Fundamentals of Machine Learning

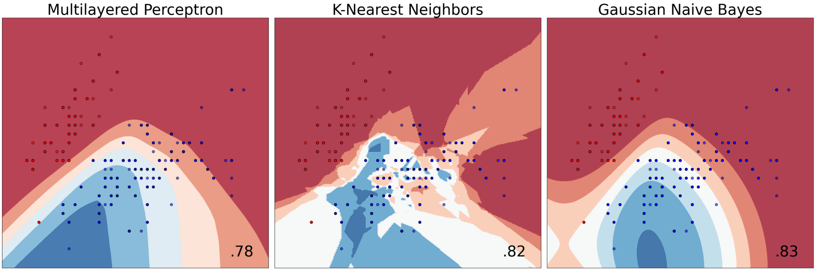
Billy Braithwaite

IT Center for Science Ltd.

October 27, 2022

CSC

About the course



The core message of the course

Georges Matheron

"Illegitimate use of scientific concepts beyond the limits within which they have an operative meaning is nothing else but a surreptitious passage into metaphysics"

Course agenda

Core concepts

Supervised Learning

SL Models

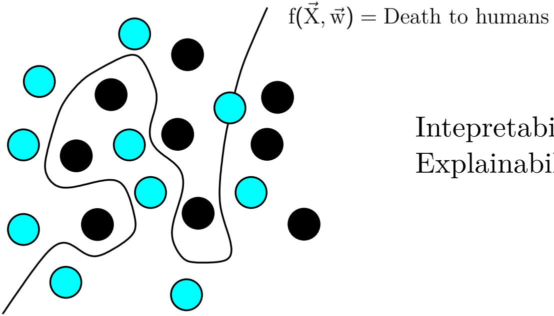
Practicalities

For the exercises, we will be using Python in notebooks.csc.fi.
Use password fun-1l7p88x5 to get into the workspace

Slides at
https://github.com/bilbrait/fundamentals-machine-learning.
Do not download excersises. They will be download automatically in notebooks.

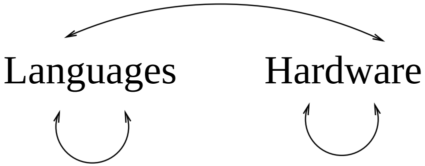
Time	Topic
09:00–09:45	Course introduction
09:45–10:00	Break
10:00–10:30	Supervised Learning
10:30–11:00	Exercise
11:00–12:00	Lunch
12:00–12:30	Models (part 0)
12:30–13:00	Exercise
13:00–13:15	Break
13:15–13:45	Models (part 1)
13:45–14:15	Exercise
14:15–14:45	Break
14:45–15:15	Model selection
15:15–15:45	Exercise

Difficulty of interpretation



Intepretability or Explainability?

What is Artificial Intelligence?



Algorithm 1: Describe what is an algorithm

Result: Definition of an algorithm

Data: What is is an algorithm?

Define unique & unambiguous set of inputs $\vec{x} \in X$;

Define unique & unambiguous set of outputs $\omega \in \Omega$;

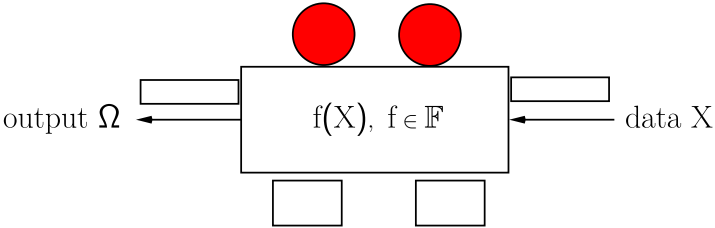
$N \leftarrow$ number of actions;

while $N \neq \infty$ **do**

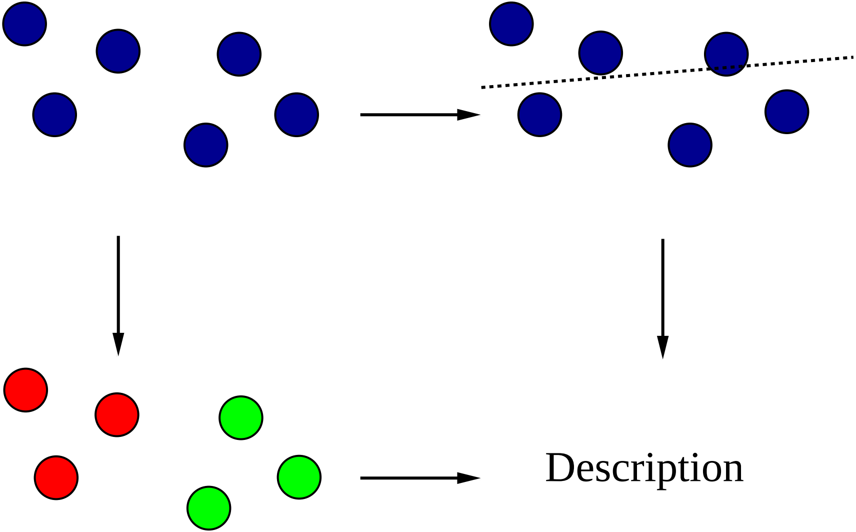
  Perform a set of unique & unambiguous actions on $\vec{x}$.

end

What is Machine Learning?



