The Vapnik-Chervonenkis Dimension

and the

Learning Capability of Neural Nets

J.P. Marques de Sá – jmsa@fe.up.pt
INEB – Instituto de Engenharia Biomédica
FEUP - Faculdade de Engenharia da Universidade do Porto

Programme

1. Supervised Learning

1.1 Supervised Learning Model
1.2 Empirical Risk and ERM Principle

2. PAC Learning

2.1 Basic Issues of Learning
2.2 Definitions
2.3 PAC Concept
2.4 Examples

3. Sample Complexity in Finite Hypothesis Spaces

3.1 Version Space
3.2 Generalization of Training Hypotheses

4. Vapnik-Chervonenkis Dimension of MLPs

4.1 Linearly Separable Dichotomies
4.2 Hypothesis Space of MLPs
4.3 VC Dimension

5. Structural Risk and SRM Principle

5.1 Growth Function and ERM
5.2 Validity of Inductive Theories
5.3 Structural and Guaranteed Risks

6. Sample Complexity in Infinite Hypothesis Spaces

6.1 Bounds on PAC Learning
6.2 Study Case

Bibliography

Anthony M, Bartlett PL (1999) Neural Network Learning: Theoretical Foundations. Cambridge University Press.

Baum EB, Haussler D (1989) What Size Net Gives Valid Generalization? Neural Computation, 1:151-160.

Bishop CM (1995) Neural Networks for Pattern Recognition. Clarendon Press, Oxford.

Blumer A, Ehrenfeucht A, Haussler D, Warmuth MK (1989) Learnability and the Vapnik-Chernovenkis Dimension. J Ass Comp Machinery, 36:929-965.

Cherkassky V, Mulier F (1998) Learning from Data, John Wiley & Sons, Inc.

Ehrenfeucht A, Haussler D, Kearns M, Valiant L (1989) A General Lower Bound on the Number of Examples Needed for Learning. Information and Computation, 82:247-261.

Fukunaga K (1990) Introduction to Statistical Pattern Recognition. Academic Press, New York.

Haykin S (1999) Neural Networks. A Comprehensive Foundation. Prentice Hall Inc., New Jersey.

Kearns MJ, Vazirani UV (1997) An Introduction to Computational Learning Theory. The MIT Press.

Marques de Sá JP (2001) Pattern Recognition. Concepts, Methods and Applications. Springer Verlag[*]

Mirchandani G, Cao W (1989) On Hidden Neurons for Neural Nets. IEEE Tr Circ Syst, 36: 661-664.

Mitchell TM (1997) Machine Learning. McGraw Hill Book Co., New York.

Simon HU (1997) Bounds on the Number of Examples Needed for Learning Functions. SIAM J. of Computing, 26:751-763.

Vapnik VN (1998) Statistical Learning Theory. Wiley, New York.

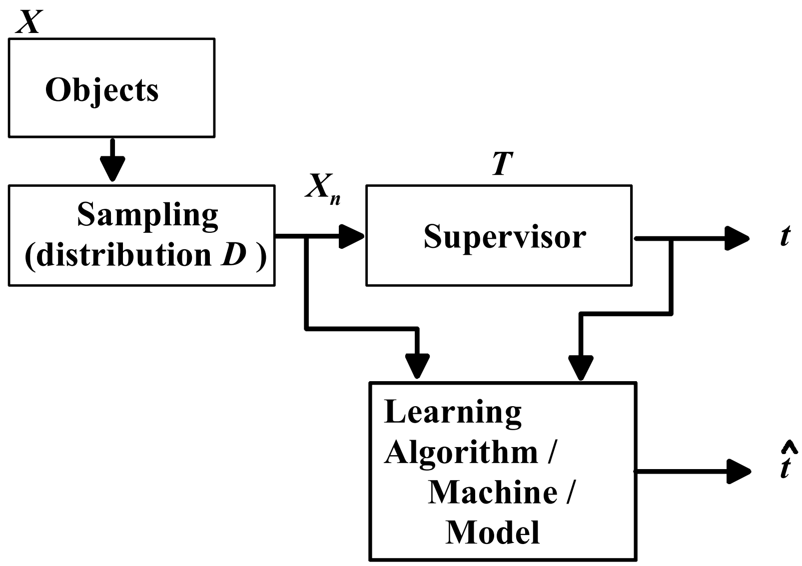
[*] Contains a CD which includes a program for the computation of multi layer perceptron VC dimension and sample complexity bounds.

Symbols

x	object (instance, example, case)
d	number of object features
n	number of objects
w	number of weights (MLP)
$\mathbf{x}$	vector (d-dimensional)
t	target value
$\hat{t}$	estimate of t
X	instance set
S	sample of n objects randomly drawn
P	discrete probability
p	pdf
Pe	error probability
$x \in X \sim D$	x drawn from X according to the distribution D

1 Supervised Learning

1.1 Supervised Learning Model



X	-	Object (instance) space
X_n	-	Sample with n objects
T	-	Target values domain (e.g. {0, 1})

Consider the *hypothesis*:

$$\begin{aligned} h: \quad X &\to T \\ x &\to \hat{t} = h(x) \end{aligned} ,$$

and the *hypothesis space* : $H = \{h : x \to X\}$.

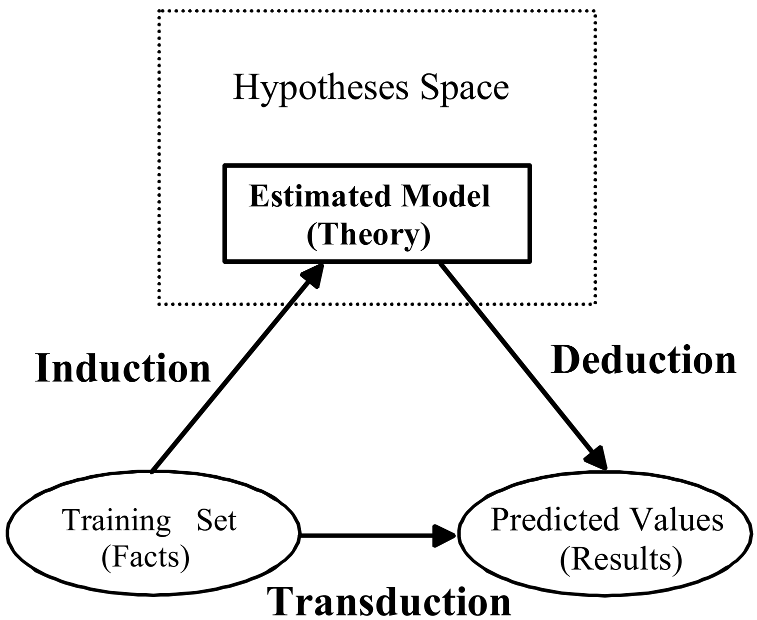
Often, x is a d-dimensional vector, $\mathbf{x}$:

$$\begin{aligned} h: \quad X \equiv \Re^d &\to T \\ \mathbf{x} &\to \hat{t} = h(\mathbf{x}) \end{aligned} .$$

Learning Objective:

Given the *sample* or *training set* $S = \{(\mathbf{x}, t(\mathbf{x}));\quad \mathbf{x} \in X_n\}$, find in H a hypothesis h that verifies:

$$h(\mathbf{x}) = t(\mathbf{x}), \quad \forall\ \mathbf{x} \in X$$



Supervised Learning = Inductive Learning

Example:

Given:

$$X = \Re^2, \qquad T = \{0, 1\},$$

$$S = \{\ (\mathbf{x}, t(\mathbf{x}));\quad \mathbf{x} \in X_n \subseteq X,\quad t(\mathbf{x}) \in T\ \};$$

$$H = \{h : X \to T;\ h(\mathbf{x}, \mathbf{w}) = \mathbf{w'}\,\mathbf{x} + w_0,\quad \mathbf{w} \in \Re^2\},$$

(parametric hypothesis space).

Determine $h \in H, \quad h(\mathbf{x}, \mathbf{w}) = t(\mathbf{x}), \quad \forall\ \mathbf{x} \in X$ (i.e., determine **w**, w_0).

How to determine $h(\mathbf{x}, \mathbf{w})$ **?**

1.2 Empirical Risk and ERM Principle

Hypothesis Risk

Let:

$\mathcal{A}=\{\alpha\}$: action/decision space (e.g. $\mathcal{A}=T$).

$\lambda(\alpha, h(\mathbf{x},\mathbf{w}))$: cost/risk of action/decision α when the machine receives $\mathbf{x}$ and has parameter $\mathbf{w}$.

