Lecture 7:
Support Vector Machines

Isabelle Guyon
guyoni@inf.ethz.ch

References

- **An training algorithm for optimal margin classifiers**

Boser-Guyon-Vapnik, COLT, 1992
http://www.clopinet.com/isabelle/Papers/colt92.ps.Z

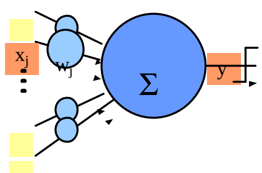
- **Book chapters 1 and 12**
- **Software LibSVM**

http://www.csie.ntu.edu.tw/~cjlin/libsvm/

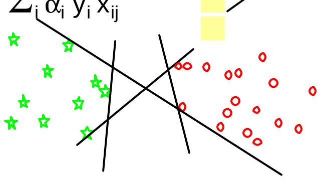
Perceptron Learning Rule

$w_j \leftarrow w_j + y_i x_{ij}$

If **x** is misclassified



$w_j = \Sigma_i \alpha_i y_i x_{ij}$

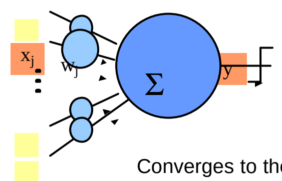


Converges to "one" solution classifying well all the examples.

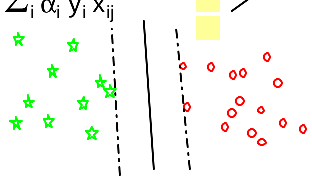
Optimum Margin Perceptron

$w_j \leftarrow w_j + y_i x_{ij}$

If **x** is the least well classified



$w_j = \Sigma_i \alpha_i y_i x_{ij}$



Converges to the optimum margin Perceptron (Mezard-Krauth, 1988).

QP formulation (vapnik, 1962).

Optimum Margin Solution

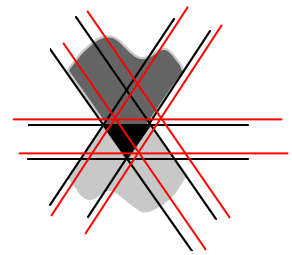
- Unique solution.
- Depends only on support vectors (SV)

$$w_j = \sum_{i \in SV} \alpha_i \, y_i \, x_{ij}$$

- SVs are examples closest to the boundary.
- Bound on leave-one-out error: $LOO \leq n_{SV}/m$
- Most "stable", good from MDL point of view.

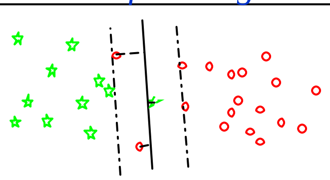
- But: sensitive to outliers and works only for linearly separable case.

Negative Margin



Multiple negative margin solutions,
which all give the same number of errors.

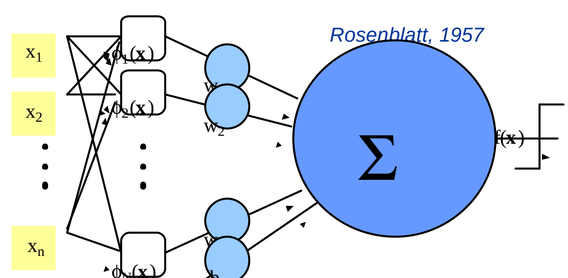
Soft-Margin



Examples within the margin area incurr a penalty and become non-marginal support vectors. Unique solution again (Cortes-Vapnik, 1995):

$$w_j = \sum_{i \in SV} \alpha_i \, y_i \, x_{ij}$$

Non-Linear Perceptron



$f(\mathbf{x}) = \mathbf{w} \bullet \Phi(\mathbf{x}) + b$

Kernel "Trick"

- $f(\mathbf{x}) = \mathbf{w} \bullet \Phi(\mathbf{x})$
- $\mathbf{w} = \sum_i \alpha_i y_i \Phi(\mathbf{x}_i)$

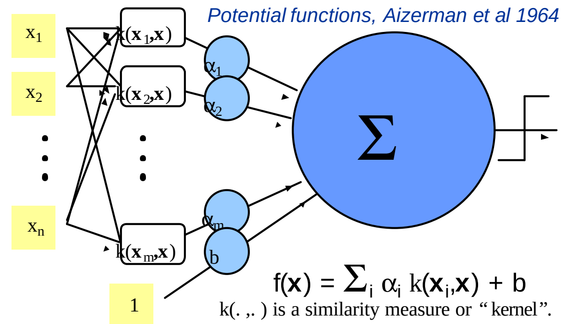
Dual forms

Aizerman-Braverman-Rozonoer -1964

- $f(\mathbf{x}) = \sum_i \alpha_i y_i k(\mathbf{x}_i, \mathbf{x})$
- $k(\mathbf{x}_i, \mathbf{x}) = \Phi(\mathbf{x}_i) \bullet \Phi(\mathbf{x})$