The role of statistics



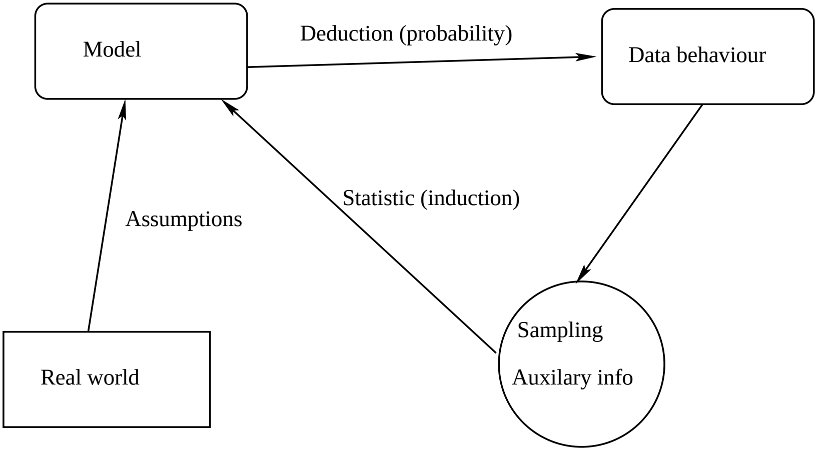
The role of Probability

Algorithm 2: Random number generator

Result: Output ω

$\omega \leftarrow 4$;   /* Chosen by a fair dice roll. Guaranteed random. */

Deductive & Inductive inference



States with & without memory

Governed by laws which behave predictably:

- Law of Gravity
- Randall cycle
- Human stupidity

States with & without memory

Governed by laws which behave predictably:

- Law of Gravity
- Randall cycle
- Human stupidity

Has no memory of the past:

- Stock markets
- Natural selection (Darwinian evolution)
- Musical compositions

Interpretation of probability

Logical (subjective):

$$\text{All ravens all black} \implies \text{All non-ravens are not black}$$

Interpretation of probability

Logical (subjective):

$$\text{All ravens all black} \implies \text{All non-ravens are not black}$$

Frequency (”empirical”):

$$\mathbb{P}(\mathrm{X} = \mathrm{x}) = \frac{\mathrm{x}}{\mathrm{N}}$$

Interpretations of statistics

Classical:

$$\frac{\mathrm{n}_{\mathrm{Raven}}}{\mathrm{N}_{\mathrm{Bird\ population}}},\ \mathrm{N}_{\mathrm{Bird\ population}} \to \infty$$

Interpretations of statistics

Classical:

$$\frac{\mathrm{n}_{\text{Raven}}}{\mathrm{N}_{\text{Bird population}}},\ \mathrm{N}_{\text{Bird population}} \to \infty$$

Subjective:

$$\mathbb{P}(\text{Duck} \mid \text{Quaks}) = \frac{\mathbb{P}(\text{Quaks} \mid \text{Duck})\mathbb{P}(\text{Duck})}{\mathbb{P}(\text{Quaks})}$$

Interpretations of statistics

Classical:

$$\frac{n_{\text{Raven}}}{N_{\text{Bird population}}}, \; N_{\text{Bird population}} \to \infty$$

Subjective:

$$\mathbb{P}(\text{Duck} \mid \text{Quaks}) = \frac{\mathbb{P}(\text{Quaks} \mid \text{Duck})\mathbb{P}(\text{Duck})}{\mathbb{P}(\text{Quaks})}$$

Utility:

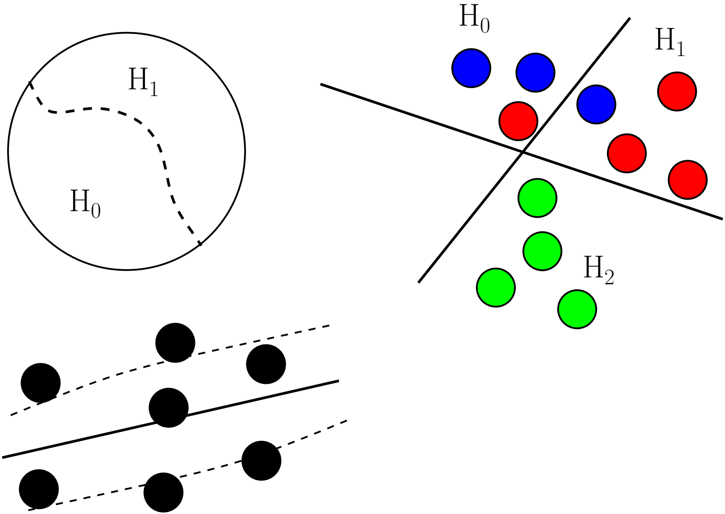
$$\mathcal{L}(\text{X} = \text{coin toss}) = \begin{cases} \$100 & \text{if X} = \text{heads} \\ \$1 & \text{if X} = \text{tails} \end{cases}$$

Interpretations of measurement

Using statistics which are invariant under permissible transformations.

- Nominal: one-to-one
- Ordinal: monotonic increasing
- Interval: linear transformations
- Ratio: similarity transformations

Different facets of statistics



Machine Learning and Probability

Disjunctive Normal Form

$$c_0 \wedge c_1 \wedge \cdots \wedge c_r,\ r \in \mathbb{Z}_+^n,$$
$$c_i \stackrel{\text{def}}{=} l_0 \vee l_1 \vee \cdots \vee l_{j_i},\ l \in \{0, 1\}$$

Conjuctive Normal Form

$$m_0 \vee m_1 \vee \cdots \vee m_r,\ r \in \mathbb{Z}_+^n,$$
$$m_i \stackrel{\text{def}}{=} l_0 \wedge l_1 \wedge \cdots \wedge l_{j_i},\ l \in \{0, 1\}$$

