Learning Agent:

Naïve Bayes, Neural Network, Support Vector Machines

IF-3054

Teknik Informatika ITB

Overview: Learning Agent

- Tujuan learning
- Teknik Learning: kNN, ID3, NaiveBayes, NN, SVM
- Perbedaan antar teknik learning
  - Proses utama learning
  - Proses klasifikasi (inferensi)
  - Representasi model hasil learning
  - Representasi keputusan

NAÏVE BAYES

Naïve Bayes

$$v_{NB} = \arg\max_{v_j \in \{yes,no\}} P(v_j) \prod_i P(a_i|v_j)$$

- $P(v_j)$: probabilitas kelas v_j
- $P(a_i|v_j)$: probabilitas atribut a_i pada v_j

Contoh Data set: Play Tennis

outlook	temp.	humidity	windy	play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes

outlook	temp.	humidity	windy	play
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

Frekuensi setiap nilai atribut

outlook			temperature			humidity			windy			play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2	3	hot	2	2	high	3	4	false	6	2	9	5
overcast	4	0	mild	4	2	normal	6	1	true	3	3		
rainy	3	2	cool	3	1								

Model Probabilitas

outlook			temperature			humidity			windy			play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2/9	3/5	hot	2/9	2/5	high	3/9	4/5	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	mild	4/9	2/5	normal	6/9	1/5	true	3/9	3/5		
rainy	3/9	2/5	cool	3/9	1/5								

Klasifikasi Unseen Example

outlook	temp.	humidity	windy	play
sunny	cool	high	true	?

$$v_{NB} = \arg\max_{v_j \in \{\text{yes,no}\}} P(v_j) \prod_i P(a_i|v_j)$$

$$= \arg\max_{v_j \in \{\text{yes,no}\}} P(v_j) P(\text{outlook}=\text{sunny}|v_j) P(\text{temp}=\text{cool}|v_j)$$

$$P(\text{humidity}=\text{high}|v_j) P(\text{windy}=\text{true}|v_j)$$

1. Kalikan probabilitas semua atribut untuk setiap kelas
2. Hasil 1 dikalikan dengan probabilitas setiap kelas
3. Klasifikasi: kelas dengan probabilitas maksimum

Proses Klasifikasi

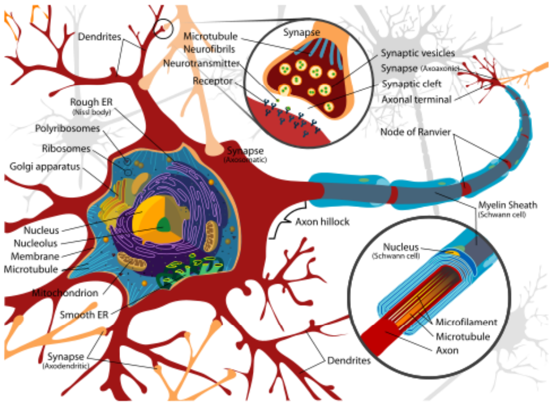
$$P(\text{play} = \text{yes}) = 9/14 \quad P(\text{play} = \text{no}) = 5/14$$

$$P(\text{yes})P(\text{sunny}|\text{yes})P(\text{cool}|\text{yes})P(\text{high}|\text{yes})P(\text{true}|\text{yes})$$
$$= 9/14 \cdot 2/9 \cdot 3/9 \cdot 3/9 \cdot 3/9 = 0.0053$$
$$P(\text{no})P(\text{sunny}|\text{no})P(\text{cool}|\text{no})P(\text{high}|\text{no})P(\text{true}|\text{no})$$
$$= 5/14 \cdot 3/5 \cdot 1/5 \cdot 4/5 \cdot 3/5 = 0.0206$$

$$\begin{aligned} v_{NB} &= \arg\max_{v_j \in \{\text{yes},\text{no}\}} P(v_j)P(\text{sunny}|v_j)P(\text{cool}|v_j)P(\text{high}|v_j)P(\text{true}|v_j) \\ &= \text{no} \end{aligned}$$