Risk (individual) of x:

$$R(h(\mathbf{x},\mathbf{w}),\mathbf{x}) = \int_{\mathcal{A}} \lambda(\alpha, h(\mathbf{x},\mathbf{w}))\,p(\mathbf{x},\alpha)\,d\alpha$$

Risk of hypothesis *h*:

$$R(h) \equiv R(h(\mathbf{x},\mathbf{w})) = \int_{Xx\mathcal{A}} \lambda(\alpha, h(\mathbf{x},\mathbf{w}))\,p(\mathbf{x},\alpha)\,d\mathbf{x}\,d\alpha$$

Objective: find **w** that minimizes $R(h)$

1 - Classification Case

$\lambda(\alpha, h(\mathbf{x},\mathbf{w})) = \lambda(\omega, h(\mathbf{x},\mathbf{w}))$, with:

- $\omega \in \Omega=\{ \omega_i \,;\, i=1,..., c\}$, set of c classes.
- $T = \{ t_i = t(\omega_i);\, i=1,..., c\}$

$$R(h) = \sum_{i=1}^{c} \int_X \lambda(\omega_i, h(\mathbf{x},\mathbf{w}))\, p(\mathbf{x},\omega_i)\, d\mathbf{x}$$

Special case:

$$\lambda(\omega, h(\mathbf{x},\mathbf{w})) = \begin{cases} 0 & \text{if} \quad t(\omega) = h(\mathbf{x},\mathbf{w}) \\ 1 & \text{if} \quad t(\omega) \neq h(\mathbf{x},\mathbf{w}) \end{cases}$$

Thus:

$$R(h) = \sum_{i=1}^{c} \int_{\substack{\cup X_j \\ j \neq i}} P(\omega_j \mid \mathbf{x})\, p(\mathbf{x})\, d\mathbf{x} = \sum_{i=1}^{c} Pe(\omega_i) = Pe$$

Let D be the distribution of $\mathbf{x}$ in X:

$$R(h) = \underset{\mathbf{x} \in X \sim D}{Pe(h)} \qquad \textit{true error of } h.$$

Bayes' minimum risk rule:

$$R(\omega_i \mid \mathbf{x}) = \sum_{j \neq i} P(\omega_j \mid \mathbf{x}) = 1 - P(\omega_i \mid \mathbf{x})$$

Minimize $R(\omega_i \mid \mathbf{w}) \rightarrow$ Max. Prob. *a Posteriori*

2 - Regression Case

$\lambda(\alpha, h(\mathbf{x},\mathbf{w})) = \lambda(y, h(\mathbf{x},\mathbf{w}))$, $\qquad y = g(\mathbf{x}) + \varepsilon$.

Special case:

$$\lambda(y, h(\mathbf{x},\mathbf{w})) = \left(y - h(\mathbf{x},\mathbf{w})\right)^2$$

$$R(h) = \int_{XxT} (y - h(\mathbf{x},\mathbf{w}))^2 \, p(\mathbf{x}, y) d\mathbf{x} dy$$

Minimize $R(h)$ → LMS

Empirical Risk Minimization (ERM) Principle:

Given a training set S, with n instances, determine the function $h(\mathbf{x},\mathbf{w})$ that minimizes:

$$R(h,n) = \int_S \lambda(\alpha, h(\mathbf{x},\mathbf{w}))\, p(\mathbf{x},\alpha)\, d\mathbf{x}\, d\alpha$$

(i.e., in the sample/training set S)

Minimum (optimal) empirical risk:

$$R_{\text{emp}}(h(\mathbf{w}^*), n) = \min_{\mathbf{w}} R(h(\mathbf{w}), n)$$

(in the sample/training set S)

Classic Theory of Statistical Classification

Fundamental assumption:

The distribution of the instances in any sample S is known and stationary.

Classic situation:

- The distributions of the instances are Gaussian.
- The *a posteriori* probabilities, computed according the Bayes Law, also determine the model/hypothesis (linear, quadratic, etc.).
- The ERM hypothesis is obtained through the estimation of the distribution parameters.

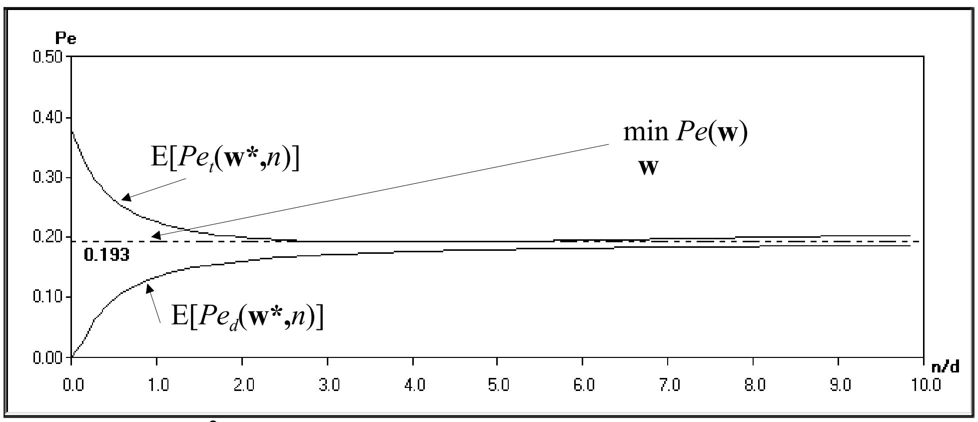
Example:

Classification with:

- $X = \Re^7$; $T = \{0, 1\}$ (two classes)
- Gaussian distributions of **x** with equal covariance, C

$$H = \left\{ h : X \to T;\ h(\mathbf{x}, \mathbf{w}) = \mathbf{w}'\,\mathbf{x} + w_0,\quad \mathbf{w} \in \Re^2 \right\} \to \textit{linear model}$$

- **w**, w_0 determined by S.



δ^2=3 (square of the Battacharyya distance).

There are exact formulas to compute:

$\mathrm{E}\left[Pe_t(h, n)\right]$: Average test error.

$$Pe_t(h, \infty) = Pe(h)$$

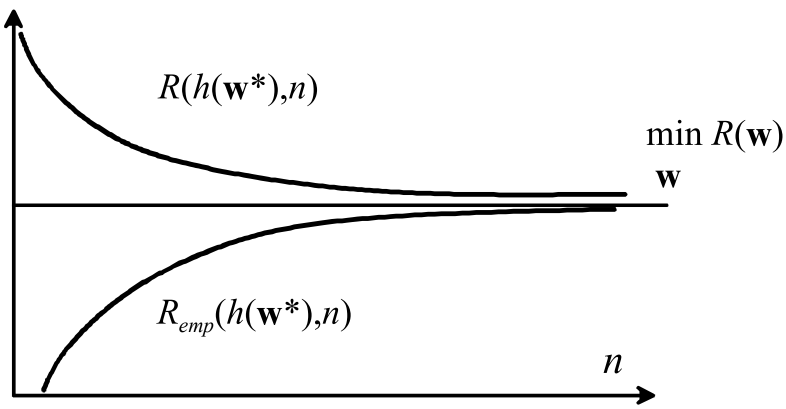
$\mathrm{E}\left[Pe_d(h, n)\right]$: Average training error (*average empirical risk*).

$$Pe_d(h, n) = Pe_{emp}(h)$$

$\min_{\mathbf{w}} Pe(h(\mathbf{w}))$: Optimal Bayes error.

General Situation

- The distributions of the instances are arbitrary.
- The model is unknown and has to be estimated.



$R_{emp}(h,n)$:	Optimal empirical risk, obtained by ERM ($Pe_d(h,n) = Pe_{emp}(h)$ for classification)
$R(h,n)$:	True risk of the ERM hypothesis ($Pe(h)$ for classification)
$\min_{\mathbf{w}} R(h(\mathbf{w}))$:	Optimal risk

The ERM principle is said to be *consistent* if:

$$R(h,n) \underset{n\to\infty}{\to} \min_{\mathbf{w}} R(\mathbf{w})$$

and

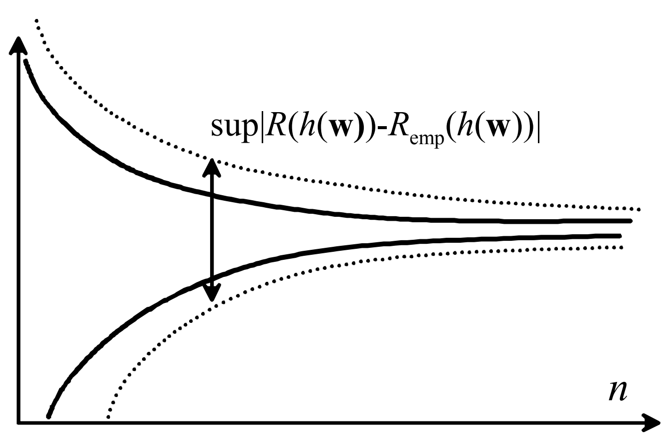
$$R_{emp}(h,n) \underset{n\to\infty}{\to} \min_{\mathbf{w}} R(\mathbf{w})$$

Fundamental Theorem of the Statistical Learning Theory (Vapnik, Chervonenkis, 1989):

For bounded cost functions the ERM principle is consistent iff:

$$\lim_{n\to\infty} P\left[\sup_{\mathbf{w}}\left|R(h(\mathbf{w})) - R_{emp}(h(\mathbf{w}))\right| > \varepsilon\right] = 0, \qquad \forall \varepsilon > 0$$

(i.e., consistency must be assesses in a “worst case” situation)



The theorem does not tell us:

- Whether or not there is convergence in probability of a given hypothesis.
- Assuming that such convergence exists, what is the minimum n required for the empirical error to be below a given value.

2 PAC Learning

2.1 Central Issues of Learning

Sample complexity:

What is the n = card(X_n) needed for the learning algorithm to converge (with high probability) to effective learning?

Computational complexity:

What computational effort is required for the learning algorithm to converge (with high probability) to effective learning?

