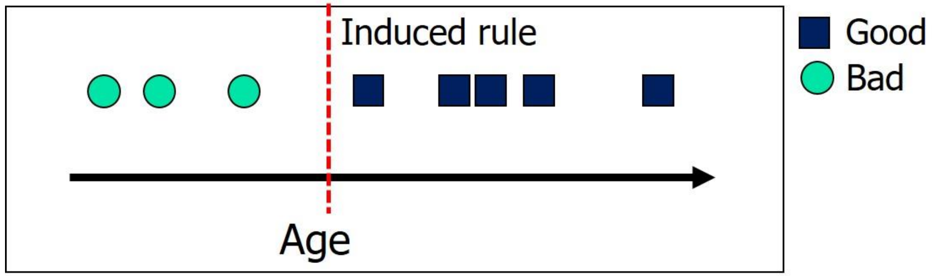
Classification

- Classification task

Predictive task where a label to be assigned to a new, unlabeled, object, given the value of its predictive attributes, is a qualitative value representing a class or category.

Example

Name	Age	Company
Andrew	51	Good
Bernhard	43	Good
Dennis	82	Good
Eve	23	Bad
Fred	46	Good
Irene	29	Bad
James	42	Good
Lea	38	Good
Mary	31	Bad

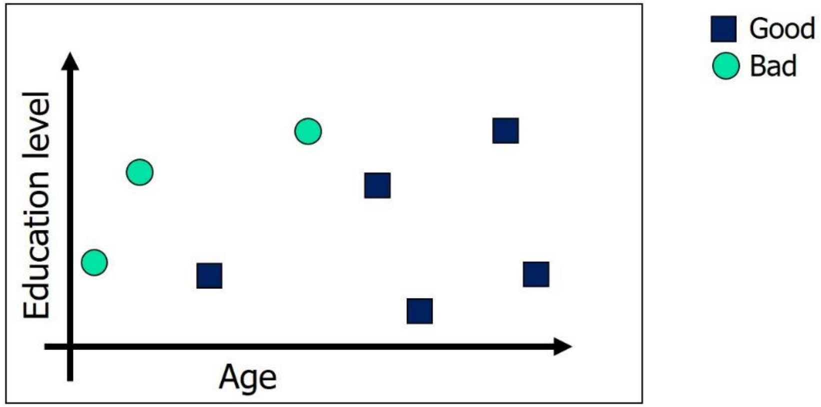


If person-age < 32
Then dinner will be Bad
Else dinner will be Good

Classification model induced for the previous binary classification task

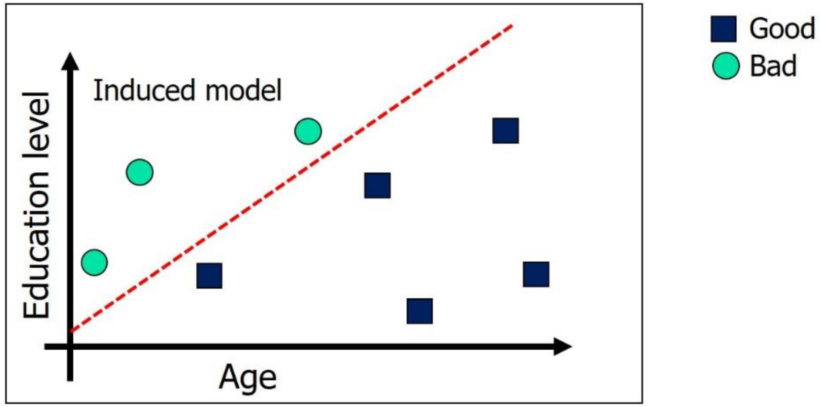
Example

Name	Age	Education level	Company
Andrew	51	1.0	Good
Bernhard	43	2.0	Good
Dennis	82	3.0	Good
Eve	23	3.5	Bad
Fred	46	5.0	Good
Irene	29	4.5	Bad
James	42	4.0	Good
Lea	38	5.0	Bad
Mary	31	3.0	Good



Example

Name	Age	Education level	Company
Andrew	51	1.0	Good
Bernhard	43	2.0	Good
Dennis	82	3.0	Good
Eve	23	3.5	Bad
Fred	46	5.0	Good
Irene	29	4.5	Bad
James	42	4.0	Good
Lea	38	5.0	Bad
Mary	31	3.0	Good



If person > decision border
Then dinner will be Bad
Else dinner will be Good

Classification model induced for the previous binary classification task

Predictive performance measures

- Assess predictive performance of a classification model
  - How frequent the predicted labels are the true class labels
  - Model predictive performance must be better than predicting in the majority class
    - Class with the largest number of objects
  - Several predictive performance measures
    - Derived from confusion matrix