Estimation versus Optimization

Estimation:

$$\prod_{i=0}^{n-1} f(x_i \mid \theta) \stackrel{\text{def}}{=} L(\vec{x} \mid \theta), \ \vec{x} \in \mathbb{F}^n$$

Estimation versus Optimization

Estimation:

$$\Pi_{i=0}^{n-1} f(x_i \mid \theta) \stackrel{\text{def}}{=} L(\vec{x} \mid \theta), \ \vec{x} \in \mathbb{F}^n$$

Optimization:

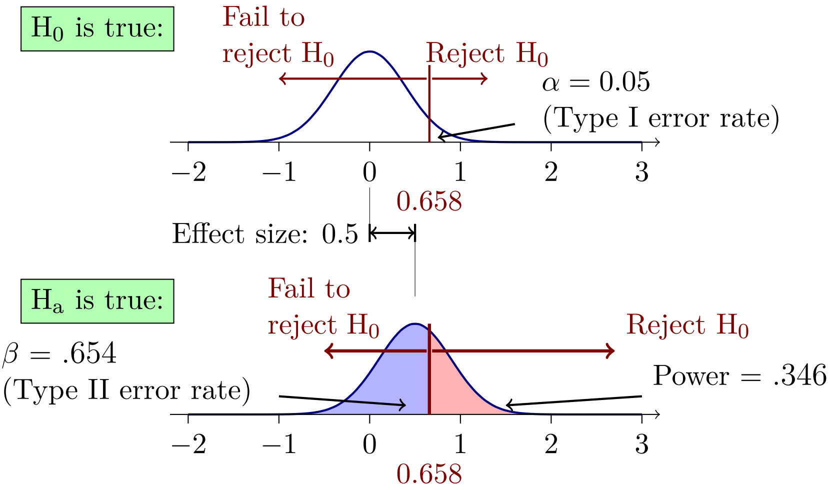
$$\hat{x} \leftarrow \underset{\vec{x} \in X \subset \mathbb{F}^n}{\arg\min} f(\vec{x}), \ \text{s.t.} \ A\vec{x} = \vec{b}, \ A \in \mathbb{F}^{n \times n}, \vec{b} \in \mathbb{F}^n$$

Supervised Learning

Statistical Inference

- Conditional density function: $\mathrm{p}(\vec{\mathrm{y}}|\vec{\mathrm{x}}) = \frac{\mathrm{p}(\vec{\mathrm{x}},\vec{\mathrm{y}})}{\mathrm{p}(\vec{\mathrm{x}})}$
- Regression: $\mathrm{r}(\vec{\mathrm{x}}) = \int \vec{\mathrm{y}}\ \mathrm{p}(\vec{\mathrm{y}}|\vec{\mathrm{x}})\mathrm{d}\vec{\mathrm{y}}$
- Density ration function: $\mathrm{R}(\vec{\mathrm{x}}) = \frac{\mathrm{p}(\vec{\mathrm{x}}_{\mathrm{num}}}{\mathrm{p}(\vec{\mathrm{x}}_{\mathrm{dem}}}$

Frequentist's approach to inference

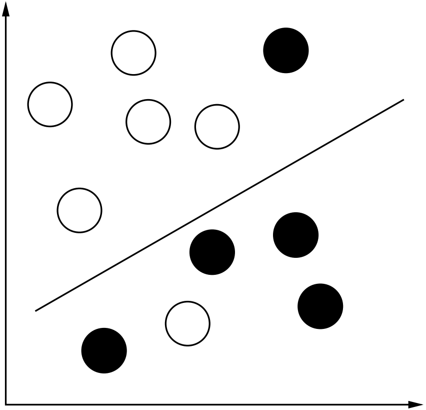


$X\vec{w} = \vec{y}$

$$
\begin{bmatrix}
x_{0,0} & x_{0,1} & x_{0,2} & \cdots & x_{0,n-1} \\
x_{0,0} & x_{0,1} & x_{0,2} & \cdots & x_{0,n-1} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
x_{m-1,0} & x_{m-1,1} & x_{m-1,2} & \cdots & x_{m-1,n-1}
\end{bmatrix}
\begin{bmatrix}
w_0 \\
w_1 \\
\\
w_{n-1}
\end{bmatrix}^T
=
\begin{bmatrix}
y_0 \\
y_1 \\
\\
y_{m-1}
\end{bmatrix}
$$

$$
X^{m \times n} = \begin{cases} m > n & \text{(overdetermined)} \\ n \gg m & \text{(underdetermined)} \end{cases}
$$

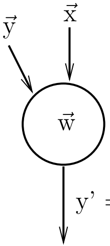
Statistical Discrimination and Regression



$$f(\vec{x}, \vec{w}) = \vec{w}^t\vec{x} + b$$

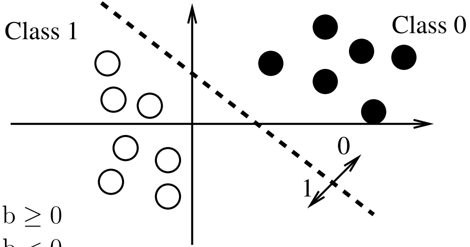
$$f(\vec{x}, \vec{w}) = (\vec{w}^t\vec{x} + b - \vec{y})^2$$

Perceptron model



$$y' = \begin{cases} 1 & : \sum_{i=0}^{n-1} w_i x_i + b \geq 0 \\ 0 & : \sum_{i=0}^{n-1} w_i x_i + b < 0 \end{cases}$$

(a)



(b)