NEURAL NETWORK

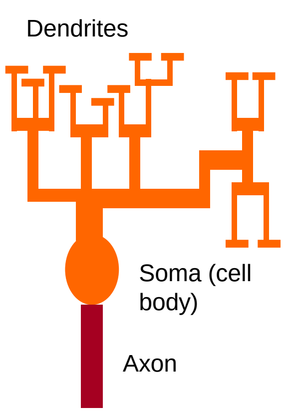
Biological Neuron



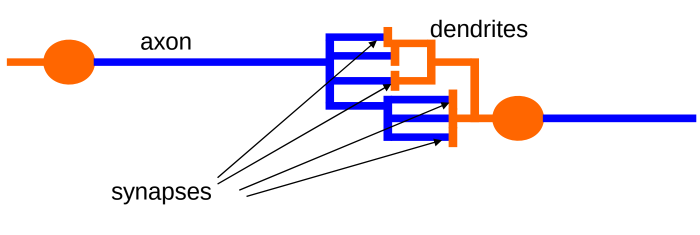
http://www.chemistry.ucsc.edu/~lokey/108A_10/images/neuron.png

- A neuron has
  - A branching input (dendrites)
  - A branching output (the axon)
- The information circulates from the dendrites to the axon via the cell body
- Axon connects to dendrites via synapses
  - Synapses vary in strength
  - Synapses may be excitatory or inhibitory

Artificial Neural Networks
- Introduction -
Peter Andras
peter.andras@ncl.ac.uk

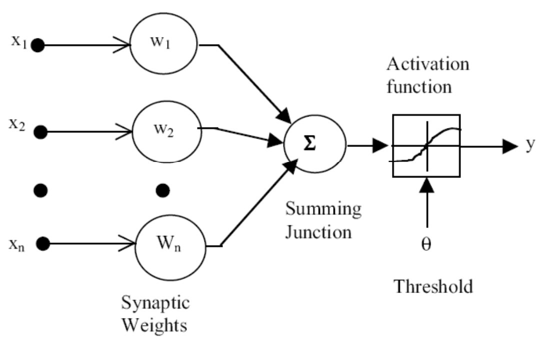


The information transmission happens at the synapses.



What is an artificial neuron ?

- Parameterized function with restricted output range
- Example: simple/single unit perceptron

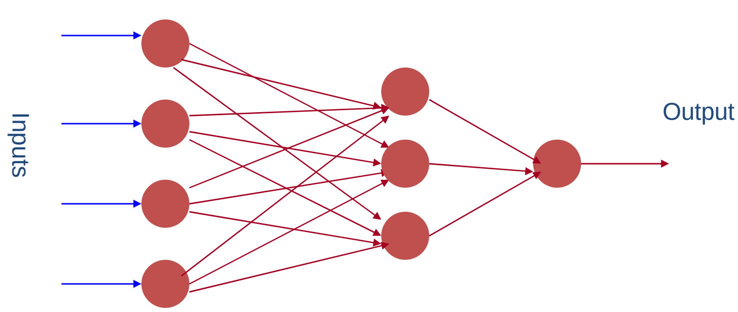


$$y = f\left(w_0 + \sum_{i=1}^{n-1} w_i x_i \right)$$

f: activation function

Tutorial on Neural Networks
Prévotet Jean-Christophe, University of Paris VI, FRANCE

Neural network mathematics



$$y_1^1 = f(x_1, w_1^1) \quad y_2^1 = f(x_2, w_2^1) \quad y_3^1 = f(x_3, w_3^1) \quad y_4^1 = f(x_4, w_4^1)$$

$$y^1 = \begin{pmatrix} y_1^1 \\ y_2^1 \\ y_3^1 \\ y_4^1 \end{pmatrix}$$

$$y_1^2 = f(y^1, w_1^2) \quad y_2^2 = f(y^1, w_2^2) \quad y_3^2 = f(y^1, w_3^2)$$

$$y^2 = \begin{pmatrix} y_1^2 \\ y_2^2 \\ y_3^2 \end{pmatrix}$$

$$y_{Out} = f(y^2, w_1^3)$$

Artificial Neural Networks
- Introduction -
Peter Andras
peter.andras@ncl.ac.uk

Neural network mathematics

Neural network: input / output transformation

$$y_{out} = F(x, W)$$

W is the matrix of all weight vectors.