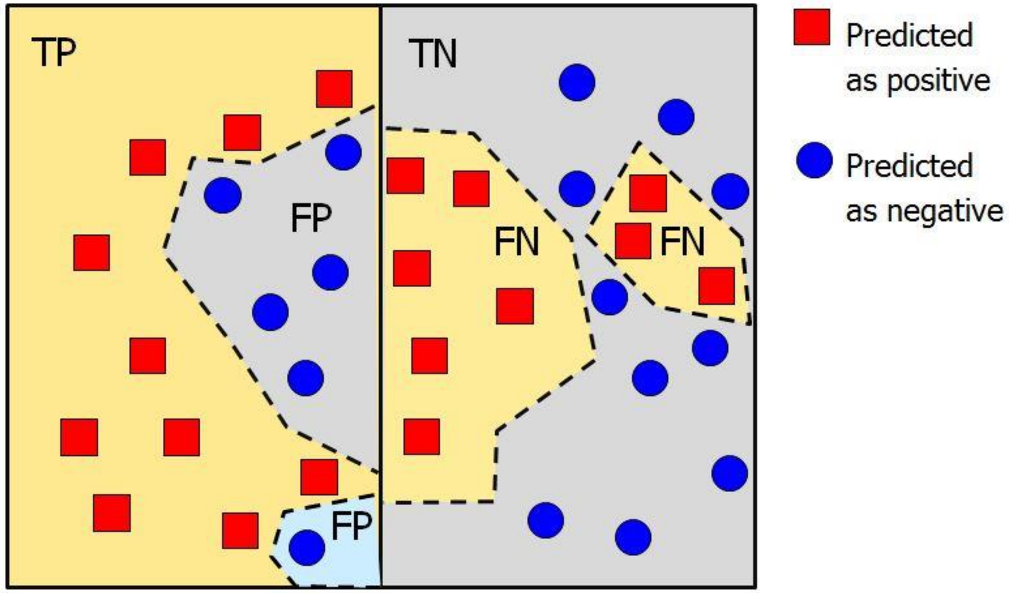
Predictive performance measures

- Confusion matrix reports the predictive performance of a binary classifier
  - True class
    - Positive class
    - Negative class
  - Predicted class

		True class	
		p	n
Predicted class	P	True positives (TP)	False positives (FP)
	N	False negatives (FN)	True negatives (TN)

Predictive performance measures

- According to the predictive attribute values, true classes and predicted classes can differ



Predictive performance measures

$$\frac{FP}{FP+TN}$$

False positive rate (FPR) = 1-TNR

$$\frac{FN}{TP+FN}$$

False negative rate (FNR) = 1-TPR

$$\boxed{\frac{TP}{TP+FN}}$$

True positive rate (TPR), also known as recall or sensitivity

$$\boxed{\frac{TN}{TN+FP}}$$

True negative rate (TNR), also known as specificity

$$\boxed{\frac{TP}{TP+FP}}$$

Positive predictive value (PPV), also known as precision

$$\boxed{\frac{TN}{TN+FN}}$$

Negative predictive value (NPV)

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

Accuracy

$$\frac{2}{1/\,precision+1/\,recall}$$

F1-measure

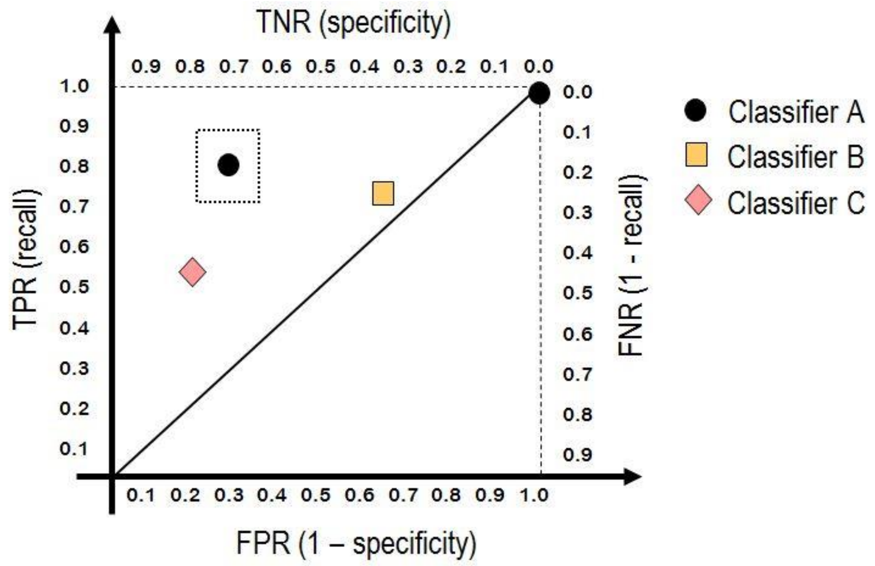
Predictive performance measures

Metric	Formula	Description
Accuracy	$\frac{\text{TP+TN}}{\text{TP+TN+FP+FN}}$	Overall performance of model
Precision	$\frac{\text{TP}}{\text{TP+FP}}$	How accurate the positive predictions are
Recall/Sensitivity	$\frac{\text{TP}}{\text{TP+FN}}$	Coverage of actual positive sample
Specificity	$\frac{\text{TN}}{\text{TN+FP}}$	Coverage of actual negative sample
F1-score	$2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$	Harmonic mean of Precision and Recall

Taken from: http://www.davidsbatista.net/blog/2018/08/19/NLP_Metrics/ (in 2019-08-05)

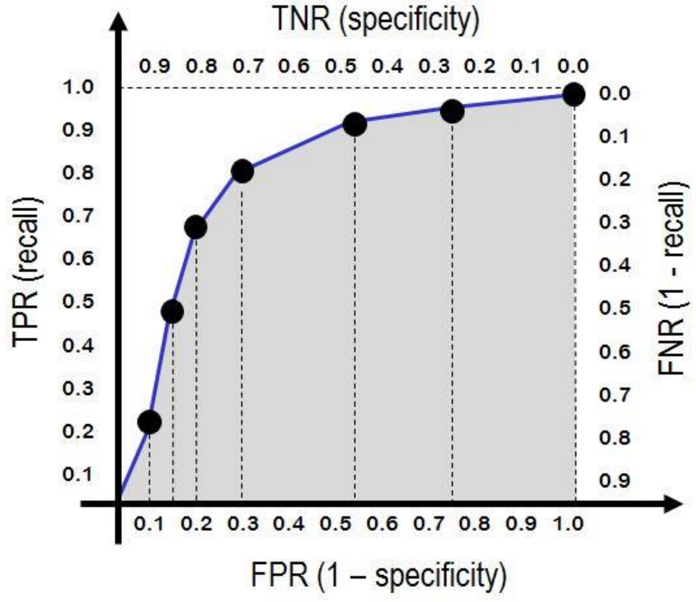
Predictive performance measures

- Some of the previous measures can be combined
  - E.g.: Receiver operating Characteristics (ROC) graph combines recall and specificity