Inductive inference from empirical data

Given a traning set $\mathrm{X_D}$, evaluate

$$\int \mathcal{L}(\mathrm{f}(\vec{\mathrm{X}}_{\mathrm{D}}, \alpha^*), \omega) \; \mathrm{dF}(\vec{\mathrm{X}}_{\mathrm{D}}), \;\; \alpha^* \in \Lambda$$

Inductive inference from empirical data

Given a traning set X_D, evaluate

$$\int \mathcal{L}(f(\vec{X}_D, \alpha^*), \omega) \, dF(\vec{X}_D), \;\; \alpha^* \in \Lambda$$

$$\frac{1}{\#\text{training samples}} \sum_{i=0}^{\#\text{training samples - 1}} \mathcal{L}(f_i(\vec{X}_D, \alpha^*), \omega_i), \;\; \alpha^* \in \Lambda$$

Two types of inductive inference

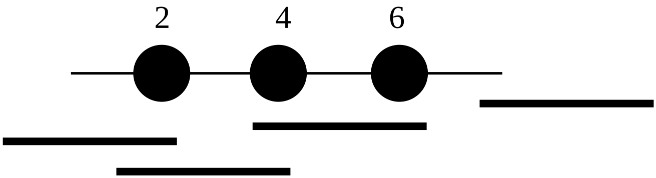
Inductive learning: find $\mathcal{L}(\mathrm{f}(\vec{\mathrm{X}}_{\mathrm{D}}, \alpha^*), \omega)$ which describes as many points as allowed by $\mathcal{L}$.

Two types of inductive inference

Inductive learning: find $\mathcal{L}(\mathrm{f}(\vec{\mathrm{X}}_{\mathrm{D}}, \alpha^*), \omega)$ which describes as many points as allowed by $\mathcal{L}$.

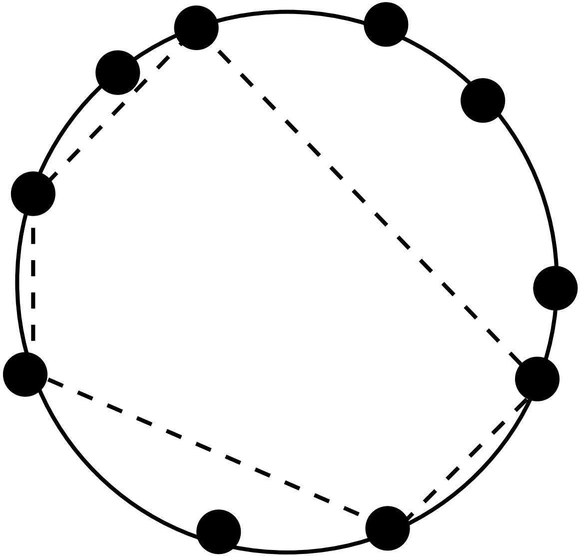
Transductive learning: find $\mathcal{L}(\mathrm{f}_{\mathrm{i}}(\vec{\mathrm{X}}_{\mathrm{D}}, \alpha^*), \omega),\ \mathrm{i} = 0, \ldots$

Sample complexity: Vapnik-Chernoviks dimension of 2

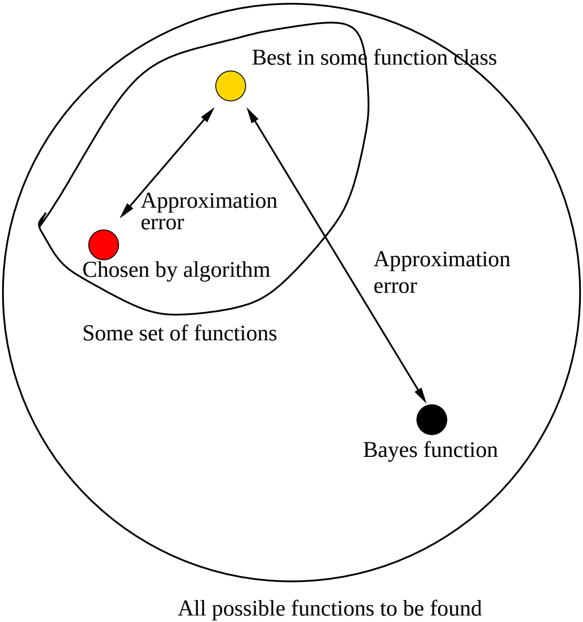


{2,4} Ok!   {4,6} Ok!   {2,6} No!

Sample complexity: Vapnik-Chernoviks dimension of ∞



Rademacher complexity



Decomposition of $\mathcal{L} : \hat{\omega} \leftarrow \mathbb{R}$ (special case)

$$\mathcal{L}(\mathrm{f}(\vec{\mathrm{x}}), \omega) = (\vec{\mathrm{x}}^{\mathrm{T}}\mathrm{w} - \omega)^2, \qquad \vec{\mathrm{x}} \in \mathcal{X}$$

$$\mathbb{V}[\hat{\omega}] = \min_{\mu} \mathbb{E}_{\hat{\omega}}[\hat{\omega} - \mu]^2 \qquad \text{(Variance)}$$

$$\mathbb{V}^{\mathrm{S}}[\hat{\omega}] = \operatorname*{argmin}_{\mu} \mathbb{E}_{\hat{\omega}}[\hat{\omega} - \mu]^2 \qquad \text{(SystematicVariance)}$$

$$(\mathbb{E}_{\hat{\omega}}[\hat{\omega}] - \mathbb{E}_{\omega}[\omega])^2 = (\mathbb{V}^{\mathrm{S}}[\hat{\omega}] - \mathbb{V}^{\mathrm{S}}[\omega])^2 \qquad \text{(Bias)}$$