Artificial Neural Networks
- Introduction -
Peter Andras
peter.andras@ncl.ac.uk

Neural Network Learning

- Data: set of value pairs: (x^t, y_t), $y_t=g(x^t) + z_t$; z_t is random measurement noise.
- Objective: find a neural network that represents the input / output transformation (a function) F(x,W) such that F(x,W) approximates g(x) for every x

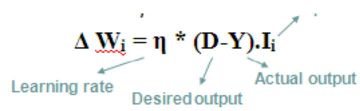
Learning Process

Error measure:

$$E = \frac{1}{N}\sum_{t=1}^{N}(F(x_t;W) - y_t)^2$$

Rule for changing the synaptic weights:

$$\Delta w_i^j = -c \cdot \frac{\partial E}{\partial w_i^j}(W)$$



$$w_i^{j,new} = w_i^j + \Delta w_i^j$$

c is the learning parameter (usually a constant)

Learning with a perceptron

Perceptron: $y_{out} = w^T x$

Data: $(x^1, y_1), (x^2, y_2), \ldots, (x^N, y_N)$

Error: $E(t) = (y(t)_{out} - y_t)^2 = (w(t)^T x^t - y_t)^2$

Learning:

$$w_i(t+1) = w_i(t) - c \cdot \frac{\partial E(t)}{\partial w_i} = w_i(t) - c \cdot \frac{\partial (w(t)^T x^t - y_t)^2}{\partial w_i}$$

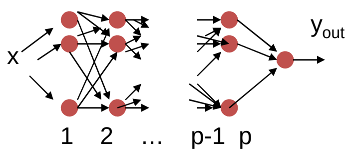
$$w_i(t+1) = w_i(t) - c \cdot (w(t)^T x^t - y_t) \cdot x_i^t$$

$$w(t)^T x = \sum_{j=1}^{m} w_j(t) \cdot x_j^t$$

Learning with MLP neural networks

MLP neural network:

with p layers



$$y_k^1 = \frac{1}{1+e^{-w^{1kT}x-a_k^1}},\ k=1,\dots,M_1$$

$$y^1 = (y_1^1,\dots,y_{M_1}^1)^T$$

$$y_k^2 = \frac{1}{1+e^{-w^{2kT}y^1-a_k^2}},\ k=1,\dots,M_2$$

$$y^2 = (y_1^2,\dots,y_{M_2}^2)^T$$

…

$$y_{out} = F(x;W) = w^{pT}y^{p-1}$$

Data: $(x^1, y_1), (x^2, y_2), \dots, (x^N, y_N)$

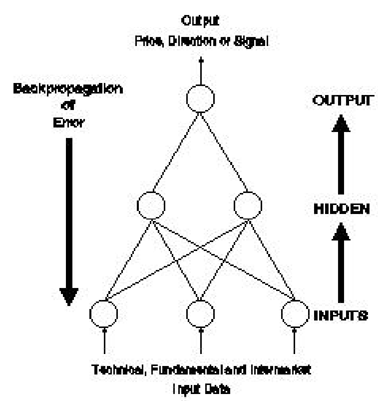
Error: $E(t) = (y(t)_{out} - y_t)^2 = (F(x^t;W) - y_t)^2$

It is very complicated to calculate the weight changes.

Learning with backpropagation

Solution of the complicated learning:

- calculate first the changes for the synaptic weights of the output neuron;
- calculate the changes backward starting from layer p-1, and propagate backward the local error terms.

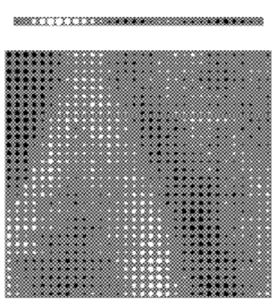


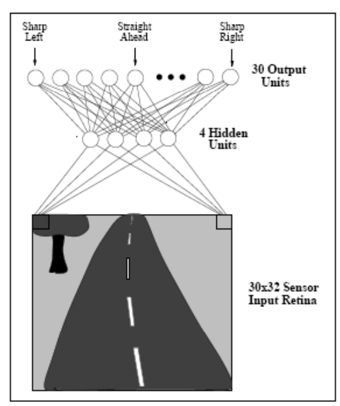
The method is still relatively complicated but it is much simpler than the original optimisation problem.