Kernel Method

Potential functions, Aizerman et al 1964



$f(\mathbf{x}) = \sum_i \alpha_i k(\mathbf{x}_i,\mathbf{x}) + b$

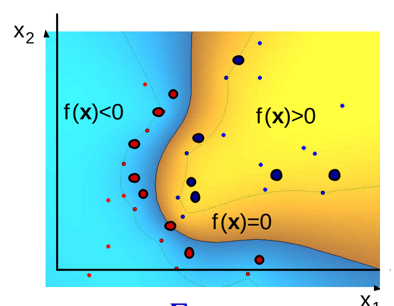
k(. ,.) is a similarity measure or "kernel".

Some Kernels (reminder)

A kernel is a dot product in *some* feature space: $k(\mathbf{s}, \mathbf{t}) = \Phi(\mathbf{s}) \bullet \Phi(\mathbf{t})$

- <u>Examples:</u>
- $k(\mathbf{s}, \mathbf{t}) = \mathbf{s} \bullet \mathbf{t}$ — Linear kernel
- $k(\mathbf{s}, \mathbf{t}) = \exp -\gamma\|\mathbf{s}-\mathbf{t}\|^2$ — Gaussian kernel
- $k(\mathbf{s}, \mathbf{t}) = \exp -\gamma\|\mathbf{s}-\mathbf{t}\|$ — Exponential kernel
- $k(\mathbf{s}, \mathbf{t}) = (1 + \mathbf{s} \bullet \mathbf{t})^q$ — Polynomial kernel
- $k(\mathbf{s}, \mathbf{t}) = (1 + \mathbf{s} \bullet \mathbf{t})^q \exp -\gamma\|\mathbf{s}-\mathbf{t}\|^2$ — Hybrid kernel

Support Vector Classifier

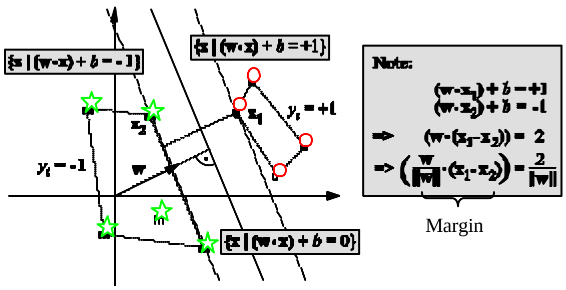


$\mathbf{x}=[x_1,x_2]$

$f(\mathbf{x}) = \sum_{k \in SV} \alpha_j y_j k(\mathbf{x}, \mathbf{x}_j)$

Boser-Guyon-Vapnik-1992

Margin and ||w||



Maximizing the margin is equivalent to minimizing $||w||$.

Quadratic Programming

- ***Hard margin:***

min $||\mathbf{w}||^2$
such that
$y_j(\mathbf{w}.\mathbf{x}_j+b)\geq 1$ for all examples.

- ***Soft margin:***

min $||\mathbf{w}||^2 + C\, \Sigma_j\, \xi_j^{\beta}$ $\quad\quad \xi_j \geq 0,\ \beta=1,2$
such that
$y_j(\mathbf{w}.\mathbf{x}_j+b)\geq(1-\xi_j)$ for all examples.

Dual Formulation

- Non-linear case: $\mathbf{x} \rightarrow \mathrm{F}(\mathbf{x})$
- min $||\mathbf{w}||^2 + C\, \Sigma_j\, \xi_j^{\beta}$ $\quad\quad \xi_j \geq 0$, $\beta=1$ or $\beta= 2$

such that
$y_j(\mathbf{w}.\mathrm{F}(\mathbf{x}_j)+b)\geq(1-\xi_j)$ for all examples.

- max $-½\, a^T \mathbf{K}\, a + a\, \mathbf{1}$ $\quad\quad \mathbf{K}= [\, y_i y_j\, k(x_i,x_j)] + (1/C)\, \delta_{ij}$

such that
$a^T\, \mathbf{y} = 0$; $0 \leq \alpha_i \leq C$

"Ridge SVC"

- ***Soft margin:***

min $||\mathbf{w}||^2 + C\, \Sigma_j\, \xi_j^{\beta}$ $\quad\quad \xi_j \geq 0,\ \beta=1,2$
such that
$y_j($~~$\mathbf{w}.\mathrm{F}(\mathbf{x}_j)+b)\geq(1-\xi_j)$ for all~~ examples.
$\quad f(\mathbf{x}_j)$