Generalized $\mathcal{L}$

Loss function		
Squared error	General error	
$\mathbb{E}_{\hat{\omega}}[\hat{\omega} - \mathbb{E}(\hat{\omega})]^2$	$\mathbb{E}_{\hat{\omega}}[\mathcal{L}(\hat{\omega}, \mathbb{V}^{\mathrm{S}}[\hat{\omega}])]$	(Variance)
$\underset{\mu}{\operatorname{argmin}}\ \mathbb{E}_{\hat{\omega}}[\hat{\omega} - \mu]^2$	$\underset{\mu}{\operatorname{argmin}}\ \mathbb{E}_{\hat{\omega}}[\mathcal{L}(\hat{\omega}, \mu)]$	
$(\mathbb{E}_{\hat{\omega}}[\hat{\omega}] - \mathbb{E}_{\omega}[\omega])^2$	$\mathcal{L}(\mathbb{V}^{\mathrm{S}}[\omega], \mathbb{V}^{\mathrm{S}}[\hat{\omega}])$	(Bias^2)

$$\mathbb{V}^{\mathrm{S}}[\cdot] \overset{\text{def}}{=} \text{Systematic Variance}$$

Desired properties of $\mathcal{L}$

1. In the special case, use the general forms of variance and bias

Desired properties of $\mathcal{L}$

1. In the special case, use the general forms of variance and bias
2. The variance of the estimator should depend on test set and not the design set.

Desired properties of $\mathcal{L}$

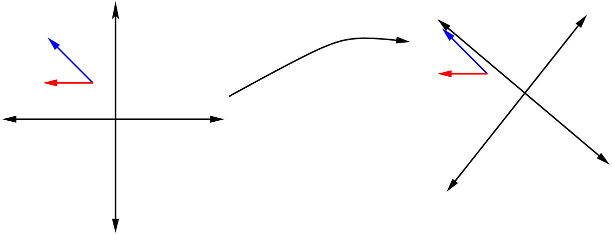
1. In the special case, use the general forms of variance and bias
2. The variance of the estimator should depend on test set and not the design set.
3. The bias of the estimator should depend on on systematic bias of design and test set.

Solving numerical extremas

$$\vec{x}_{k+1} = \vec{x}_k + H^k(\vec{y} - A\vec{x}_{k-1}), \;\; k = 0..$$

Solving numerical extremas

$$\vec{\mathrm{x}}_{\mathrm{k}+1} = \vec{\mathrm{x}}_{\mathrm{k}} + \mathrm{H}^{\mathrm{k}}(\vec{\mathrm{y}} - \mathrm{A}\vec{\mathrm{x}}_{\mathrm{k}-1}),\ \mathrm{k} = 0..$$



Artificial Neural Networks & Support Vector Machines

Neurocomputing: Graphical models

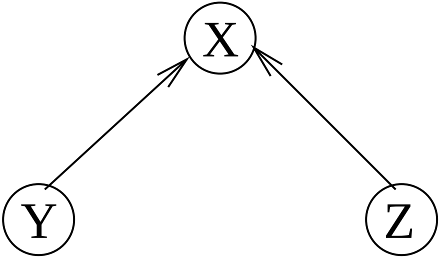
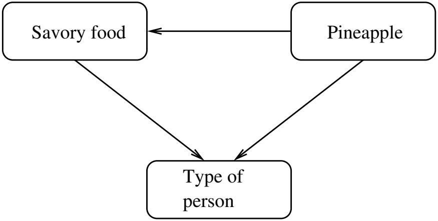


Figure: Bayesian network

Neurocomputing: Graphical models

Pineapple	Savory food F	Savory food T
F	1−a	a
T	1−b	b



Pineapple F	Pineapple T
1−a	a

Savory food	Pineapple	
F	F	Up standing fellow
F	T	Up standing fellow
T	F	Up standing fellow
T	T	Human garbage

Figure: Bayesian network

Neurocomputing: Graphical models

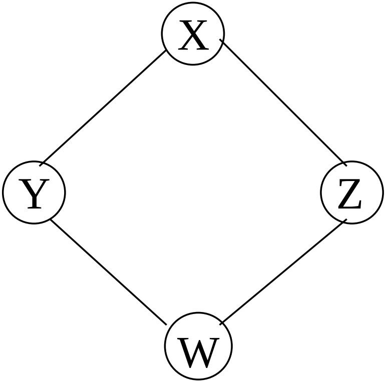


Figure: Markov network

Neurocomputing: Graphical models

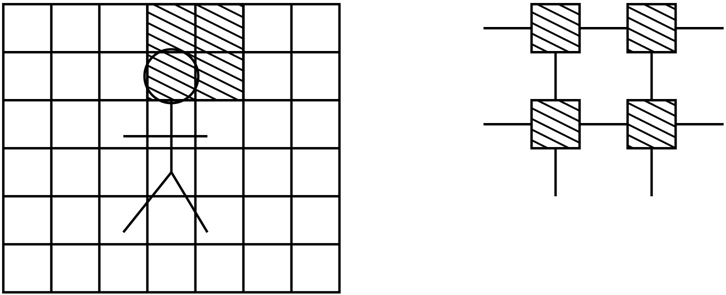
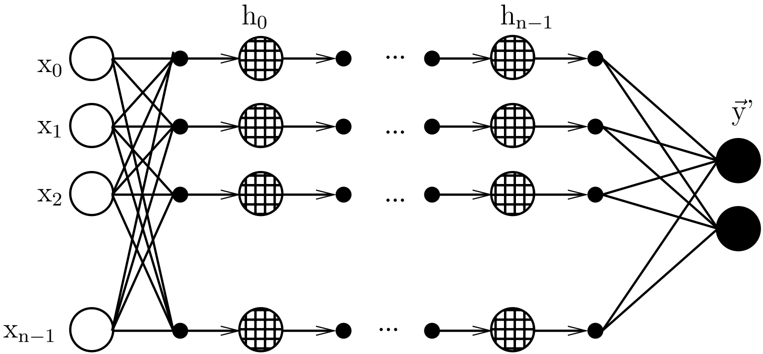
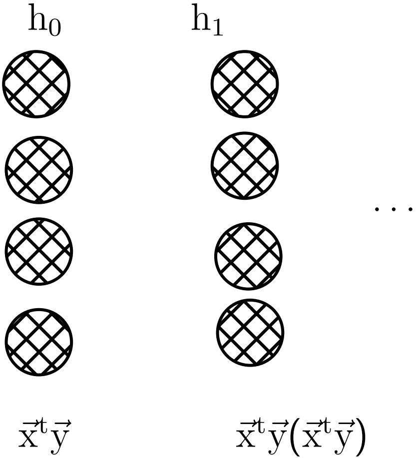