Algorithm performance:

How many objects will be misclassified (error) until the algorithm converges to effective learning?

2.2 Definitions

X **- Instance domain.**

$$X = \text{Set of persons}$$

C **- Concept space,** $C \subseteq 2^X$ **(set of dichotomies of** ***X*****)**

$$C = \{\text{Caucasian, Portuguese, obese, ...}\}$$

$$c = \text{obese}$$

t_c **- Target function, concept indicator**

$$t_{\text{obese}} \in T: \quad X \quad \rightarrow \quad \{0, 1\}$$

$$t_{\text{obese}}(\text{John}) = 1$$

Frequently, we take $c \equiv t_c$: obese(John) = 1

D **- Sample distribution, stationary**

$$D = \text{distribution of persons in a supermarket}$$

Note: When the sample distribution of the objects obeys a known model, one is able, in principle, to determine an exact answer to the preceding questions (parametric statistical classification).

L - Set of learning algorithms

$$L = \{l : S \rightarrow H\}$$

The *learner* $l \in L$ considers:

- A training set *S*, generated according to *D*.
- A set of possible hypothesis *H* having in view to learn the concept.

Example:

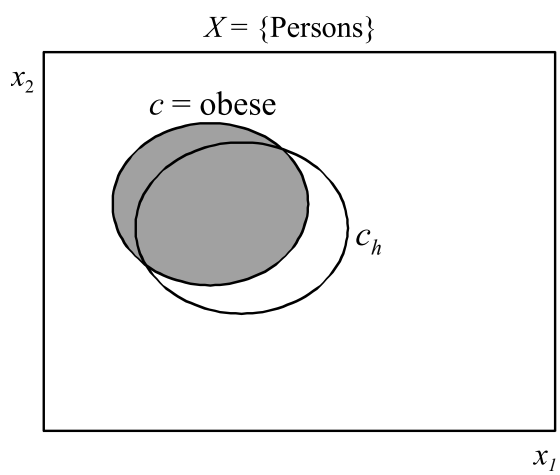
$$H = \{h : X \rightarrow \{0,1\} : \\ h(x) = w_2 x_2 + w_1 x_1 + w_0; \quad x_1 \equiv \text{height}(x), x_2 \equiv \text{width}(x), w_0, w_1, w_2 \in \Re\}$$

c_h - Set induced by *h* in *X*

$$c_h = \{x \in X; \quad h(x) = 1\}$$

Example:

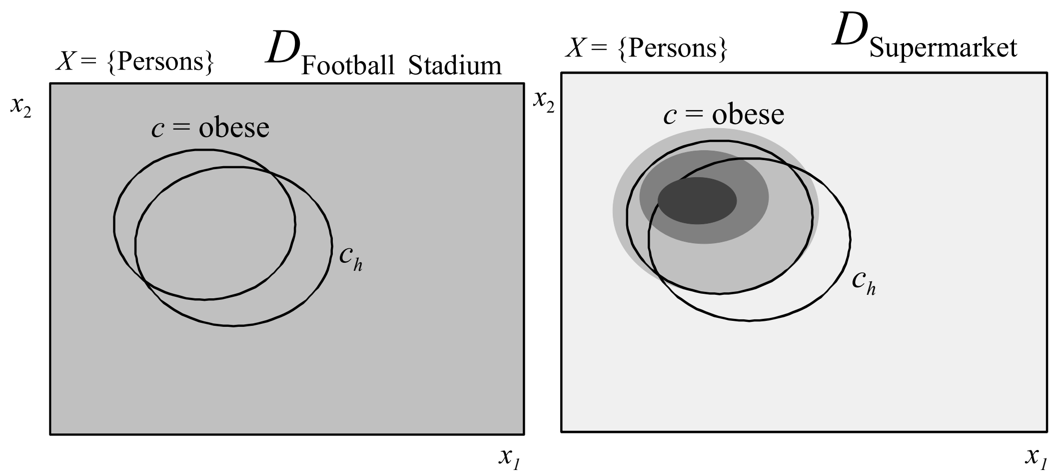
$$c_h = \{x \in X; \quad \mathbf{w}'\mathbf{x} + w_0 = 1\}; \ \mathbf{w}' = [w_1 \quad w_2], \ \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$



Pe **- Error (true error) of hypothesis *h***

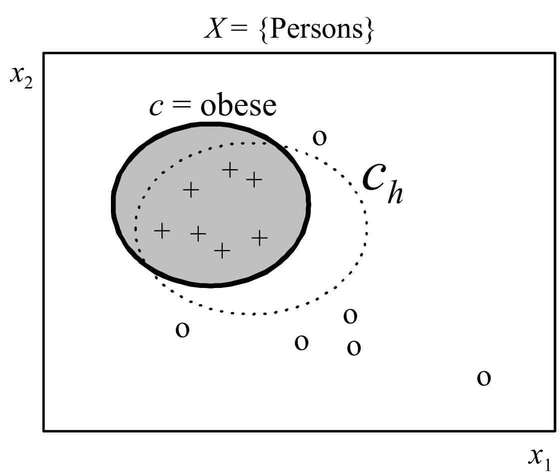
$$Pe(h) \equiv Pe_D(h) = \underset{x \in X \sim D}{P}\left(c(x) \neq h(x)\right)$$

The error depends on the distribution D:



Consistent hypothesis

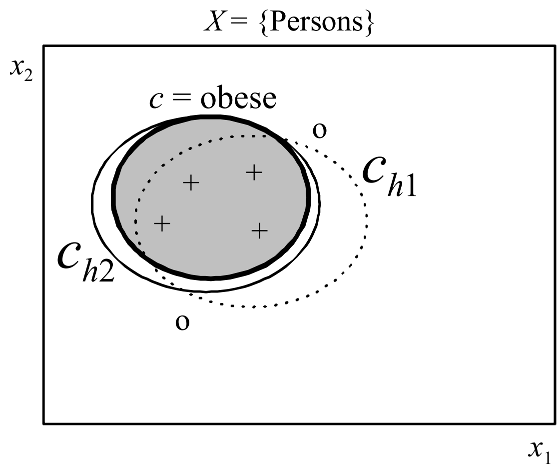
h is consistent iff $\forall x \in X_n, \quad c(x) = h(x)$, i.e., $Pe_{emp}(h)$=0



2.3 PAC Concept

Given $l \in L$, generating hypothesis *h*, is it realistic to expect *Pe*(*h*)=0?

In general ($X_n \neq X$), there may exist several *h*s consistent with the training set and we do not know which one learns the concept.



h_1 e h_2 are both consistent; however, h_2 learns the concept better (smaller true error).

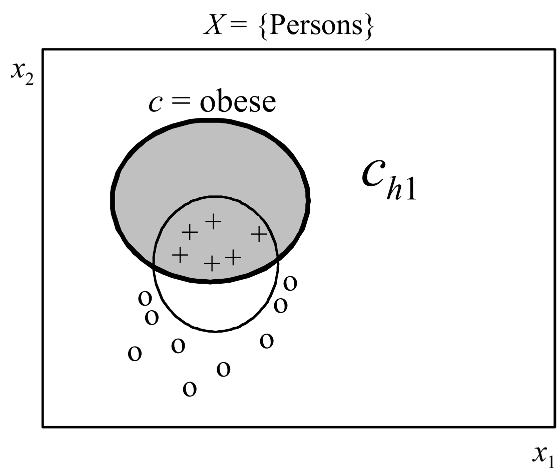
One can only hope that:

$$Pe_D(h) \leq \varepsilon ,$$

ε : *error parameter*.

The learner is approximately correct...

As the training set is randomly drawn there is always a non-null probability that the drawn sample contains misleading instances.



Thus, we can only expect that:

$$P(Pe_D(h) \le \varepsilon) \ge 1 - \delta$$

δ: *confidence parameter*.

The learner is probably approximately correct...

Definition of PAC learning - *Probably Approximately Correct*:

Let C represent a set (class) of concepts defined in X and l a learner using $X_n \subseteq X$ and a hypothesis space H.

C is *PAC-learnable* by l (l is a PAC learning algorithm for C), if:

$$\forall\, c \in C, \quad \forall\, D\,(\text{in } X), \quad \forall\, \varepsilon, \delta, \quad 0 < \varepsilon, \delta < 0.5,$$

$$l \in L \quad \text{determines} \quad h \in H, \quad P(Pe(h) \le \varepsilon) \ge 1 - \delta\,,$$

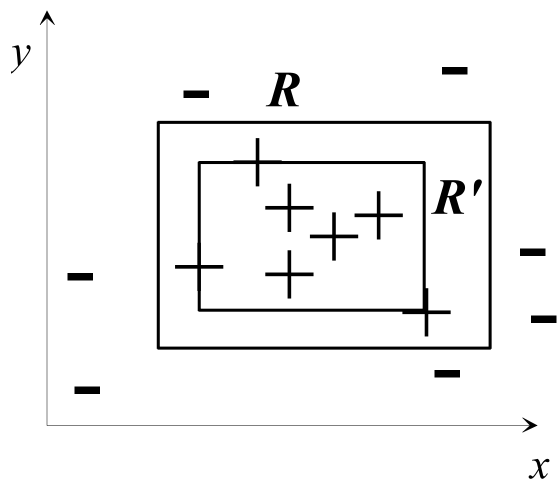
in polynomial time in $1/\varepsilon$, $1/\delta$, n and size(c).

size(c) - Number of independent elements used in the representation of the concepts.