$1-y_j\, f(\mathbf{x}_j)<0,\ \xi_j=0,$ no penalty, not SV
$1-y_j\, f(\mathbf{x}_j)=0,\ \xi_j=0,$ no penalty, marginal SV
$1-y_j\, f(\mathbf{x}_j)=\xi_j>0,$ penalty ξ_j, non-marginal SV

- ***Ridge SVC:***

Loss $L(\mathbf{x}_j)=\max(0,\ 1-y_j\, f(\mathbf{x}_j))^{\beta}$
Risk $\Sigma_i\, L(\mathbf{x}_j)$
min $(1/C)\ ||\mathbf{w}||^2 + \Sigma_j\, L(\mathbf{x}_j)$ $\quad\quad$ *regularized risk*

Ridge Regression (reminder)

- Sum of squares:

  $R = \Sigma_i (f(\mathbf{x}_i) - y_i)^2 = \Sigma_i \underbrace{(1-y_i f(\mathbf{x}_i))}^2$

- Add "regularizer": Classification case

  $R = \Sigma_i (1-y_i f(\mathbf{x}_i))^2 + \lambda || \mathbf{w} ||^2$

- Compare with SVC:

  $R = \Sigma_i \max(0, 1-y_j f(\mathbf{x}_j))^\beta + \lambda || \mathbf{w} ||^2$

Structural Risk Minimization

- Nested subsets of models, increasing complexity/capacity:

  $S_1 \subset S_2 \subset \dots S_N$ *Vapnik-1984*

- Example, rank with $||\mathbf{w}||^2$

  $S_k = \{ w \mid ||\mathbf{w}||^2 < A_k \}$, $A_1 < A_2 < \dots < A_k$

- Minimization under constraint:

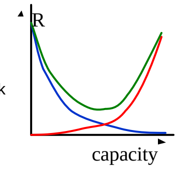
  min $R_{emp}[f]$ s.t. $||\mathbf{w}||^2 < A_k$

- Lagrangian:

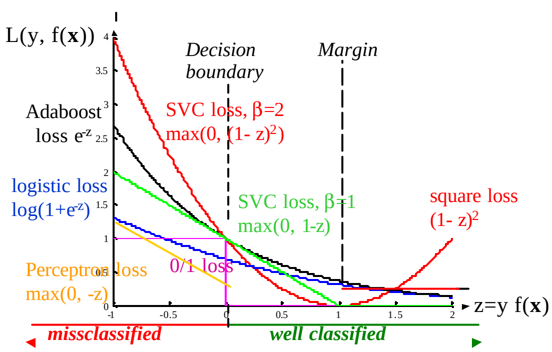
  $R_{reg}[f] = R_{emp}[f] + \lambda ||\mathbf{w}||^2$

- Radius-margin bound:

  $LOOcv = 4 r^2 ||\mathbf{w}||^2$ *Vapnik-Chapelle-2000*



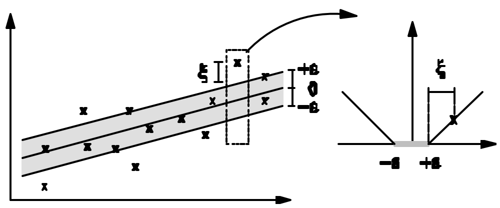
Loss Functions



Regularizers

- $|| \mathbf{w} ||_2^2 = \Sigma_i w_i^2$: 2-norm regularization (ridge regression, original SVM)
- $|| \mathbf{w} ||_1 = \Sigma_i |w_i|$: 1-norm regularization (Lasso Tibshirani 1996, 1-norm SVM 1965)
- $|| \mathbf{w} ||_1$ =length($\mathbf{w}$) : 0-norm (Weston et al., 2003)

Regression SVM



- Epsilon insensitive loss:
  $| y_i - f(x_i) |_\varepsilon$

Unsupervised learning

SVMs for:

- density estimation: Fit F(x) (Vapnik, 1998)
- finding the support of a distribution (one-class SVM) (Schoelkopf et al, 1999)
- novelty detection (Schoelkopf et al, 1999)
- clustering (Ben Hur, 2001)

Summary

- For statistical model inference, two ingredients needed:
  - **A loss function**: defines the residual error, i.e. what is not explained by the model; characterizes the data uncertainty or "noise".
  - **A regularizer**: defines our "prior knowledge", biases the solution; characterizes our uncertainty about the model. We usually bet on simpler solutions (Ockham's razor).