Figure: Markov network

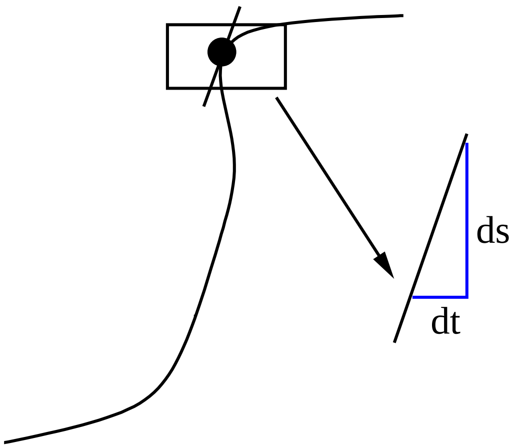
Neurocomputing: Artificial Neural Networks



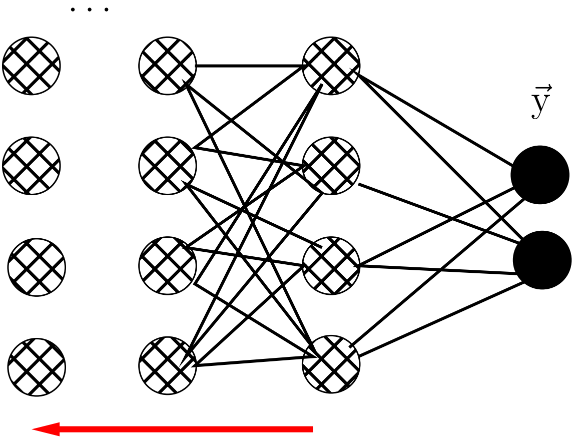
Neural Layers as Spanned Spaces



Pause: what is a derivate



Backpropagation



Interpretation of units: regression

$$(\vec{x}^t\vec{w} + \vec{b} - \vec{y}^2_)$$

$g(\vec{x}^t\vec{w})$ as an activation unit. Setting $g(a) = a$, implies linear optimization.

Interpretation: output units describe the variance of $p(\vec{y}|\vec{x})$.

Interpretation of units: discrimination

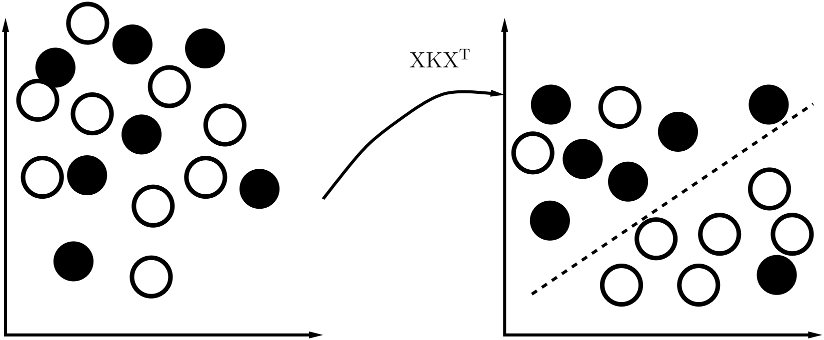
Becomes a Bayesian interpretation of $\mathrm{y_k}(\vec{\mathrm{x}}) = \mathbb{P}(\vec{\mathrm{x}}|\mathcal{C}_\mathrm{k})$

Using the $(\vec{\mathrm{x}}^{\mathrm{t}}\vec{\mathrm{w}} + \vec{\mathrm{b}} - \vec{\mathrm{y}}^2_)$ in a discrimination context, ouput units is the total covariance matrix of the training data.

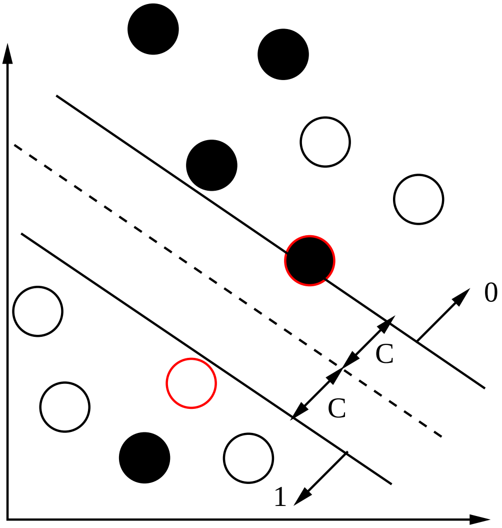
Interpretation of units

Lesson: interpretation depends on what statistical problem is in question, what error (loss) function is used.