Representation	**size(c)**
Boolean canonical conjunctive expression	Nr of Boolean literals
Decision tree	Nr of tree nodes
Multi-layer perceptron (MLP)	Nr of MLP weights

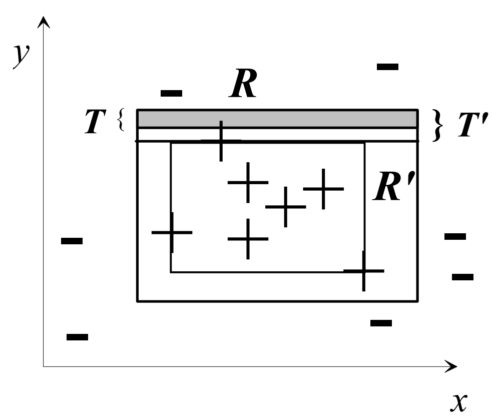
2.4 Examples

1 - The concept class corresponding to rectangles aligned with the axes in $\Re^2$, is PAC-learnable (see e.g. Kearns, Vazirani, 1997).



- R : concept to be learned
- The learner l generates the hypothesis R' : rectangle that perfectly fits the positive examples.

Thus: $R' \subset R$ and R'-R is the reunion of 4 rectangular strips (e.g. T')



Given ε let T be the strip (for a given D) corresponding to: $P(\mathbf{x} \in T) = \dfrac{\varepsilon}{4}$.

What is the probability that: $P(\mathbf{x} \in T') > \dfrac{\varepsilon}{4}$?

$P(\mathbf{x} \in T') > \dfrac{\varepsilon}{4} \quad \Rightarrow \quad T' \supset T \quad \Rightarrow \quad T$ does not contain any point of X_n.

Probability that T does not contain any point of X_n:

$$\left(1 - \frac{\varepsilon}{4}\right)^n$$

Hence:

$$P\left(P\left(x \in T'\right) > \frac{\varepsilon}{4} \right) = \left(1 - \frac{\varepsilon}{4}\right)^n \quad \Rightarrow$$

$$P\left(\left(R' - R\right) > \varepsilon\right) \le \delta \qquad \text{with} \qquad \frac{\delta}{4} = \left(1 - \frac{\varepsilon}{4}\right)^n \quad \Rightarrow$$

$$P(Pe(h) \le \varepsilon) \ge 1 - \delta$$

Therefore, given ε and δ the concept is PAC-learnable for n such that:

$$\left(1 - \frac{\varepsilon}{4}\right)^n \le \frac{\delta}{4}$$

$$\left(1 - x\right) \le e^{-x} \quad \Rightarrow \quad 4e^{-n\varepsilon/4} \le \delta \quad \Rightarrow \quad n \ge \left(\frac{4}{\varepsilon}\right) \ln\left(\frac{4}{\delta}\right)$$

n is polynomial in $1/\varepsilon$ and $1/\delta$. For instance, for $\varepsilon = \delta = 0.05$: $n > 351$

2 - Let:

$$X = \{(a_1, a_2, \ldots, a_d);\quad a_i \in \{0, 1\}\} \equiv \{0, 1\}^d$$

Each a_i represents the value of a Boolean variable:

$$a_i = 0 \quad \to \quad \bar{x}_i$$
$$a_i = 1 \quad \to \quad x_i$$

Let C be the class of Boolean conjunctions, e.g.:

$$x_1.\bar{x}_3.x_4$$

$\forall c \in C, \quad \text{size}(c) \le 2d$.

The class of Boolean conjunctions is PAC-learnable (see e.g. Kearns e Vazirani, 1997).

Algorithm: Remove from $x_1\bar{x}_1x_2\bar{x}_2 \ldots x_d\bar{x}_d$ any literal not matching a true value of the respective variable in an instance **x** with $t(\mathbf{x})=1$.

Example:

$$S = \{\ ((0,0,1),1),\quad ((0,1,0),0),\quad ((0,1,1),1)\ \}$$

$$x_1\bar{x}_1x_2\bar{x}_2x_3\bar{x}_3 \quad \to \quad \bar{x}_1\bar{x}_2x_3 \quad \to \quad \bar{x}_1x_3$$

It can be shown that: $$n \ge \frac{2d}{\varepsilon}(\ln(2d) + \ln(\frac{1}{\delta}))$$

3 - Let $X = \{0, 1\}^d$ and C be the class of Boolean disjunctive forms with three terms:

u + v + w;

each term is a conjunctive form with at most $2d$ literals.

$$\forall c \in C, \quad \text{size}(c) \le 6d$$

The following can be shown (see discussion e.g. in Kearns and Vazirani, 1997):

- Learning this concept class is equivalent to solving the problem of colouring graph nodes using 3 colours, in such a way that all edges have different node colours. This is supposedly a NP problem, implying a non-PAC learning of the former problem.

- If a conjunctive representation of the problem is accepted then it becomes PAC !

3 Sample Complexity in Finite Hypothesis Spaces

Is it possible to obtain a lower bound for the sample complexity, valid in any situation?

equivalently

How many objects must a training set at least have so that, with high probability, one can determine an effective hypothesis?

3.1 Version Space

<u>**Definitions:**</u>

Version space:

$$VS_{H,S} = \{\, h \in H; \quad \forall\, (x, c(x)) \in S, \quad h(x) = c(x) \,\}$$

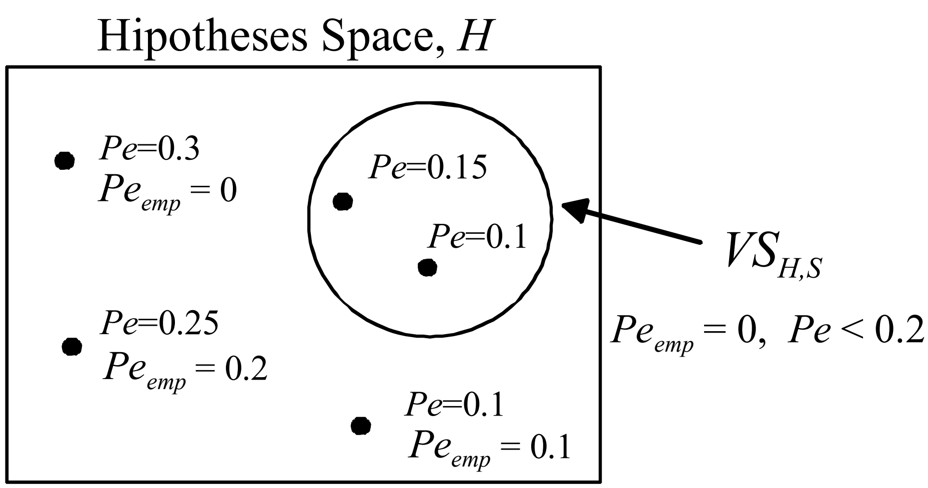
Set of consistent hypotheses,
with *training error* $Pe_{emp}(h) = 0$.

ε-Exhausted version space:

Let c be a concept. The version space is ε-exhausted with respect to c and D if any hypothesis of $VS_{H,S}$ has an error below ε.

$$\forall\ h \in VS_{H,S}, \quad Pe(h) < \varepsilon$$

Example of a 0.2-exhausted version space:



3.2 Generalization of Training Hypotheses

Theorem:

For a finite H with $|H|$ distinct hypotheses and a sample S with $n \geq 1$ objects, randomly drawn from a target concept c, then, for $0 \leq \varepsilon \leq 1$, the probability of the version space $VS_{H,S}$ not being ε-exhausted (with respect to c) is less or equal than:

$$\left|H\right| e^{-\varepsilon n}$$

Informal notion:
The probability of finding a good training hypothesis (consistent with the training set) but, as a matter of fact, a bad hypothesis (with true error greater than ε) is smaller than $\left|H\right| e^{-\varepsilon n}$, where n is the number of training objects.

Demonstration:

1. Let $h_1, h_2, ..., h_k$ be all the hypotheses with $Pe \geq \varepsilon$.

2. $VS_{H,S}$ is not ε-exhausted if $\exists\, h_i \in VS_{H,S} \quad i = 1, \ldots, k$

3. $Pe(h_i) \geq \varepsilon \quad \Rightarrow \quad P(h_i(x) = c(x)) = 1 - \varepsilon, \quad \forall x \in X_n$

4. $P\left(h_i \text{ consistent}\right) = P(h_i(x_1) = c(x_1) \wedge \ldots \wedge h_i(x_n) = c(x_n)) = (1-\varepsilon)^n$

5. $P(h_1 \text{ consistent} \vee \ldots \vee h_k \text{ consistent}) =$
$k(1-\varepsilon)^n \leq |H|(1-\varepsilon)^n \leq |H|\, e^{-\varepsilon n}$

The number of needed training examples in order to attain a probability below a given value, δ, is:

$$\left|H\right| e^{-\varepsilon n} \leq \delta \qquad \Rightarrow \qquad n \geq \frac{1}{\varepsilon}(\ln\left|H\right| + \ln(\frac{1}{\delta}))$$

<u>Notes:</u>

1. Note the similarity between the obtained expression with the previous ones

2. Note that the n bound can be quite pessimistic. As a matter of fact the Theorem states a probability growing with $|H|$ (it can be bigger than 1!)