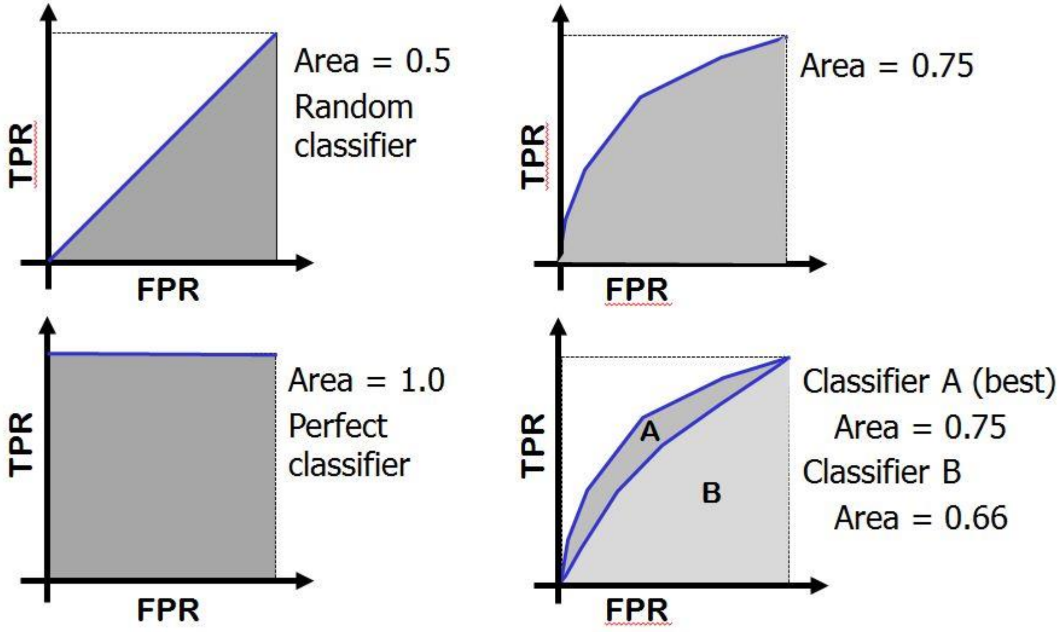
Predictive performance measures

- A better predictive performance estimate can be obtained using several ROC points
  - When connected form an area under the ROC curve (AUC)
  - Area under the ROC curve
    - Can be calculated by adding sub-areas
    - The larger the area, the better



Predictive performance measures

- AUC can be used to illustrate the predictive performance of different classifiers
  - Classifier with the best predicti performance is closer to the lef top



Generalization

- We want to evaluate how our method can perform under new data

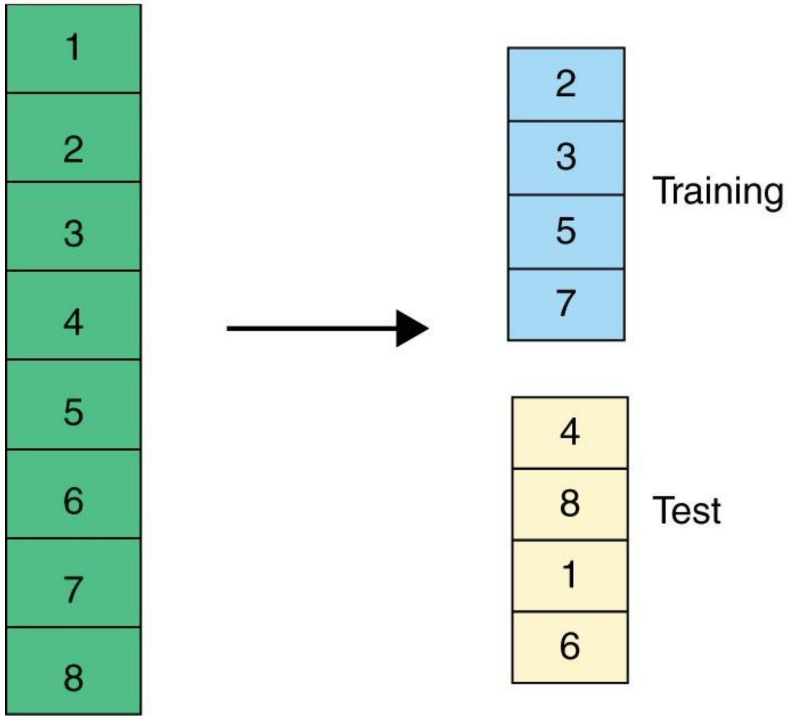
Generalization

- We separate the training data set into two mutually exclusive parts:
  - One for training - model parameter tuning, and
  - One for testing - evaluating the induced model on new data for which the labels are known