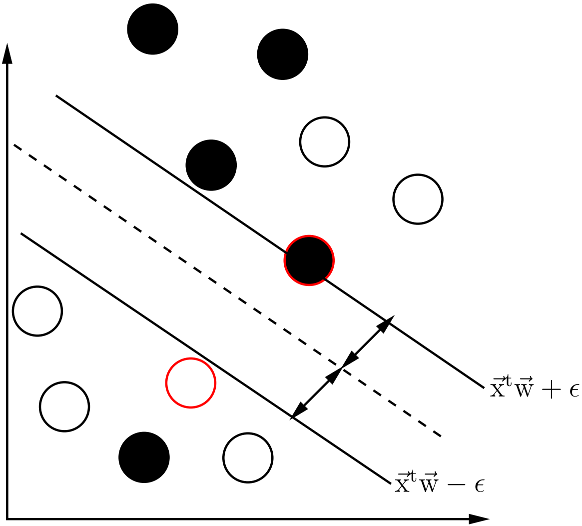
SVM - Kernel mappings



Max margins



Least-squares approach with S.V.M



Closer look at Optimization or Search

$$\underset{\vec{\mathrm{w}}}{\text{minimize}}\ \phi_\gamma(\vec{\mathrm{w}}) = \text{cost of search}(\vec{\mathrm{w}}) + \gamma \times \text{give penalty}(\vec{\mathrm{w}})$$

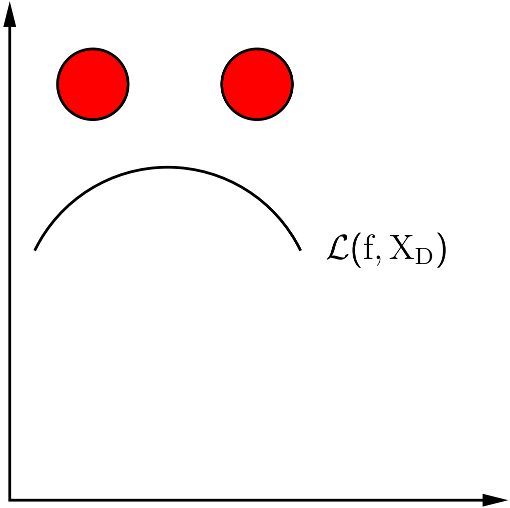
Closer look at Optimization or Search

$$\underset{\vec{\mathrm{w}}}{\text{minimize}}\ \phi_\gamma(\vec{\mathrm{w}}) = \text{cost of search}(\vec{\mathrm{w}}) + \gamma \times \text{give penalty}(\vec{\mathrm{w}})$$

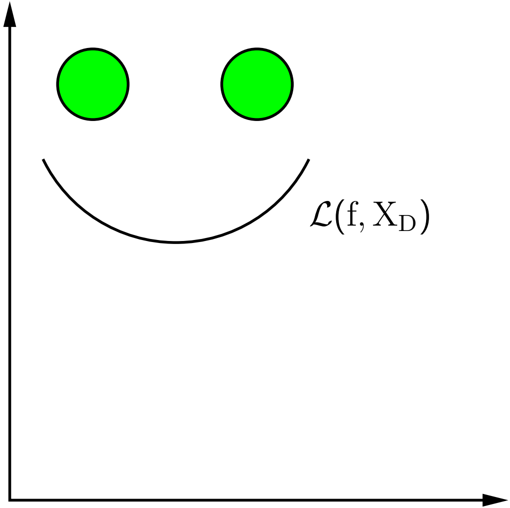
$$\underset{\vec{\mathrm{w}},\mathrm{b},\vec{\epsilon}}{\text{minimize}}\ \frac{1}{2}\vec{\mathrm{w}}^{\mathrm{t}}\vec{\mathrm{w}} + \mathrm{C}\sum_{\mathrm{i}}^{\mathrm{m}} \epsilon_{\mathrm{i}}$$

$$\text{subject to } \mathrm{y_i}(\vec{\mathrm{w}}^{\mathrm{t}}\vec{\mathrm{x}} + \mathrm{b}) \geq 1 - \epsilon_{\mathrm{i}},\ \epsilon_{\mathrm{i}} \geq 0,\ 1 \leq \mathrm{i} \leq \mathrm{m}$$

Regularization: Why?

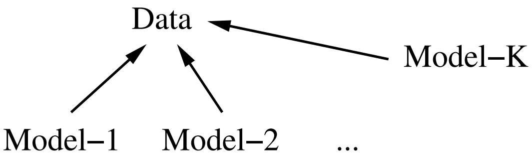


Regularization: Why?



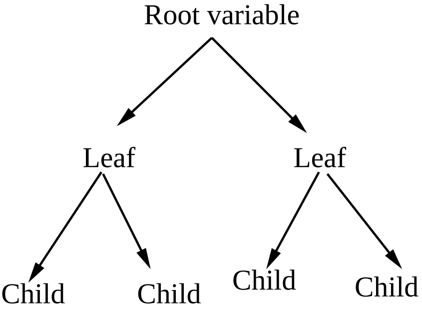
Ensemble Models

Ensemble Models



Expectation(Model–1 + Model–2 + ... + Model–K)

Decision Tree



Entropy: $\mathrm{p}_{\mathrm{Leaf}} = \text{Count class specific data instances}$

$\mathrm{H}(\mathrm{Leaf}) = -\sum \mathrm{p}_{\mathrm{Leaf}} \log(\mathrm{p}_{\mathrm{Leaf}})$

Binary tree: types & properties

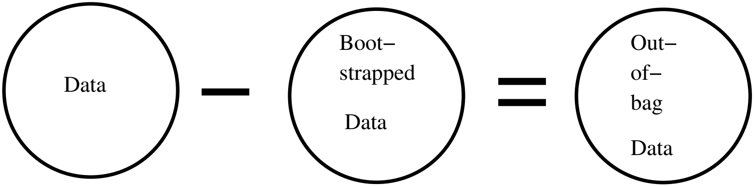
- A binary has always a single root node and all nodes have at least two child nodes.
- Proper tree: nodes have either 0 or 2 childs.
- Balanced: the height of left and right branches do not differ by one level.
- Requires relatively little storrage if balanced: O(logn) bits, where n is the height of the tree.
- Searching in a complete binary tree requires $O(|V| + |E|)$ operations.