3. Note that the Theorem does not apply to infinite $|H|$. For this situation one needs another complexity measure of H.

4 Vapnik-Chervonenkis Dimension of MLPs

We measure the complexity of H not by the number of distinct hypotheses but, instead, by the number of distinct instances that can be discriminated by H.

4.1 Linearly Separable Dichotomies

Definition:

A set of points is *regularly distributed* in $\Re^d$ if no subset of (d+1) points is contained in a hyperplane of $\Re^d$.

Theorem (Cover, 1965):

The number of linearly separable dichotomies (i.e. by a linear discriminant) of n points regularly distributed in $\Re^d$, is:

$$D(n,d) = \begin{cases} 2\sum_{i=0}^{d} C(n-1,i) & , \quad n > d+1; \\ 2^n & , \quad n \le d+1. \end{cases}$$

Case d=2:

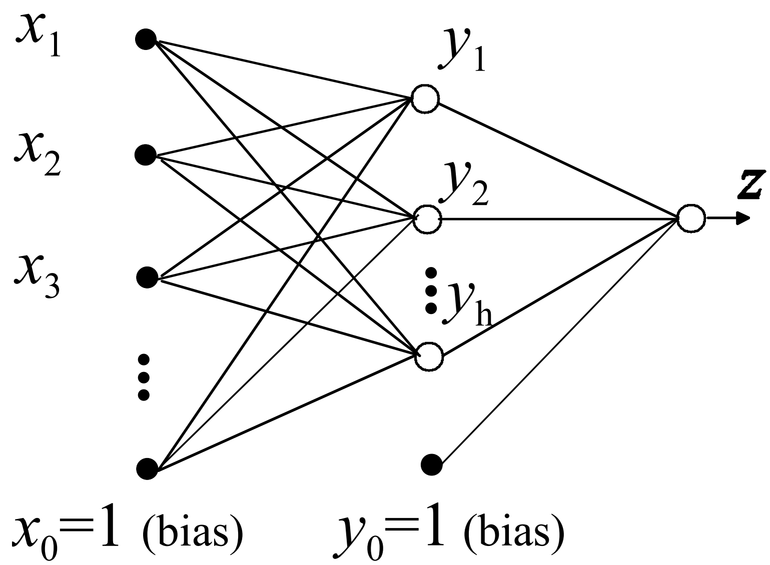
For n=3, all 2^3=8 dichotomies are linearly separable;
For n=4, only 14 out of 16 dichotomies are linearly separable;
For n=5, only 22 out of 32 dichotomies are linearly separable.

Number of points	2	3	4	5	6	7	8
Dichotomies	4	8	16	32	64	128	256
Linearly separable dichotomies	4	8	14	22	32	44	58

4.2 Hypotheses Space of MLPs

Let H be the hypotheses space of a MLP with:

- Two layers
- A hidden layer with m neurons
- One output
- Neuronal activation function: threshold function.



Model complexity:

Number of neurons (processing units):

$$u = m + 1$$

Number of weights (model parameters):

$$w = (d+1)m + m + 1$$

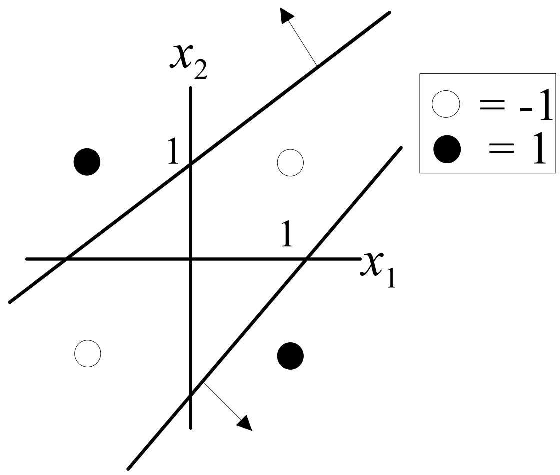
Model representation capability:

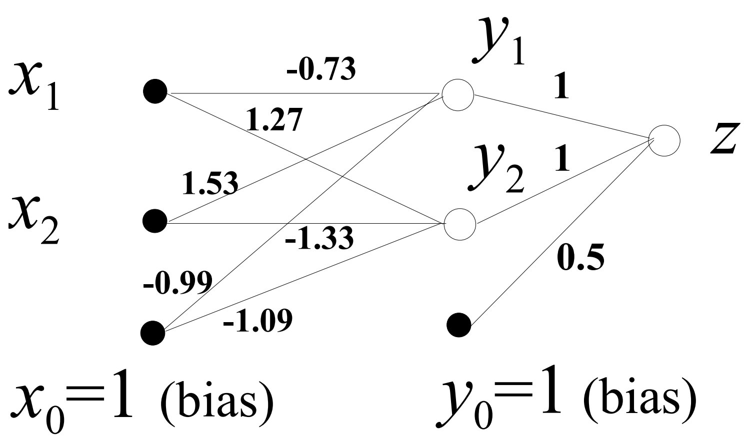
Each neuron of the first layer implements a linear discriminant, dividing the space into half-spaces:

$$y_j = f(\mathbf{w}'\mathbf{x} + w_0), \quad f\text{, threshold function (e.g. in } \{-1, 1\})$$

The output layer implements logical combinations of the half-spaces.

XOR example:





x_1	x_2	y_1	y_2	z = y_1 OR y_2
1	1	-1	-1	-1
1	-1	-1	1	1
-1	1	1	-1	1
-1	-1	-1	-1	-1

Theorem (Mirchandani and Cao, 1989):

The maximum number of regions linearly separable in $\Re^d$, by a MLP (satisfying the mentioned conditions) with *m* hidden neurons, is:

$$R(m,d)=\sum_{i=0}^{\min(m,d)} C(m,i). \qquad (1)$$

Note that: $R(m, d) = 2^m$ for $m \le d$.

Corolary:

Lower bound for the number of training set objects:

$$n \ge R(m, d)$$

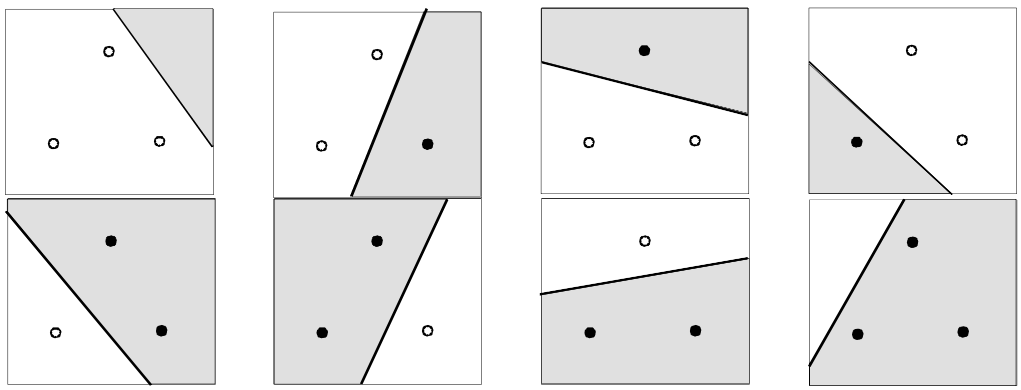
Case *d*=2:

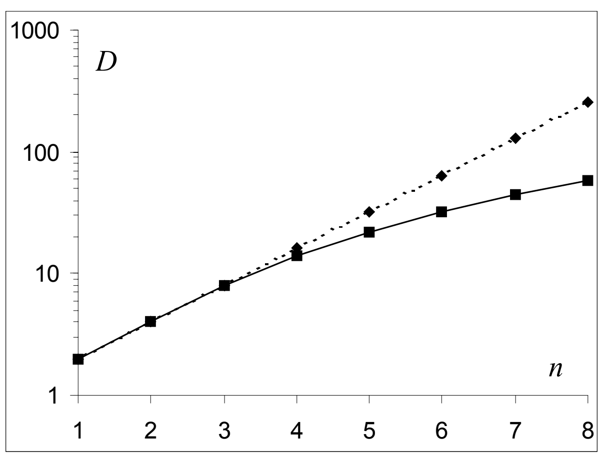
Number of linearly separable regions:

m	1	2	3	4	5	6	7	8
$R(m, 2)$	2	4	7	11	16	22	29	37

Case *d*=2, *m*=1:

- $R(1, 2) = 2$ linearly separable regions, by one linear discriminant.
- Maximum number of points allowing all possible dichotomies with one linear discriminant: $n = 3$

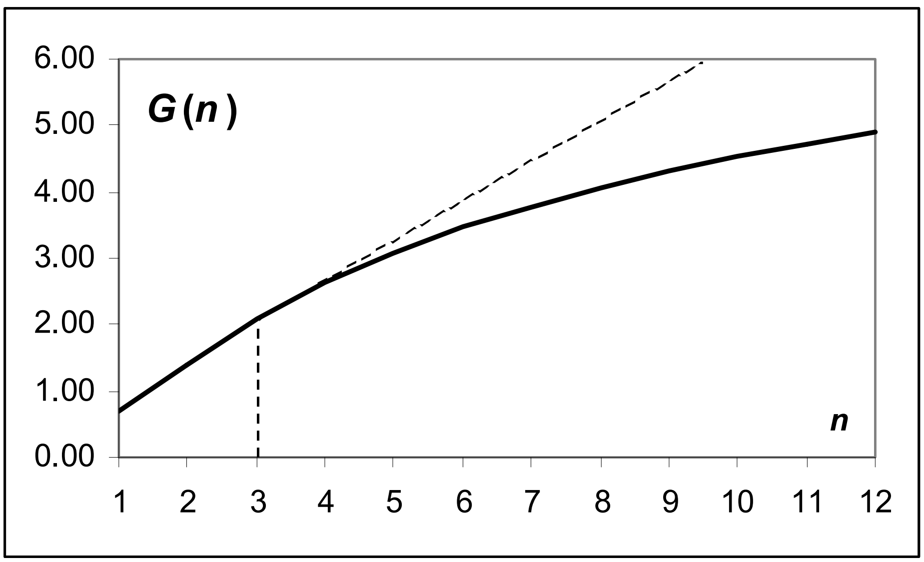




$D(n,2)$: Number of linearly separable dichotomies by a MLP2:1.

