# Machine Learning models
# - Probabilistic model

# Machine Learning models

Machine learning models can be distinguished according to their main intuition:

- Geometric models use intuitions from geometry such as separating (hyper-)planes, linear transformations and distance metrics.
- Probabilistic models view learning as a process of reducing uncertainty.
- Logical models are defined in terms of logical expressions.

Alternatively, they can be characterised by their *modus operandi*:

- Grouping models divide the instance space into segments; in each segment a very simple (e.g., constant) model is learned.
- Grading models learning a single, global model over the instance space.

# Things We’d Like to Do

- Spam Classification
  - Given an email, predict whether it is spam or not
- Medical Diagnosis
  - Given a list of symptoms, predict whether a patient has cancer or not
- Weather
  - Based on temperature, humidity, etc… predict if it will rain tomorrow

# Probabilistic Model

- ➢Uncertainty & Probability
- ➢Baye's rule
- ➢Choosing Hypotheses- Maximum a posteriori
- ➢Maximum Likelihood - Baye's concept learning
- ➢Maximum Likelihood of real valued function

# Uncertainty

- Our main tool is the probability theory, which assigns to each sentence numerical degree of belief between *0* and *1*

- It provides a way of summarizing the uncertainty

# Variable

- Boolean random variables: cavity might be true or false
- Discrete random variables: weather might be sunny, rainy, cloudy, snow
  - *P(Weather=sunny)*
  - *P(Weather=rainy)*
  - *P(Weather=cloudy)*
  - *P(Weather=snow)*
- Continuous random variables: the temperature has continuous values

# Where do probabilities come from?

- Frequents:
  - From experiments: form any finite sample, we can estimate the true fraction and also calculate how accurate our estimation is likely to be
- Subjective:
  - Agent's believe
- Objectivist:
  - True nature of the universe, that the probability up heads with probability 0.5 is a probability of the coin

# Contd...

- Before the evidence is obtained; prior probability
  - *P(a)* the prior probability that the proposition is true
  - *P(cavity)=0.1*

- After the evidence is obtained; posterior probability
  - *P(a|b)*
  - The probability of a given that all we know is b
  - *P(cavity|toothache)=0.8*

# Axioms of Probability

Zur Anzeige wird der QuickTime™ Dekompressor „TIFF (Unkomprimiert)" benötigt.

(Kolmogorov's axioms,
first published in German 1933)

- All probabilities are between 0 and 1. For any proposition *a* $0 \leq P(a) \leq 1$

- $P(true)=1, \;\; P(false)=0$

- The probability of disjunction is given by

$$P(a \vee b) = P(a) + P(b) - P(a \wedge b)$$

# Contd..

- Product rule

$$P(a \wedge b) = P(a \mid b)P(b)$$

$$P(a \wedge b) = P(b \mid a)P(a)$$

# Theorem of total probability

If events $A_1, \ldots, A_n$ are mutually

exclusive with then

$$\sum_{i=1}^{n} P(A_i) = 1$$

$$P(B) = \sum_{i=1}^{n} P(B|A_i)P(A_i)$$

$$P(B) = \sum_{i=1}^{n} P(B, A_i)$$

# Bayes Theorem

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- $P(h)$ = prior probability of hypothesis h
- $P(D)$ = prior probability of training data $D$
- $P(h/D)$ = probability of $h$ given $D$
- $P(D/h)$ = probability of $D$ given $h$

# Choosing Hypotheses

- Generally want the most probable hypothesis given the training data

- **Maximum a posteriori** hypothesis $h_{MAP}$:

$$h_{MAP} = \arg\max_{h \in H} P(h|D)$$

# Contd..

$$h_{MAP} = \arg\max_{h \in H} P(h|D)$$

$$= \arg\max_{h \in H} \frac{P(D|h)P(h)}{P(D)}$$

$$= \arg\max_{h \in H} P(D|h)P(h)$$

# Contd..

- If assume $P(h_i)=P(h_j)$ for all $h_i$ and $h_j$, then can further simplify, and choose the

- **Maximum likelihood** (ML) hypothesis

$$h_{ML} = \arg\max_{h_i \in H} P(D|h_i)$$

# Naïve Bayesian Classification

- If i-th attribute is categorical:
  $P(d_i|C)$ is estimated as the relative freq of samples having value $d_i$ as i-th attribute in class C
- If i-th attribute is continuous:
  $P(d_i|C)$ is estimated thru a Gaussian density function
- Computationally easy in both cases