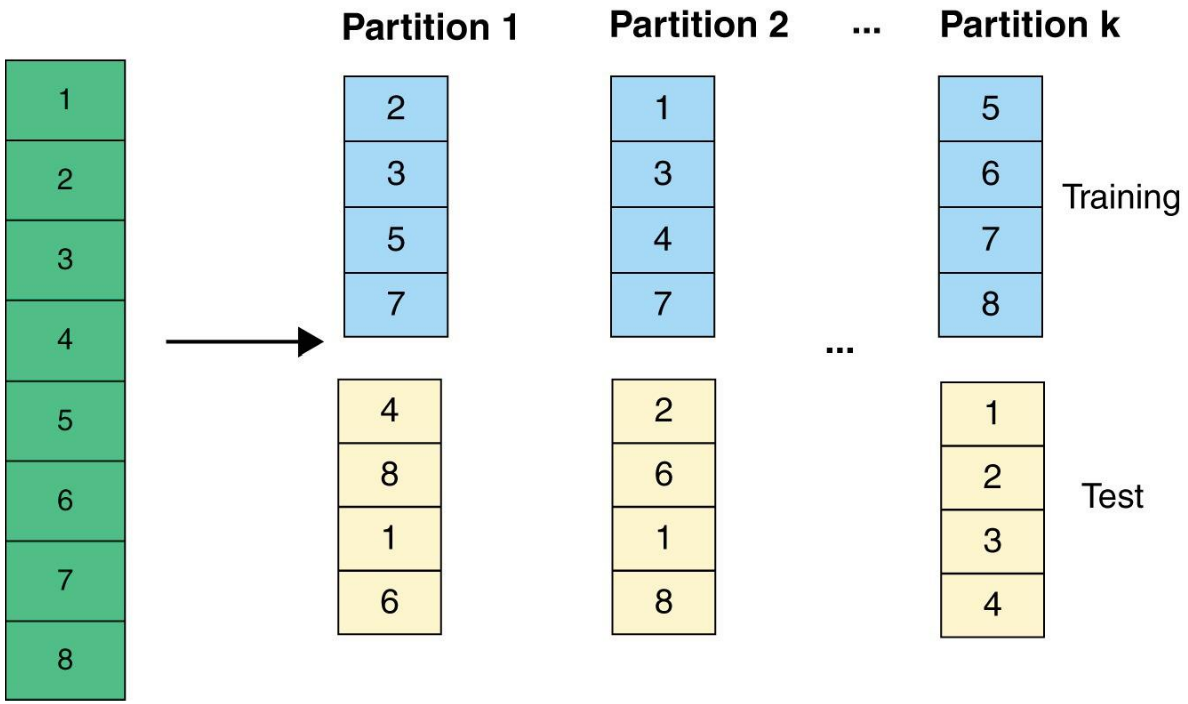
- Two important issues are:
  - How to estimate the method performance for new data
  - What performance measure will be used in this estimation