Artificial Neural Network

- ANN: most effective learning methods for complex real world sensor data
  - ALVINN, face recognition, handwritten recognition
  - Financial prediction
- Well suited to problems in which the training data corresponds to noisy, complex sensor data (cameras, microphones)
  - Input/output: discrete, real value, vector of value
  - Human readability of result is not important

ALVINN: 70 mph





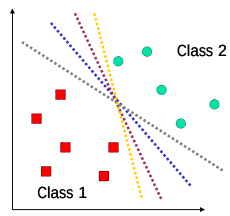
960 unit input → 30 output unit

SUPPORT VECTOR MACHINE

Support Vector Machine

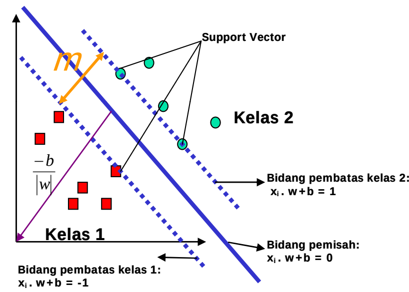
- Diperkenalkan tahun 1992 oleh Vapnik, Boser dan Guyon
- Relatif baru tetapi memiliki performansi yang lebih baik di berbagai aplikasi seperti *bioinformatics*, klasifikasi teks dan pengenalan tulisan tangan dan lain sebagainya

SVM pada *Linearly Separable Data*



Cari bidang pemisah dengan margin (m) terbesar

$$\min \frac{1}{2}|w|^2$$
$$s.t\ \ y_i(x_i.w+b)-1 \geq 0$$



- Bidang pemisah terbaik dengan margin terbesar memiliki generalisasi yang lebih baik
- 2 kelas dapat dipisahkan oleh sepasang bidang pembatas yang sejajar. Bidang pembatas pertama membatasi kelas pertama sedangkan bidang pembatas kedua membatasi kelas kedua

Bidang pemisah terbaik (2)

- Pencarian bidang pemisah terbaik dapat dirumuskan menjadi:

$$\min \frac{1}{2}|w|^2$$
$$s.t\ \ y_i(x_i.w+b)-1 \geq 0$$

- Supaya lebih mudah diselesaikan ubah ke formula lagrangian

$$\min_{w,b} L_p(w,b,\alpha) \equiv \frac{1}{2}|w|^2 - \sum_{i=1}^{n} \alpha_i y_i (x_i.w+b) + \sum_{i=1}^{n} \alpha_i$$
$$\alpha \geq 0$$

- Minimumkan Lp terhadap b dan w, diperoleh:

$$\frac{\partial}{\partial b} L_p(w,b,\alpha) = 0 \quad \Rightarrow \quad \sum_{i=1}^{n} \alpha_i y_i = 0$$

$$\frac{\partial}{\partial w} L_p(w,b,\alpha) = 0 \quad \Rightarrow \quad w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

26 02/10/2007

Bidang Pemisah Terbaik (3)

- Substitusi ke dalam Lp diperoleh *dual problem* Ld yang memiliki konstrain berbeda

$$L_D(\alpha) \equiv \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1,\, j=1}^{n} \alpha_i \alpha_j y_i y_j x_i . x_j \quad , \alpha_i > 0$$