Exercise Class

Filters: see chapter 3

Method		X			Y			Comments
Name	Formula	B	M	C	B	M	C	
Bayesian accuracy	Eq. 3.1	+	s		+	s		Theoretically the golden standard, rescaled Bayesian relevance Eq. 3.2.
Balanced accuracy	Eq. 3.4	+	s		+	s		Average of sensitivity and specificity; used for unbalanced dataset, same as AUC for binary targets.
Bi-normal separation	Eq. 3.5	+	s		+	s		Used in information retrieval.
F-measure	Eq. 3.7	+	s		+	s		Harmonic of recall and precision, popular in information retrieval.
Odds ratio	Eq. 3.6	+	s		+	s		Popular in information retrieval.
Means separation	Eq. 3.10	+	i	+	+			Based on two class means, related to Fisher's criterion.
T-statistics	Eq. 3.11	+	i	+	+			Based also on the means separation.
Pearson correlation	Eq. 3.9	+	i	+	+	i	+	Linear correlation, significance test Eq. 3.12, or a permutation test.
Group correlation	Eq. 3.13	+	i	+	+	i	+	Pearson's coefficient for subset of features.
χ^2	Eq. 3.8	+	s		+	s		Results depend on the number of samples m.
Relief	Eq. 3.15	+	s	+	+	s	+	Family of methods, the formula is for a simplified version ReliefX, captures local correlations and feature interactions.
Separability Split Value	Eq. 3.41	+	s	+	+	s		Decision tree index.
Kolmogorov distance	Eq. 3.16	+	s	+	+	s	+	Difference between joint and product probabilities.
Bayesian measure	Eq. 3.16	+	s	+	+	s	+	Same as Vajda entropy Eq. 3.23 and Gini Eq. 3.39.
Kullback-Leibler divergence	Eq. 3.20	+	s	+	+	s	+	Equivalent to mutual information.
Jeffreys-Matusita distance	Eq. 3.22	+	s	+	+	s	+	Rarely used but worth trying.
Value Difference Metric	Eq. 3.22	+	s		+	s		Used for symbolic data in similarity-based methods, and symbolic feature-feature correlations.
Mutual Information	Eq. 3.29	+	s	+	+	s	+	Equivalent to information gain Eq. 3.30.
Information Gain Ratio	Eq. 3.32	+	s	+	+	s	+	Information gain divided by feature entropy, stable evaluation.
Symmetrical Uncertainty	Eq. 3.35	+	s	+	+	s	+	Low bias for multivalued features.
J-measure	Eq. 3.36	+	s	+	+	s	+	Measures information provided by a logical rule.
Weight of evidence	Eq. 3.37	+	s	+	+	s	+	So far rarely used.
MDL	Eq. 3.38	+	s		+	s		Low bias for multivalued features.

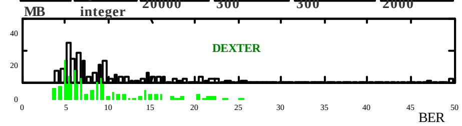
Homework 7

1) Download the software for homework 7.
2) Inspiring your self by the examples, write a new feature ranking filter object. Choose one in Chapter 3 or invent your own.
3) Provide the pvalue and FDR (using a tabulated distribution or the probe method).
4) Email a zip file your object and a plot of the FDR to guyoni@inf.ethz.ch with subject "Homework7" no later than:
   Tuesday December 13th.

Dexter

DEXTER filters **texts**

Size	Type	Features	Training	Validation	Test
0.9 MB	Sparse integer	20000	300	300	2000



Best entries:

BER~3.3-3.9%   AUC~0.97-0.99

Frac_feat~1.5%   Frac_probe~50%

Baseline Dexter

```
➢ my_classif=svc({'coef0=1', 'degree=1', 'gamma=0', 'shrinkage=0.5'});
➢ my_model=chain({s2n('f_max=300'), normalize, my_classif})
```

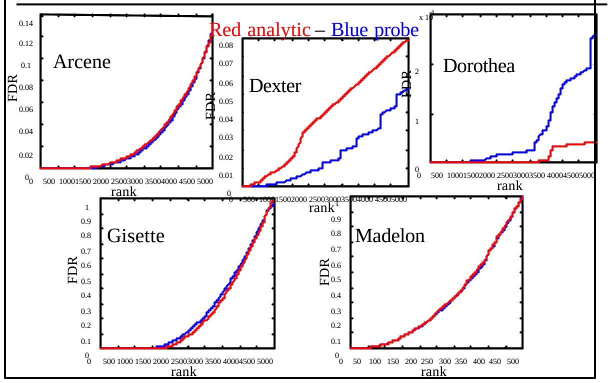
- Results:

% feat	% probe	Train BER%	Valid BER%	Test BER%	Train AUC	Valid AUC	Test AUC
1.5	16.33	0.33	7	5	1	0.982	0.988

Evalution of pval and FDR

- Ttest object:
  - computes pval analytically
  - FDR~pval*n_{sc}/n
- probe object:
  - takes any feature ranking object as an argument (e.g. s2n, relief, Ttest)
  - pval~n_{sp}/n_p
  - FDR~pval*n_{sc}/n

Analytic vs. probe



Relief