Model validation

Holdout validation

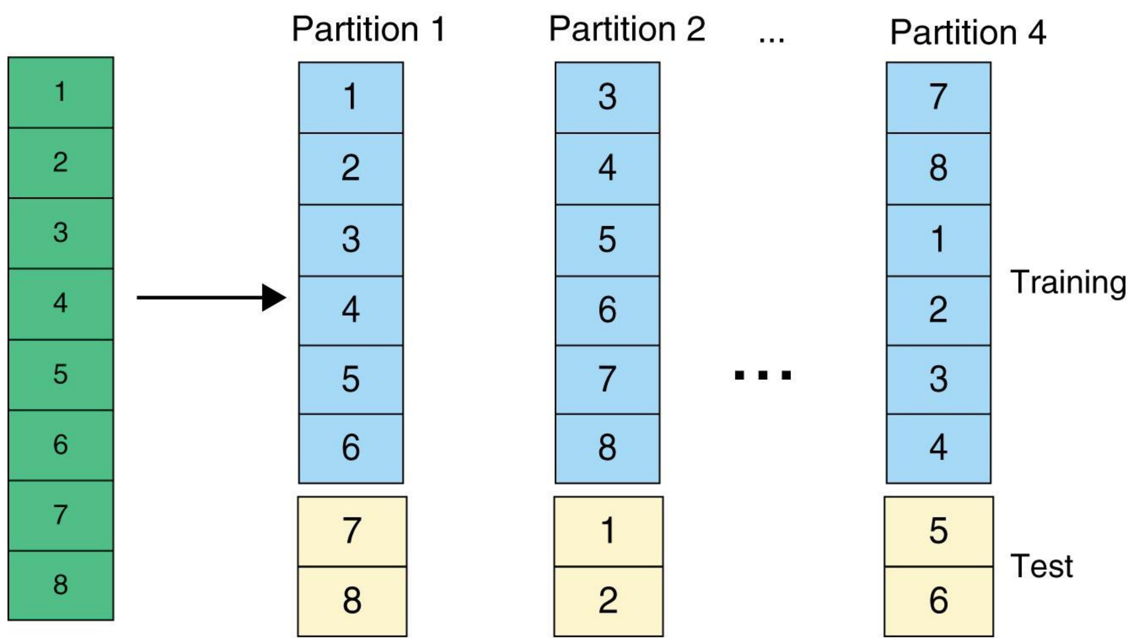


Random sub-sampling

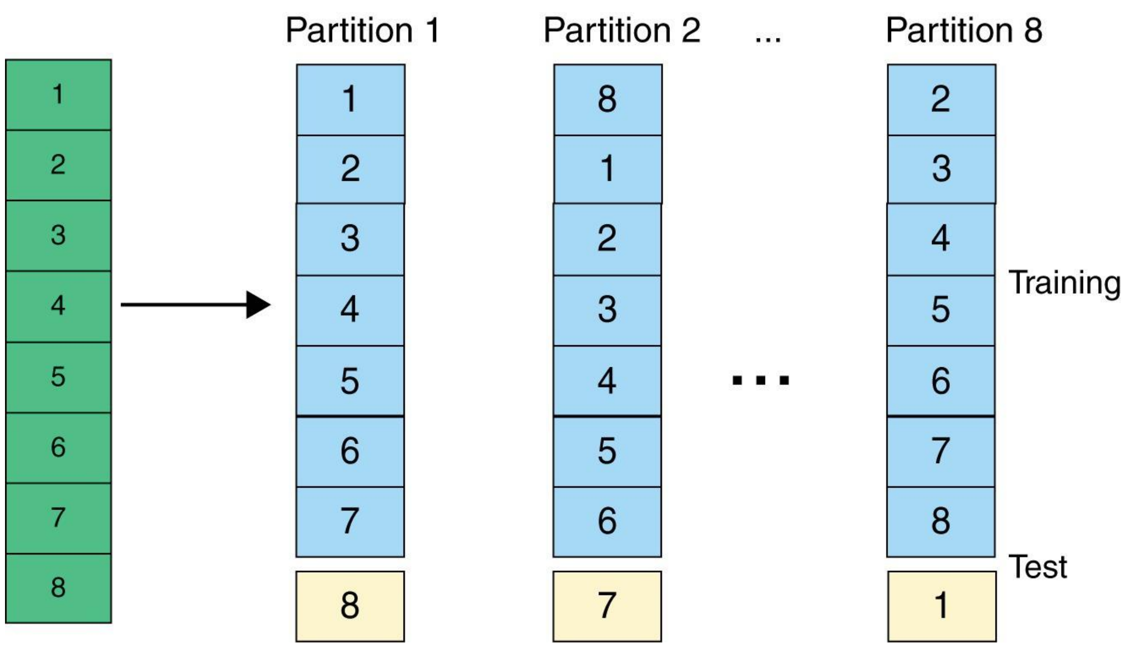


Model validation

k-fold cross validation

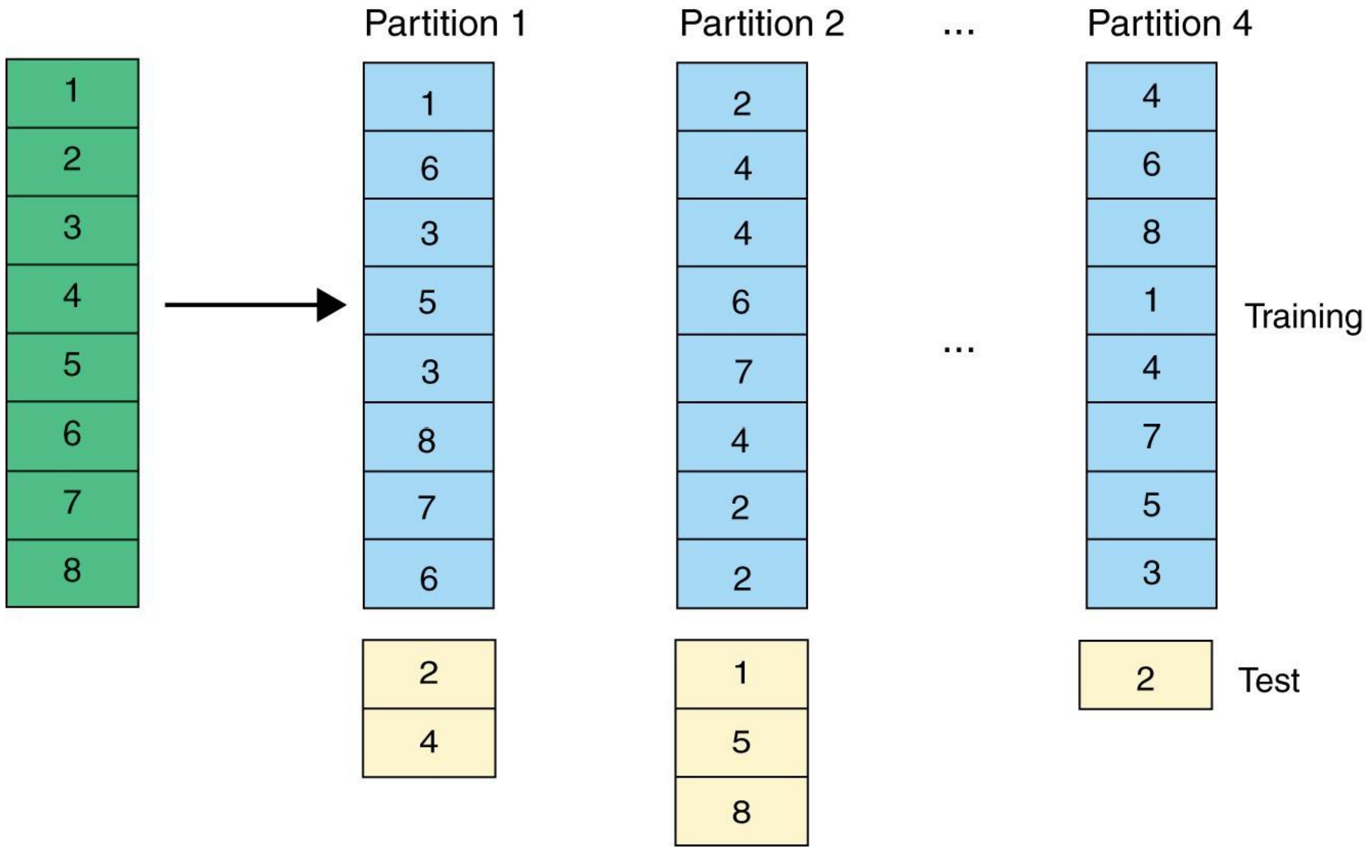


Leave-one-out



Model validation

Bootstrap

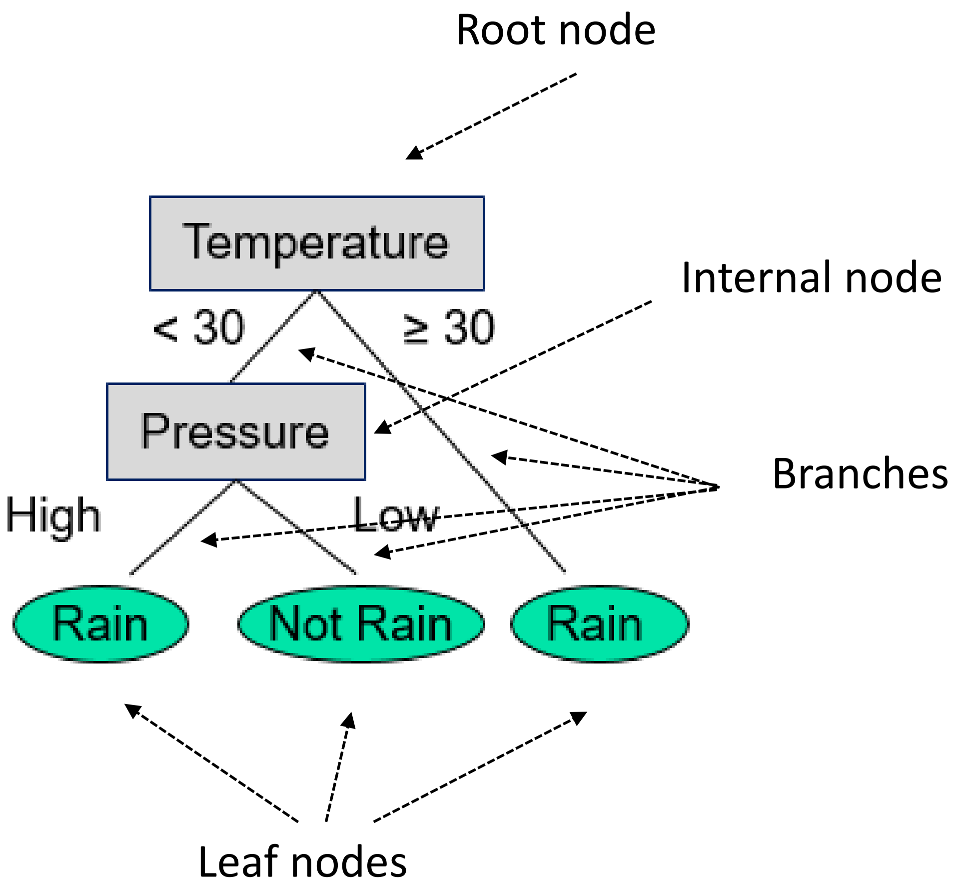


Decision tree induction algorithms

- Classification trees
  - Decision trees for classification tasks
- Learn by partitioning predictive attributes in a decision tree format
  - Greedy learning approach
  - From root node to leaf nodes