- Up to n=3 all 2^n linearly separable dichotomies are obtainable. It is only beyond this value that the MLP is able to generalize.
- n=3 measures the sample complexity of a MLP with m=1.
- $N(n)$: Number of linearly separable dichotomies, implementable by a MLP in n points ($D(n,2)$ for m=1).
- The MLP *growth function* is defined as:

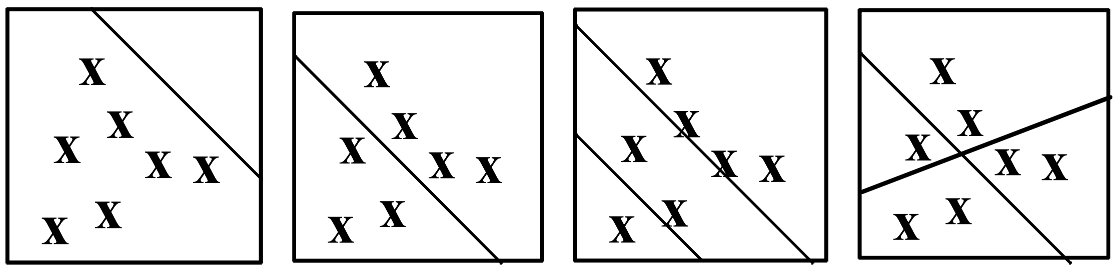
$$G(n) = \ln N(n)$$



The $G(n)$ evolution is always as illustrated.

Case *m*=2, *d*=2 :

There is a maximum of $R(2,2)=4$ linearly separable regions (with two discriminants)

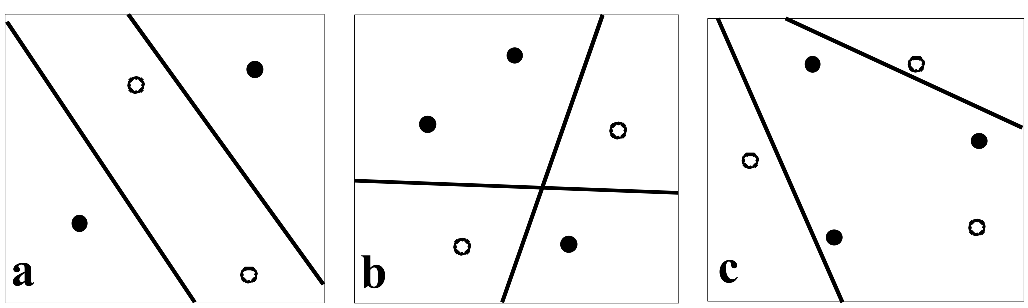


Linearly separable dichotomies that can be obtained:

$n = 4$: all.

$n = 5$: lying in a convex hull: all.

n = 6 : lying in a convex hull: some dichotomies are not obtainable.



Definition:

A set with n points is *shattered* by the MLP if $N(n) = 2^n$.

4.3 Dimensão de Vapnik-Chervonenkis

Definition:

The Vapnik-Chervonenkis dimension, d_{VC}, of an MLP is the cardinality of the largest regularly distributed set of points that can be shattered by the MLP.

Informal notion:

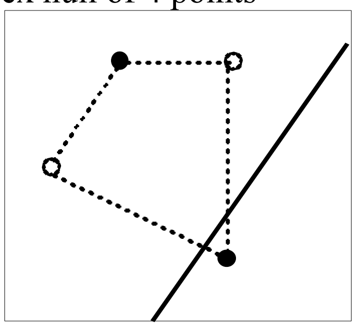
Largest number of training set examples that can be learned without error for all possible {0, 1} labellings.

$n \leq d_{VC}$: consistent learning without generalization.

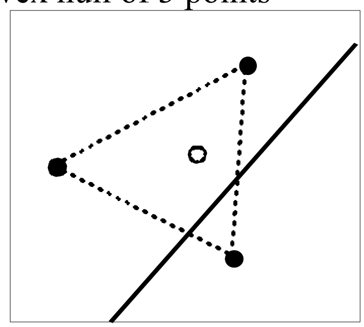
CASE *d*=2, *m*=1:

Is there a 4 points set that can be shattered? No. Hence $d_{VC} = 3$.

Convex hull of 4 points



Convex hull of 3 points



Calculation of d_{VC}:

Lower bound is easy to find:

d_{VC} (MLP) $\geq k$: Find one set of k points that can be shattered by the MLP.

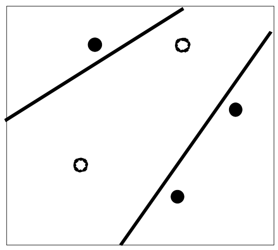
$$d_{VC}\text{ (MLP)} \geq R(m, d)$$

Upper bound is difficult to find:

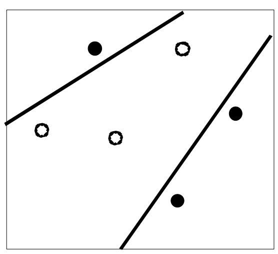
d_{VC} (MLP) $\leq k$: Prove that no set of k+1 points can be shattered by the MLP.

CASO *d*=2, *m*=2:

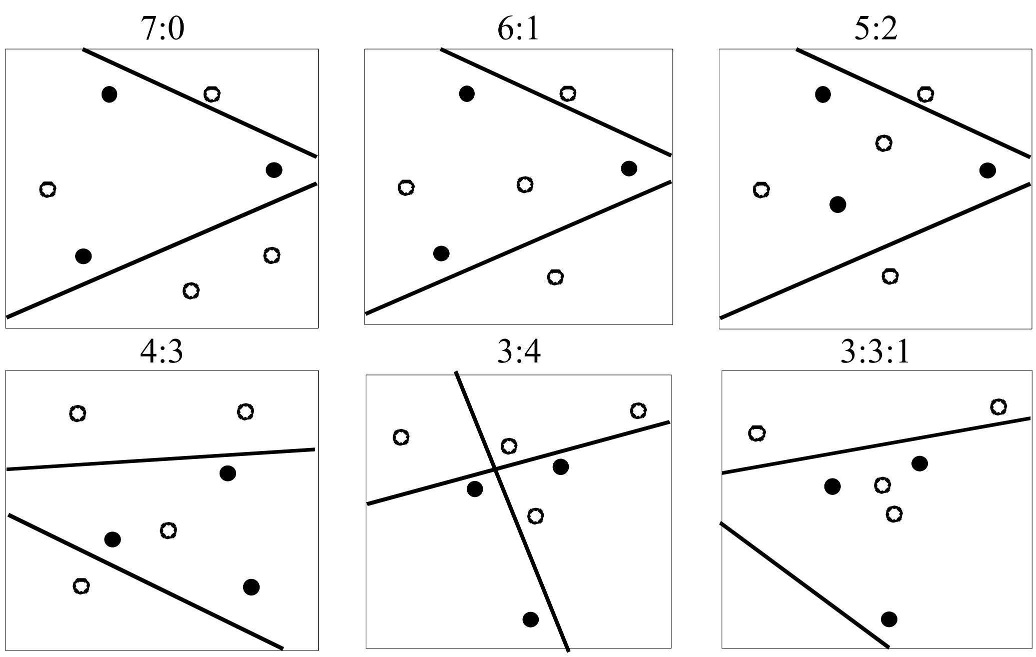
n=5 ? Yes.



n=6 ? Yes, for a convex hull of 5 points.