- Karena merupakan dual problem maka

$$\min_{w,b} L_p = \max_{\alpha} L_D$$

- Selesaikan Ld dan Cari nilai α_i
- Data ke-i dengan nilai $\alpha_i > 0$ merupakan support vector
- Fungsi keputusan (persamaan bidang pemisah terbaik):

$$f(x_d) = \sum_{i=1}^{ns} \alpha_i y_i x_i . x_d + b$$

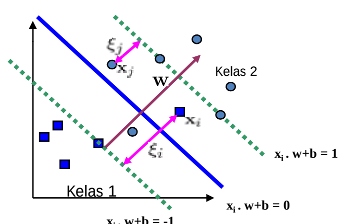
- Xd → data yang akan diklasifikasikan

$$\text{if} \;\; f(x) \geq +1 \;\; \text{then} \;\; y_i = +1, \;\; \text{Kelas} \;\; 2$$
$$\text{if} \;\; f(x) \leq -1 \;\; \text{then} \;\; y_i = -1, \;\; \text{Kelas} \;\; 1$$

- **Persamaan bidang pemisah hanya dipengaruhi oleh data support vector**

SVM pada *non linearly separable data*

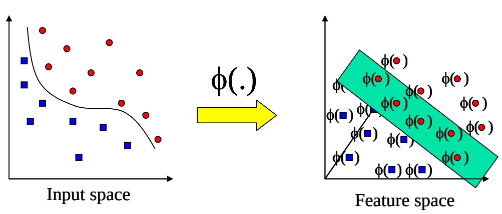
Bagaimana jika data tidak dapat dipisahkan secara linier?
Tambahkan variabel ξ_i (error dalam klasifikasi)
Parameter C ditentukan pengguna menyatakan besar penalti terhadap error



Pembelajaran:

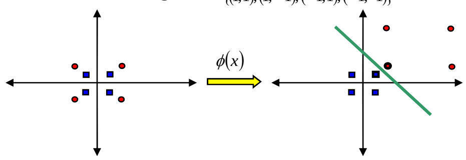
$$\min \frac{1}{2}|w|^2 + C\left(\sum_{i=1}^{n} \xi_i\right)$$
$$s.t \;\; x_i . w + b \leq -1 + \xi_i$$

Transformasikan data sehingga *linearly separable* di *feature space*



SVM pada *non linearly separable data (2)*

- Contoh:
- Misalkan dataset
  - Data kelas positif $\{(2,2),(2,-2),(-2,2),(-2,-2)\}$
  - Data kelas negatif $\{(1,1),(1,-1),(-1,1),(-1,-1)\}$



$$\phi(x_1,x_2)=\begin{cases}\sqrt{x_1^2+x_2^2}>2 \rightarrow \left(4-x_2+|x_1-x_2|,\ 4-x_1+|x_1-x_2|\right)\\ \sqrt{x_1^2+x_2^2}\le 2 \rightarrow (x_1,x_2)\end{cases}$$

- Dengan transformasi diperoleh
  - Data kelas positif $\{(2,2),(6,2),(6,6),(2,6)\}$
  - Data kelas negatif $\{(1,1),(1,-1),(-1,1),(-1,-1)\}$

SVM pada *non linearly separable data (3)*

Klasifikasi: $f(x)=\sum_{i=1}^{ns}\alpha_i y_i x_i . x+b \Rightarrow f(x)=\sum_{i=1}^{ns}\alpha_i y_i \phi(x_i)\phi(x)+b$

- Sulit untuk mengetahui $\phi(x)$ dan feature space biasanya memiliki dimensi yang lebih besar
- Solusinya "*kernel trick*", yang perlu diketahui adalah $K(x_i,x)=\phi(x_i)\phi(x)$
- Dengan fungsi K (fungsi Kernel), maka fungsi $\phi(x)$ tidak perlu diketahui

Klasifikasi:

$$f(x)=\sum_{i=1}^{ns}\alpha_i y_i K(x_i,x)+b$$

Fungsi Kernel yang umum digunakan:

Linear Kernel → $K(x_i,x_j)=x_i^T x_j$

Polynomial kernel→ $K(x_i,x_j)=(\gamma . x_i^T x_j+r)^p, \gamma>0$

RBF kernel → $K(x_i,x_j)=\exp\left(-\gamma|x_i-x_j|^2\right), \gamma>0$

Sigmoid kernel → $K(x_i,x_j)=\tanh(\gamma x_i^T x_j+r)$

Multi Class SVM

- SVM yang dijelaskan diatas hanya dapat mengklasifikasikan data ke dalam dua kelas → SVM biner
- Untuk klasfikasi multiclass gabungkan beberapa SVM biner.
- Ada 3 metode umum:
  - *One-against-all*
  - *One-against-one*
  - DAGSVM