Bootstrapping

A Poor man's Bayes distribution

The data is sampled B times, after which the resulting data is fitted with, for example, a cubic spline.

Bagging



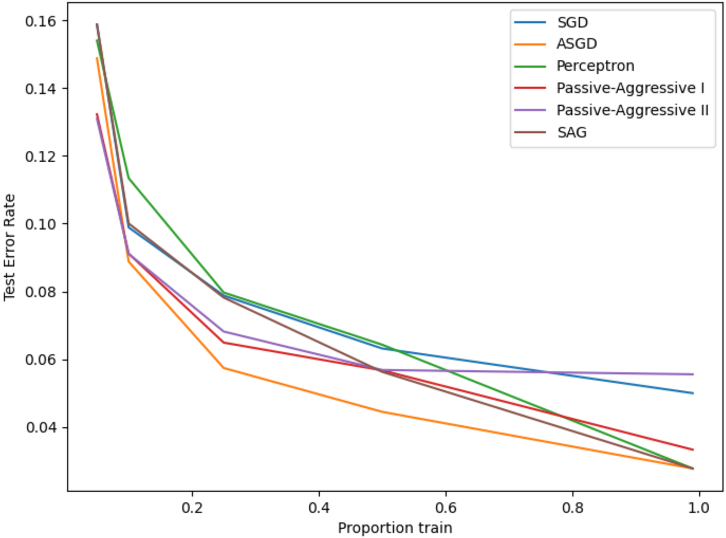
Model Selection

Model Selection

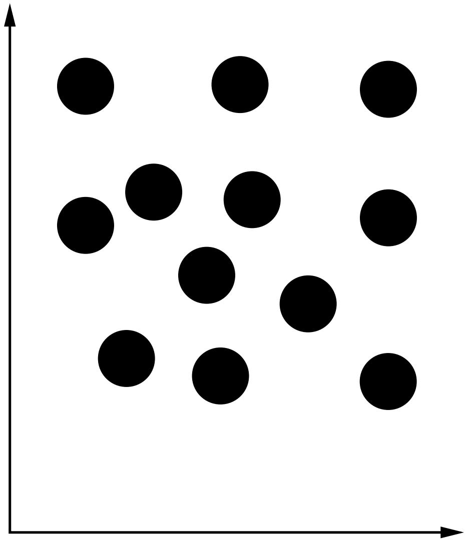
$$\underset{\vec{w},b,\vec{\epsilon}}{\text{minimize}}\ \frac{1}{2}\vec{w}^t\vec{w} + C\sum_{i}^{m}\epsilon_i$$

$$\text{subject to } y_i(\vec{w}^t\vec{x} + b) \geq 1 - \epsilon_i,\ \ \epsilon_i \geq 0,\ \ 1 \leq i \leq m$$

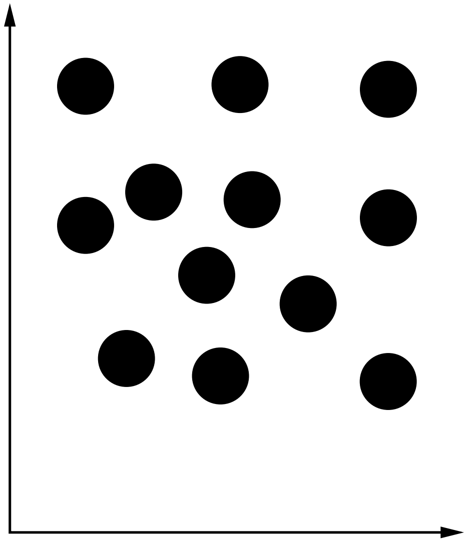
Comparing different models



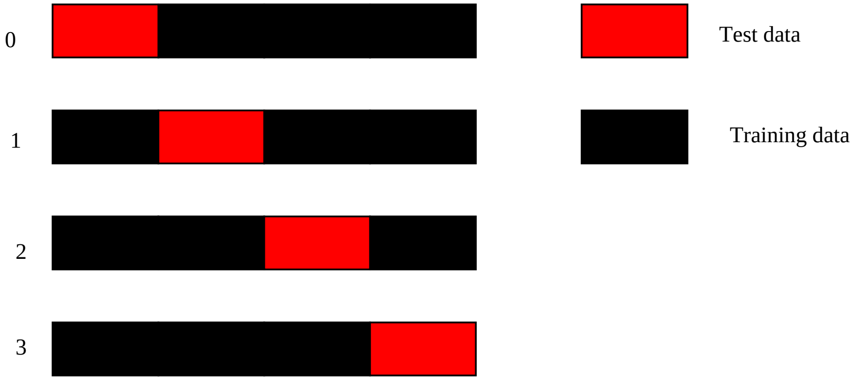
Grid search



Random search



Cross-validation



Bayesian analysis (NOT Bayesian Optimization)

Live demo