  - Separate training examples using impurity measures
    - The more pure the child nodes, the better

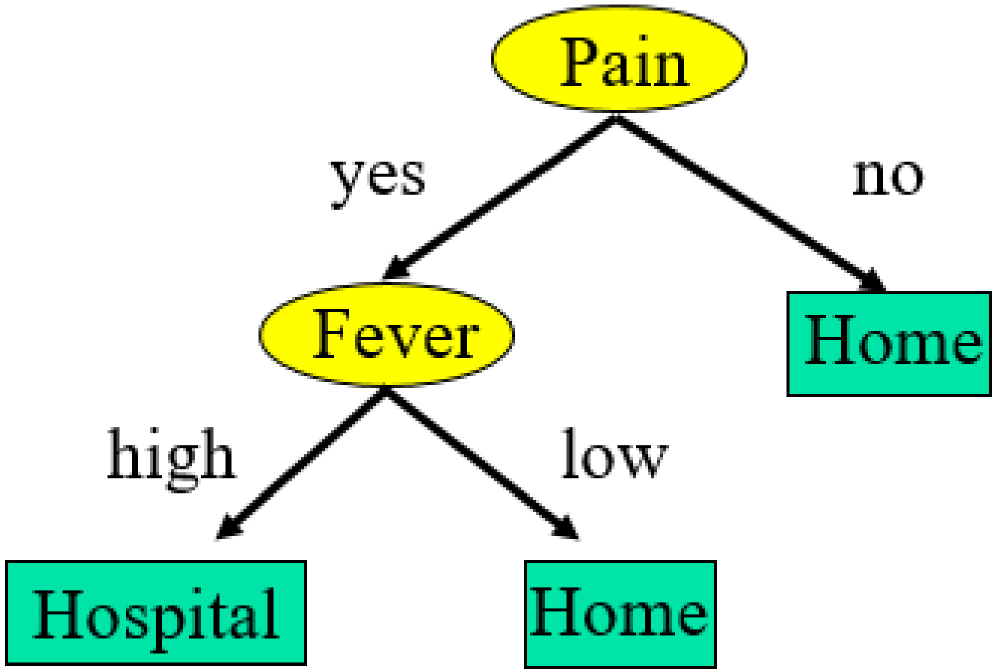
Decision tree induction algorithms

- Decision tree (DT)
  - Root node and internal nodes represent predictive attributes
  - Branches represent decisions
  - Leaves represent classes or values



Example

Name	Pain	Temperature	Outcome
Andrew	no	high	Home
Bernhard	yes	high	Hospital
Mary	no	high	Home
Dennis	yes	low	Home
Eve	yes	high	Hospital
Fred	yes	high	Hospital
Lea	no	low	Home
Irene	yes	low	Home
James	yes	high	Hospital