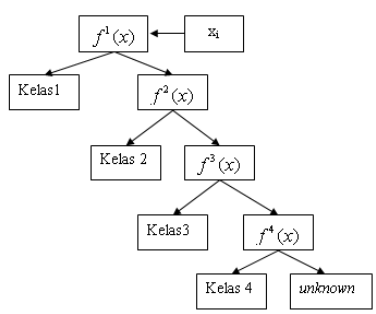
Multi Class SVM→*one-against-all*

- Dibangun k model klasifikasi (k adalah jumlah kelas)
- Setiap pelatihan model menggunakan data dari semua kelas
- Prediksi kelas data umumnya seperti gambar sebelah kanan atau berdasarkan nilai maksimum f(X)

Pembelajaran:

$y_i = 1$	$y_i = -1$	Hipotesis
Kelas 1	Bukan kelas 1	$f^1(x) = (w^1)x + b^1$
Kelas 2	Bukan kelas 2	$f^2(x) = (w^2)x + b^2$
Kelas 3	Bukan kelas 3	$f^3(x) = (w^3)x + b^3$
Kelas 4	Bukan kelas 4	$f^4(x) = (w^4)x + b^4$

Klasifikasi:



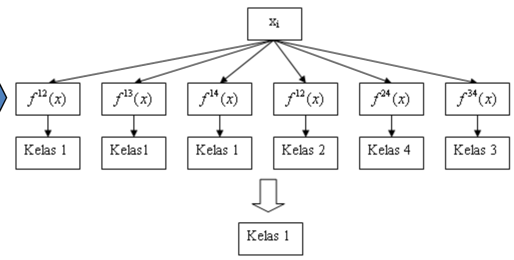
Multi Class SVM→*one-against-one*

- Dibangun $\frac{k(k-1)}{2}$ model klasifikasi (k adalah jumlah kelas)
- Setiap pelatihan model menggunakan data dari dua kelas
- Prediksi Kelas data dengan metode voting

Pembelajaran:

$y_i = 1$	$y_i = -1$	Hipotesis
Kelas 1	Kelas 2	$f^{12}(x) = (w^{12})x + b^{12}$
Kelas 1	Kelas 3	$f^{13}(x) = (w^{13})x + b^{13}$
Kelas 1	Kelas 4	$f^{14}(x) = (w^{14})x + b^{14}$
Kelas 2	Kelas 3	$f^{23}(x) = (w^{23})x + b^{23}$
Kelas 2	Kelas 4	$f^{24}(x) = (w^{24})x + b^{24}$
Kelas 3	Kelas 4	$f^{34}(x) = (w^{34})x + b^{34}$

Klasifikasi:



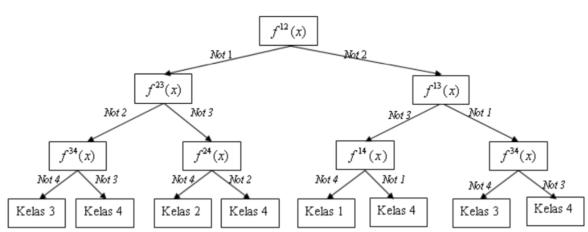
Multi Class SVM→DAG (*Directed Acyclic Graph*) SVM

- Proses pembelajaran sama dengan One-Against-One

Pembelajaran:

$y_i = 1$	$y_i = -1$	Hipotesis
Bukan Kelas 2	Bukan Kelas 1	$f^{12}(x) = (w^{12})x + b^{12}$
Bukan Kelas 3	Bukan Kelas 1	$f^{13}(x) = (w^{13})x + b^{13}$
Bukan Kelas 4	Bukan Kelas 1	$f^{14}(x) = (w^{14})x + b^{14}$
Bukan Kelas 3	Bukan Kelas 2	$f^{23}(x) = (w^{23})x + b^{23}$
Bukan Kelas 4	Bukan Kelas 2	$f^{24}(x) = (w^{24})x + b^{24}$
Bukan Kelas 4	Bukan Kelas 3	$f^{34}(x) = (w^{34})x + b^{34}$