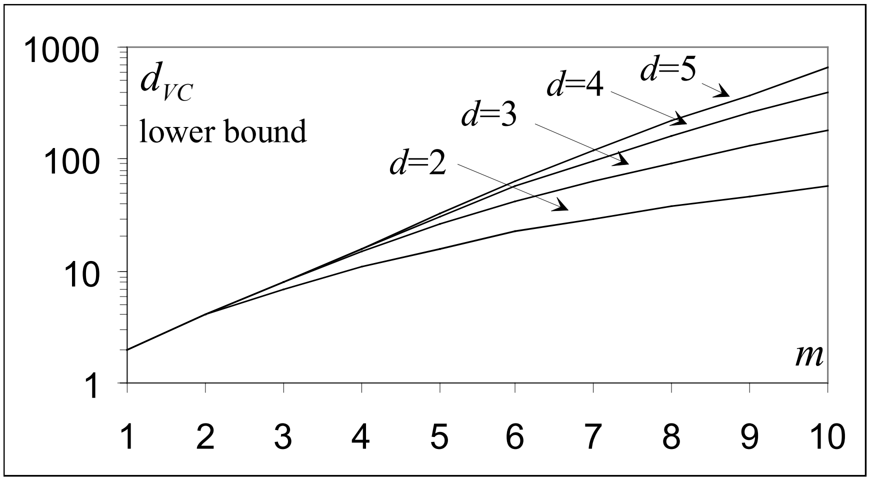


n=7 ? No. Hence, $d_{VC} = 6$.



<u>Lower bound of d_{VC}:</u>

$$d_{VC}\,(\text{MLP}) = R(m, d)$$



<u>Upper bound of d_{VC},</u> For an MLP with u neurons and w weights (Baum and Haussler, 1989):

$$d_{VC} \leq 2w\log_2(eu) \qquad (2).$$

Case d=2: Case d=5:

m	1	2	...	10
lower bound = R(m,d)	2	4		638
$\boldsymbol{d_{VC}}$	3	6		?
upper bound	9	54		696

5 Structural Risk and VC Dimension

5.1 Growth function and ERM

The ERM principle is consistent iff:

$$\lim_{n\to\infty} P\left[\sup_{\mathbf{w}}\left|R(h(\mathbf{w})) - R_{emp}(h(\mathbf{w}))\right| > \varepsilon\right] = 0, \qquad \forall\varepsilon > 0$$

The convergence is called *fast* if:

$$\forall\varepsilon > 0 \quad \exists n > n_0, b, c > 0 \quad P\left[\sup_{\mathbf{w}}\left|R(h(\mathbf{w})) - R_{emp}(h(\mathbf{w}))\right| > \varepsilon\right] < be^{-cn\varepsilon^2}$$

<u>The following can be proved:</u>

- **The ERM principle is consistent and of fast learning iff:**

$$\lim_{n\to\infty}\frac{G(n)}{n} = 0$$

- ***G*(*n*) is either linear in *n* or, beyond a certain value of *n*, is bounded by**

$$G(n) \le d_{VC}(1 + \ln\frac{n}{d_{VC}})$$

- **Thus, if d_{VC} is finite the ERM principle is consistent and of fast learning**.

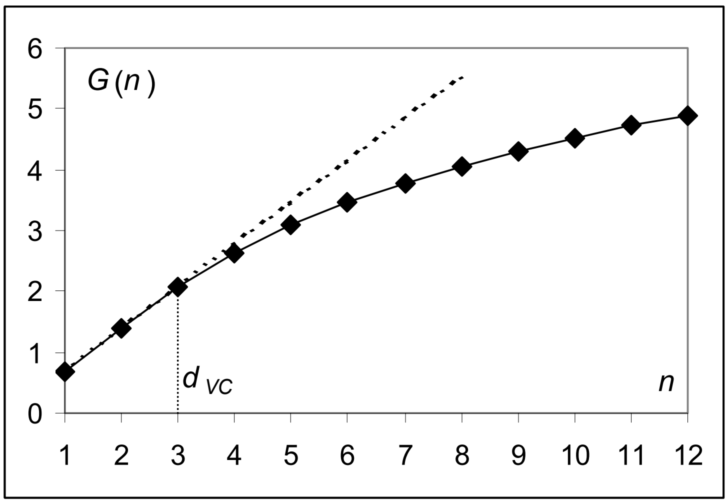
Example

MLP with d=2, m=1.

$$N(n) = 2n + (n-1)(n-2) = n^2 - n + 2, \text{ for } n>3$$

Therefore,

$$G(n) = \ln(n^2 - n + 2)\ , \text{ for } n>3$$



Let, for n>3 :

$$H(n) = d_{VC}(1 + \ln\frac{n}{d_{VC}}) = 3(1 + \ln\frac{n}{3}) = 3\ln\frac{ne}{3} = \ln(\frac{ne}{3})^3$$

For x > 1/2:

$$\ln x = \frac{x-1}{x} + \frac{(x-1)^2}{2x^2} + \ldots + \frac{(x-1)^k}{kx^k} + \ldots$$

But: $\frac{1}{x}\frac{(x-1)^k}{kx^k} \underset{x\to\infty}{\to} 0 \qquad \Rightarrow \qquad \frac{H(n)}{n} \underset{n\to\infty}{\to} 0$

Since: $n^2 - n + 2 < (\frac{ne}{3})^3$ We have: $\lim_{n\to\infty}\frac{G(n)}{n} = 0$

Practical importance of the preceding example:

Given:

One (arbitrary) dichotomy (concept).

Consider:

The perceptron implementing one linear discriminant designed with a training set (randomly drawn according to any distribution *D*)

Then:

Its empirical and true risks are guaranteed to converge to the optimal risk.

(i.e., the perceptron has generalization capability)

Likewise for any MLP since the d_{VC} is finite.

Regression case:

Let:

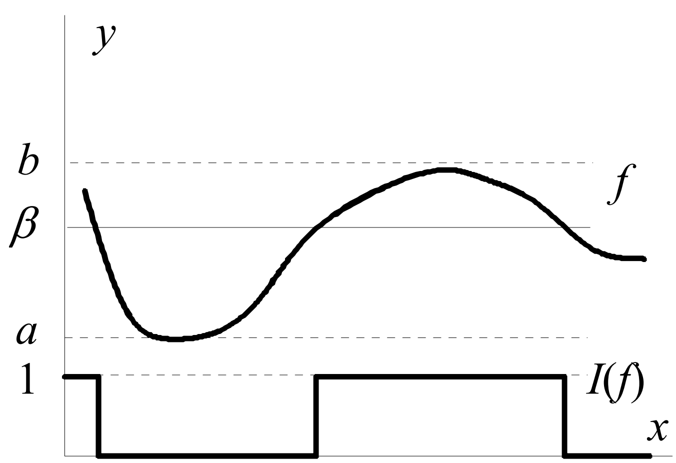
$f(x, \omega)$ be a family of functions bounded in [a, b]

and β a constant in the [a, b] interval.

Definition:

The d_{VC} of the $f(x, \omega)$ family is the d_{VC} of the following family of *indicator functions* with parameters ω and β.

$$I(f(\mathbf{x},\omega) > \beta) = \begin{cases} 1 & f(\mathbf{x},\omega) > \beta, \\ 0 & \text{otherwise.} \end{cases}$$

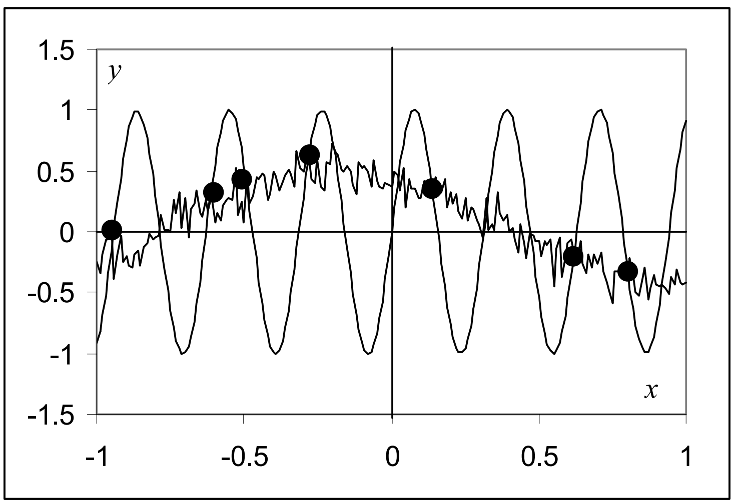


Example with infinite d_{VC} :

$$f(x, w) = \sin wx$$

$$I\,(\sin wx > 0)$$

Given any set of n points it is always possible to find a sine that interpolates (shatters) them.



Training set (black dots) with null empirical error.

The empirical error is always zero.

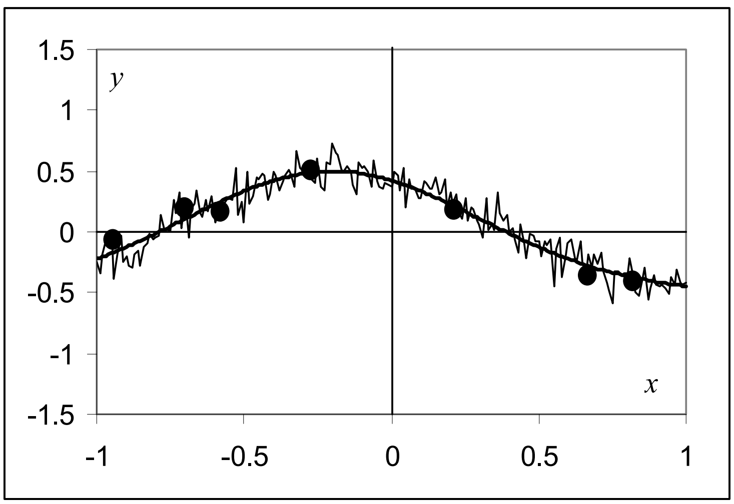
The true error is different from zero.