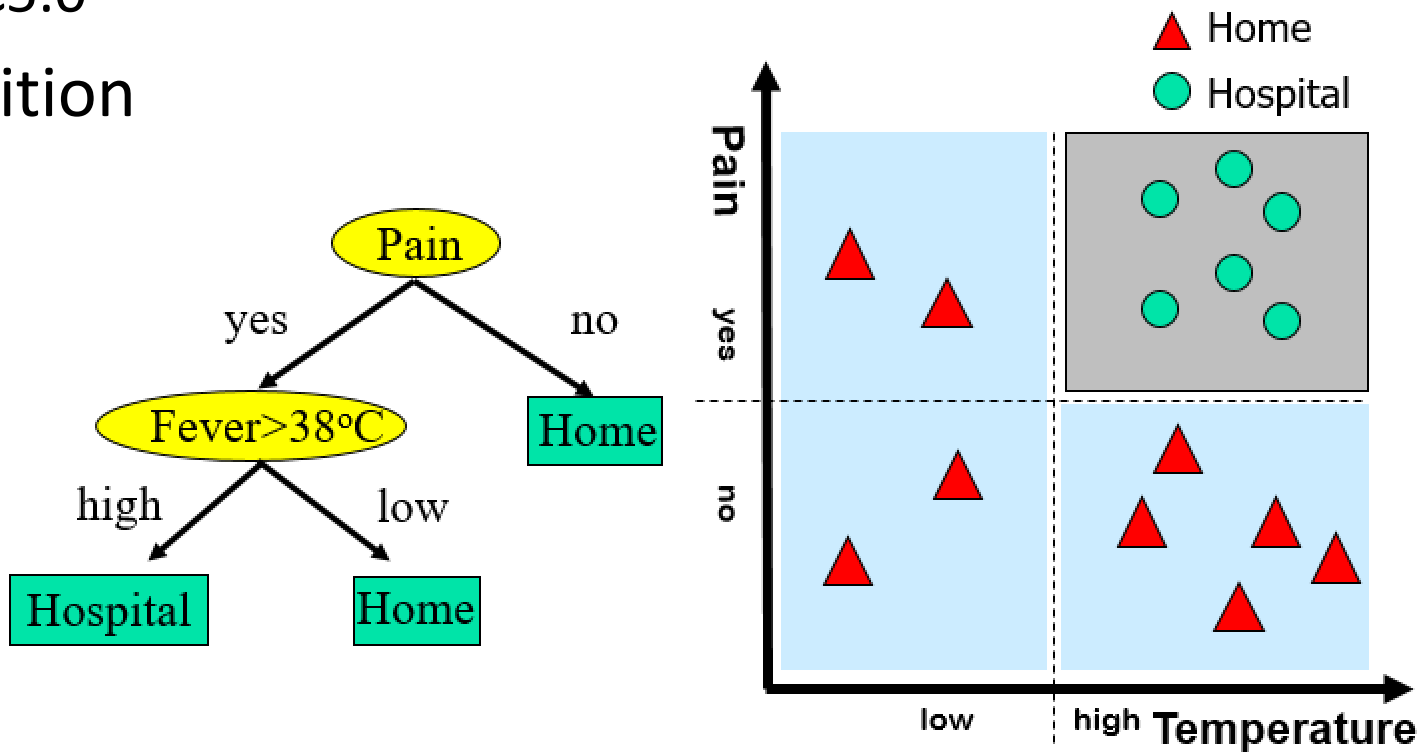
DT induction algorithms

Algorithm Hunt decision tree induction algorithm

INPUT D_{train} current node training set
INPUT p the impurity measure
INPUT n the number of objects in the training set
if all objects in D_{train} belongs to the same class y **then**
    The current node is a leaf node labeled with class y
else
    Select a predictive attribute to split D_{train} using the impurity measure p
    Split D_{train} in subsets according to its current values
    Apply Hunt algorithm to each subset

DT induction algorithms

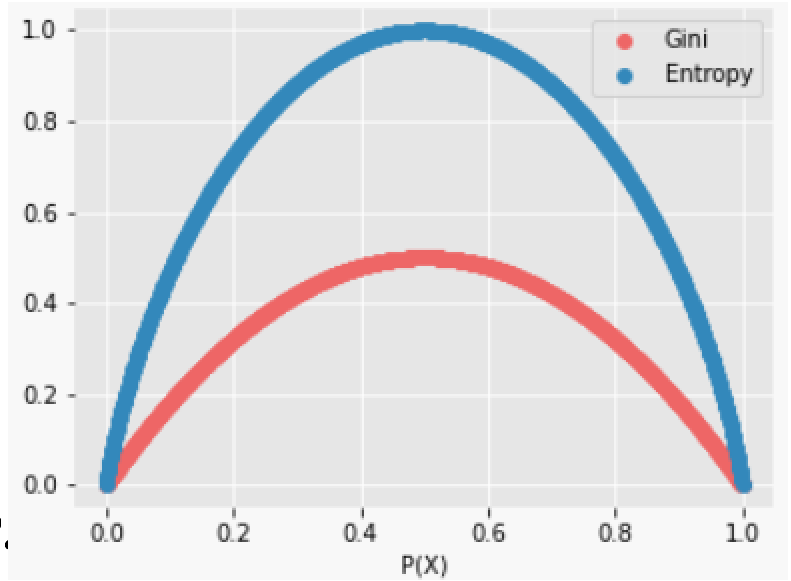
- There are many DT induction algorithms
  - E.g. CART and C5.0
- Input space partition



DT induction algorithms

Main impurity measures

- For classification
  - GiniIndex=1– $\sum_j p_j^2$
    - $Gini_{min}$=1–(1^2)=0
    - $Gini_{max}$=1–($0.5^2 + 0.5^2$) = 0.5
  - Entropy=$-\sum_j p_j \times log_2(p_j)$
    - $Entropy_{min} = -1 \times log_2$ (1)=0
    - $Entropy_{max} = -0.5 \times\ log_2$ (0.5) $-0.5 \times\ lo$