Klasifikasi:

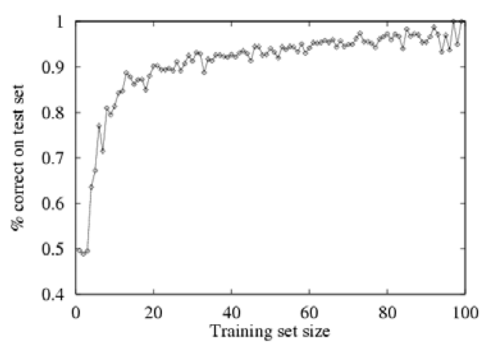


SVM Software

- LibSVM
  - Umum, dapat digunakan untuk berbagai aplikasi, tidak dioptimasi untuk SVM Linier (fungsi kernel linier)
  - Mendukung Multi Class SVM One-Against-One, One Class SVM dan Support Vector Regression
  - C++, Java, Phyton, C#, Matlab
- LibLinear
  - versi LibSVM yang dioptimasi untuk kernel linier
  - Mendukung Multi Class SVM One-Against-All
  - C++
- SVMLight
  - C++, populer dalam aplikasi klasifikasi teks
- Info tambahan tentang SVM:
  - http://www.kernel-machines.org
  - www.svms.org
  - http://agbs.kyb.tuebingen.mpg.de
  - http://support-vector-machines.org

Performansi Hipotesis

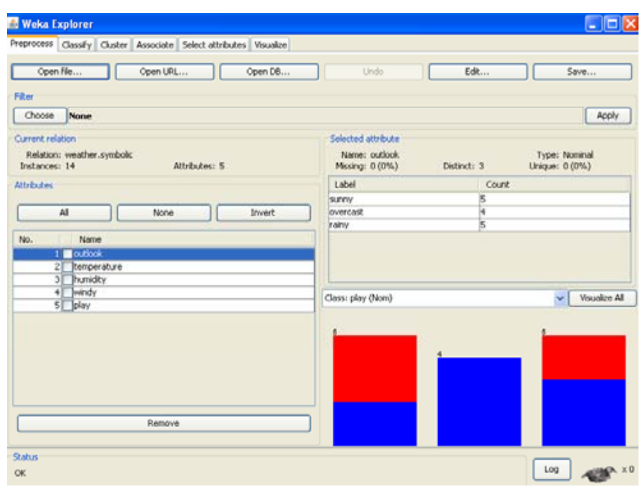
- Performansi hipotesis pada data training bukan indikator performansinya pada data baru
- Estimasi performansi hipotesis: performansi pada data tes
- Learning curve = % correct on test set as a function of training set size



Review: Learning Agent

- Teknik Learning: kNN, ID3, NaiveBayes, NN, SVM
- Perbedaan antar teknik learning
  - Proses utama learning
  - Proses klasifikasi (inferensi)
  - Representasi model hasil learning
  - Representasi keputusan

Tools WEKA



http://www.cs.waikato.ac.nz/ml/weka/

WEKA: Format Data

outlook	temp.	humidity	windy	play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

```
@relation weather.symbolic

@attribute outlook {sunny, overcast, rainy}
@attribute temperature {hot, mild, cool}
@attribute humidity {high, normal}
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,hot,high,FALSE,no
sunny,hot,high,TRUE,no
overcast,hot,high,FALSE,yes
rainy,mild,high,FALSE,yes
rainy,cool,normal,FALSE,yes
rainy,cool,normal,TRUE,no
overcast,cool,normal,TRUE,yes
sunny,mild,high,FALSE,no
sunny,cool,normal,FALSE,yes
rainy,mild,normal,FALSE,yes
sunny,mild,normal,TRUE,yes
overcast,mild,high,TRUE,yes
overcast,hot,normal,FALSE,yes
rainy,mild,high,TRUE,no
```

MLK/IF-ITB/2006 39

WEKA: Learning