# Play-tennis example: estimating $P(x_i|C)$

| Outlook | Temperature | Humidity | Windy | Class |
|---|---|---|---|---|
| sunny | hot | high | false | N |
| sunny | hot | high | true | N |
| overcast | hot | high | false | P |
| rain | mild | high | false | P |
| rain | cool | normal | false | P |
| rain | cool | normal | true | N |
| overcast | cool | normal | true | P |
| sunny | mild | high | false | N |
| sunny | cool | normal | false | P |
| rain | mild | normal | false | P |
| sunny | mild | normal | true | P |
| overcast | mild | high | true | P |
| overcast | hot | normal | false | P |
| rain | mild | high | true | N |

| | |
|---|---|
| P(p) = 9/14 | |
| P(n) = 5/14 | |

| | |
|---|---|
| **outlook** | |
| P(sunny\|p) = 2/9 | P(sunny\|n) = 3/5 |
| P(overcast\|p) = 4/9 | P(overcast\|n) = 0 |
| P(rain\|p) = 3/9 | P(rain\|n) = 2/5 |
| **temperature** | |
| P(hot\|p) = 2/9 | P(hot\|n) = 2/5 |
| P(mild\|p) = 4/9 | P(mild\|n) = 2/5 |
| P(cool\|p) = 3/9 | P(cool\|n) = 1/5 |
| **humidity** | |
| P(high\|p) = 3/9 | P(high\|n) = 4/5 |
| P(normal\|p) = 6/9 | P(normal\|n) = 2/5 |
| **windy** | |
| P(true\|p) = 3/9 | P(true\|n) = 3/5 |
| P(false\|p) = 6/9 | P(false\|n) = 2/5 |

# Naive Bayesian Classifier (II)

- Given a training set, we can compute the probabilities

| Outlook | P | N |
|---|---|---|
| sunny | 2/9 | 3/5 |
| overcast | 4/9 | 0 |
| rain | 3/9 | 2/5 |
| **Tempreature** | | |
| hot | 2/9 | 2/5 |
| mild | 4/9 | 2/5 |
| cool | 3/9 | 1/5 |

| Humidity | P | N |
|---|---|---|
| high | 3/9 | 4/5 |
| normal | 6/9 | 1/5 |
| | | |
| **Windy** | | |
| true | 3/9 | 3/5 |
| false | 6/9 | 2/5 |

# Play-tennis example: classifying X

- An unseen sample X = <rain, hot, high, false>

- $P(X|p) \cdot P(p) =$
  $P(rain|p) \cdot P(hot|p) \cdot P(high|p) \cdot P(false|p) \cdot P(p) =$
  $3/9 \cdot 2/9 \cdot 3/9 \cdot 6/9 \cdot 9/14 = 0.010582$
- $P(X|n) \cdot P(n) =$
  $P(rain|n) \cdot P(hot|n) \cdot P(high|n) \cdot P(false|n) \cdot P(n) =$
  $2/5 \cdot 2/5 \cdot 4/5 \cdot 2/5 \cdot 5/14 = 0.018286$

- Sample X is classified in class n (don't play)

# The independence hypothesis…

- … makes computation possible
- … yields optimal classifiers when satisfied
- … but is seldom satisfied in practice, as attributes (variables) are often correlated.
- Attempts to overcome this limitation:
  - Bayesian networks, that combine Bayesian reasoning with causal relationships between attributes

# Training dataset

Class:
C1:buys_computer='yes'
C2:buys_computer='no'

Data sample:

X =
(age<=30,
Income=medium,
Student=yes
Credit_rating=Fair)

| age | income | student | credit_rating | buys_computer |
|---|---|---|---|---|
| <=30 | high | no | fair | no |
| <=30 | high | no | excellent | no |
| 30…40 | high | no | fair | yes |
| >40 | medium | no | fair | yes |
| >40 | low | yes | fair | yes |
| >40 | low | yes | excellent | no |
| 31…40 | low | yes | excellent | yes |
| <=30 | medium | no | fair | no |
| <=30 | low | yes | fair | yes |
| >40 | medium | yes | fair | yes |
| <=30 | medium | yes | excellent | yes |
| 31…40 | medium | no | excellent | yes |
| 31…40 | high | yes | fair | yes |
| >40 | medium | no | excellent | no |

# Naïve Bayesian Classifier: Example

- Compute $P(X|C_i)$ for each class

likelihood

P(age="<30" | buys_computer="yes") = 2/9=0.222
P(age="<30" | buys_computer="no") = 3/5 =0.6
P(income="medium" | buys_computer="yes")= 4/9 =0.444
P(income="medium" | buys_computer="no") = 2/5 = 0.4
P(student="yes" | buys_computer="yes)= 6/9 =0.667
P(student="yes" | buys_computer="no")= 1/5=0.2
P(credit_rating="fair" | buys_computer="yes")=6/9=0.667
P(credit_rating="fair" | buys_computer="no")=2/5=0.4

P(buys_computer=„yes“)=9/14
P(buys_computer=„no“)=5/14

Prior

- X=(age<=30 ,income =medium, student=yes,credit_rating=fair)

Text Sample!

**$P(X|C_i)$ :** P(X|buys_computer="yes")= 0.222 x 0.444 x 0.667 x 0.0.667 =0.044

P(X|buys_computer="no")= 0.6 x 0.4 x 0.2 x 0.4 =0.019

**$P(X|C_i)*P(C_i)$ :** P(X|buys_computer="yes") * P(buys_computer="yes")=0.028
P(X|buys_computer="no") * P(buys_computer="no")=0.007

0.028 > 0.007
P(X|yes)P(yes)
P(X|no)P(no)

- X belongs to class "buys_computer=yes"