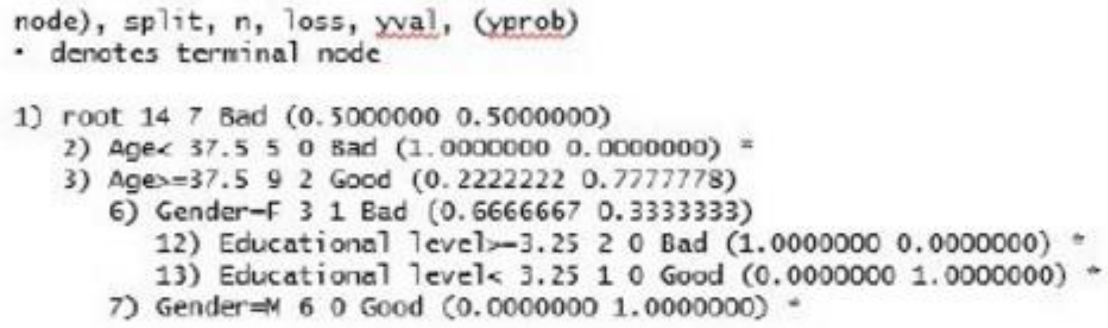


- For regression
  - Variance reduction

Search-based algorithms: decision trees

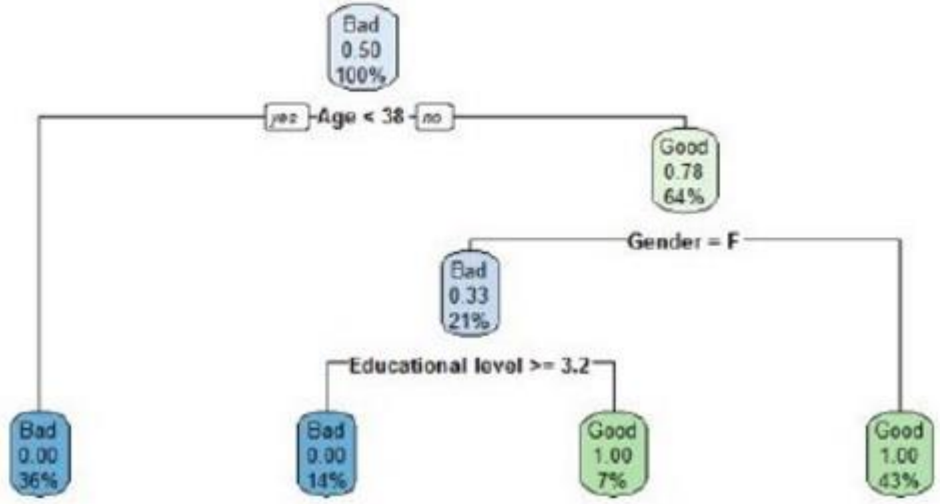
Accessing and evaluating results:

- Decision tree models are interpretable
- They can be represented as a graph like the one in the right Figure or as a set of rules as shown the left Figure



```
node), split, n, loss, yval, (yprob)
* denotes terminal node

1) root 14 7 Bad (0.5000000 0.5000000)
  2) Age< 37.5 5 0 Bad (1.0000000 0.0000000) *
  3) Age>=37.5 9 2 Good (0.2222222 0.7777778)
    6) Gender=F 3 1 Bad (0.6666667 0.3333333)
      12) Educational level>=3.25 2 0 Bad (1.0000000 0.0000000) *
      13) Educational level< 3.25 1 0 Good (0.0000000 1.0000000) *
    7) Gender=M 6 0 Good (0.0000000 1.0000000) *
```



Search-based algorithms: decision trees

Setting the hyper-parameters:

- Each algorithm can have different hyper-parameters to be set
- Most hyper-parameters that can be found in implementations of decision tree induction algorithms are to control the pruning, both pre and post pruning
- The most common of these hyper-parameters is the minimum number of objects a leaf node must have
- If very low it can promote over-fitting

DT induction algorithms pros & cons

- Pros
  - Interpretable both as a graph and as a set of rules
  - Pre-processing free
    - Robust to outliers, missing data, correlated and irrelevant attributes and do not need previous normalization
- Cons
  - The definition of a rule to split a node is evaluated locally without enough information to know if it guarantees the global optimum
  - Splits the bi-dimensional space with horizontal and vertical lines, which creates difficulties to model some problems

An example

```
> rpart.tree
n= 1788
node), split, n, deviance, yval
      * denotes terminal node
 1) root 1788 859766200 4382.295
   2) InicioViagem>=71122 104   9184189 3198.577 *
   3) InicioViagem< 71122 1684 695858900 4455.398
     6) DiaSemana=domingo      ,sábado        335  37278610 3842.752 *
     7) DiaSemana=quarta-feira ,quinta-feira ,segunda-feira,sexta-feira  ,terça-feira   1349 501618300 4607.538
      14) InicioViagem< 26481.5 119  15156970 3585.496
        28) InicioViagem< 25549 82   3928321 3402.988 *
        29) InicioViagem>=25549 37   2444027 3989.973 *
      15) InicioViagem>=26481.5 1230 350131300 4706.419
        30) InicioViagem< 49033.5 660 122045000 4527.662
          60) TipoDia=tolerancia 11   1055496 3494.273 *
          61) TipoDia=normal,ponte 649 109043500 4545.177 *
        31) InicioViagem>=49033.5 570 182577200 4913.400
          62) DiaAno< 55.5 260  45889890 4652.908
           124) DiaAno>=54.5 15   1340716 3766.533 *
           125) DiaAno< 54.5 245  32042760 4707.176
             250) InicioViagem>=67928 28   3853810 4128.357 *
             251) InicioViagem< 67928 217  17597650 4781.862 *
          63) DiaAno>=55.5 310 104247700 5131.877
           126) InicioViagem>=65837.5 45  13490400 4533.200 *
           127) InicioViagem< 65837.5 265  71889780 5233.540 *
```