Example with finite d_{VC} :

Family of radial kernels:

$$f(x,c,\sigma) = K\left(\frac{|x-c|}{\sigma}\right)$$

$$d_{VC} = 2$$



The ERM principle is consistent and of fast learning

5.2 Validity of Inductive Theories

How to assess whether an inductive Theory is true or false?

Demarcation principle (Karl Popper, 1968):

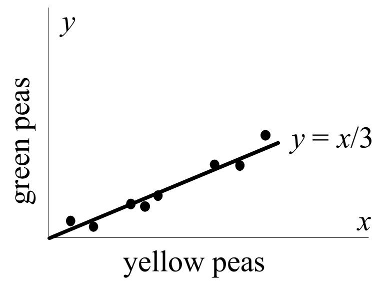
For an inductive Theory to be true it is necessary that the Theory can be *falsifiable*, i.e., assertions (facts) can be presented in the domain of the Theory that it cannot explain.

Consider an inductive Theory to which corresponds a hypotheses space with finite d_{VC}.

Then, the growth function is bounded, i.e. there are facts in the domain of the Theory that it cannot explain.

Examples:

Heredity (Mendel)



Theory: Each generation presents a constant proportionality, a, between dominant and recessive characters.

Hypotheses Space: $$H = \{y = f(x) = ax; \quad a \in \Re^+\}$$

d_{VC} finite; ERM consistent

Assertion falsifying the Theory: the proportion between green and yellow peas increases with each generation.

Astrology

"Theory": The "influence" of a planet in the individual x depends on the planet position in the Zodiac (angular elevation α and azimuth θ) and on the month, m, in which the individual was born.

Hypotheses Space:

$$H = \{f(\alpha, \theta, m); \quad a \in [0,90], \theta \in [-180,180], \quad m \in \{1, \cdots, 12\}\}$$

d_{VC} infinite; The "Theory" explains all the facts.

5.3 Structural and Guaranteed Risks

We consider only the classification case, where the risk is associated to the probability of misclassification.

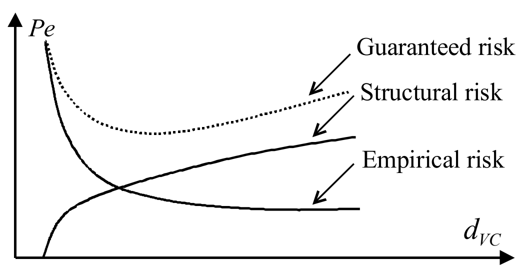
Error bound with finite d_{VC} (Vapnik, 1998):

$$P\left(\sup_{\mathbf{w}}\left|Pe(h(\mathbf{w})) - Pe_{emp}(h(\mathbf{w}))\right| > \varepsilon\right) < \left(\frac{2en}{d_{VC}}\right)^{d_{VC}} e^{-\varepsilon^2 n/4}$$

Thus, for finite d_{VC}, learning is PAC, with:

$$Pe(h(\mathbf{w})) \le Pe_{emp}(h(\mathbf{w})) + \sqrt{\frac{d_{VC}}{n}\ln\left(\frac{2n}{d_{VC}}+1\right) - \frac{1}{n}\ln\left(\frac{\alpha}{n}\right)}$$

The second term quantifies the *structural complexity* of the model.



Structural Risk Minimization (SRM) principle:

- Define a sequence of MLPs with growing d_{VC} (adding hidden neurons).
- For each MLP minimize the empirical risk.
- Progress to a more complex MLP until reaching the minimum of the guaranteed risk.

6 Sample Complexity in Infinite Hypothesis Spaces

6.1 Bounds on PAC Learning

Definition:

Let C be a class of concepts, $C \subseteq 2^X$. The Vapnik-Chervonenkis dimension of C, $d_{VC}(C)$, is the cardinality of the largest finite set of points $X_n \subseteq X$ that is shattered by C.
If arbitrarily large sets of points can be shattered by C, $d_{VC}(C)$ is infinite.

Theorem (Blumer *et al.*, 1989):

Let C be a class of concepts and H a hypothesis space. Then:

i. C is PAC-learnable iff $d_{VC}(C)$ is finite.

ii. If $d_{VC}(C)$ is finite, then:

(a) For $0 < \varepsilon < 1$ and sample size at least

$$n_u = \max\left[\frac{4}{\varepsilon}\log_2\left(\frac{2}{\delta}\right), \frac{8d_{VC}(C)}{\varepsilon}\log_2\left(\frac{13}{\varepsilon}\right)\right], \qquad (3)$$

any consistent algorithm is of PAC learning for C.

(b) For $0 < \varepsilon < 1/2$ and sample size less than

$$n_l = \max\left[\frac{1-\varepsilon}{\varepsilon}\ln\left(\frac{1}{\delta}\right), d_{VC}(C)\left(1-2\left(\varepsilon\left(1-\delta\right)+\delta\right)\right)\right], \quad (4)$$

no learning algorithm, for any hypothesis space H, is of PAC learning for C.

Application to MLPs:

Lower bound, n_l: ε: acceptable *Pe*
Use formula (4) with formula (1).

Upper bound, n_u: Use formula (2) with formula (3).
(unrealistically high)

Baum and Haussler (1989) have shown that an MLP with *u* neurons, *w* weights and training error ε will have a test error of at most 2ε for:

$$n_u = \frac{32w}{\varepsilon}\ln\left(\frac{32u}{\varepsilon}\right),$$

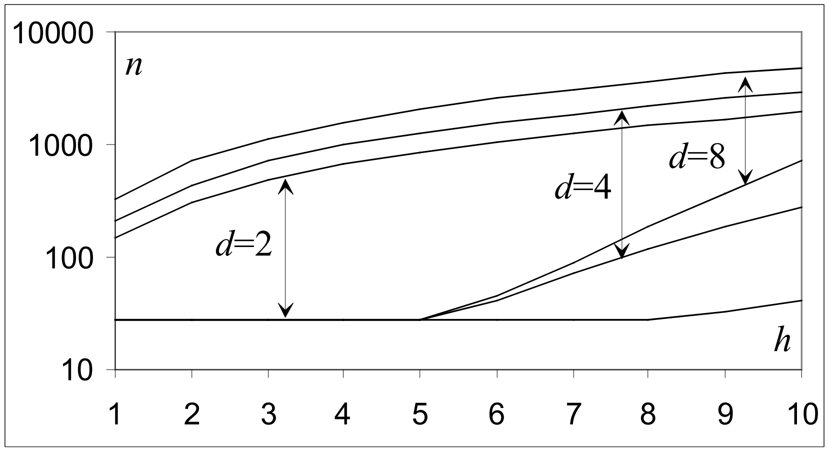
with confidence parameter

$$\delta = 8\left(2uen_u / w\right)^w e^{-\varepsilon n_u/16}.$$

δ is very low (δ<0.005) even for low values of *d* and *m*.

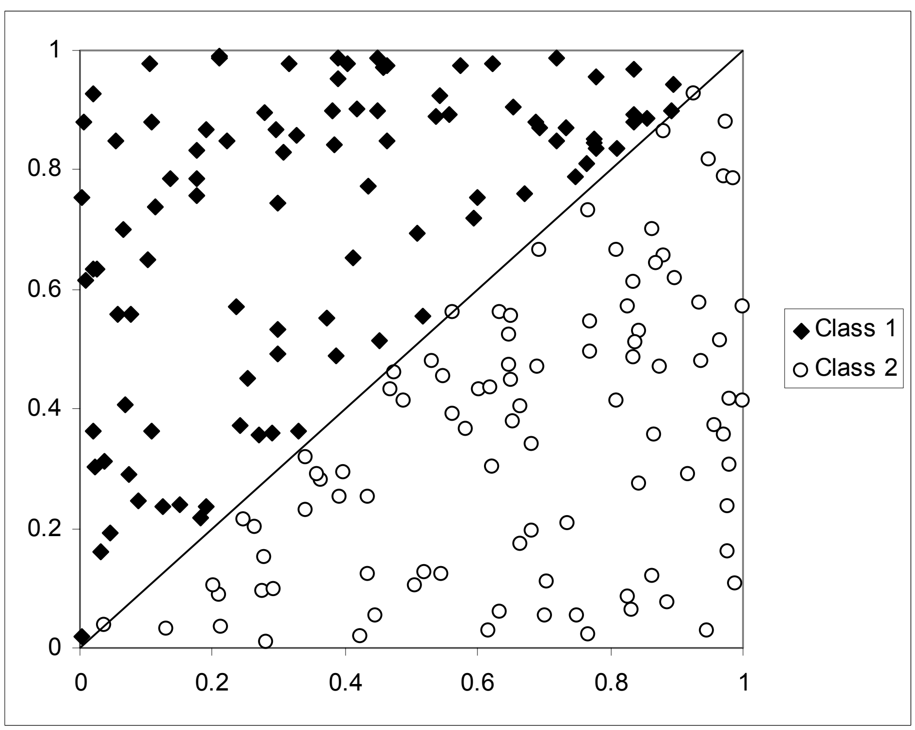
Practical rule: w/ε for complex MLPs.

Bounds of *n* for ε = 0.05 and δ = 0.01.



6.2 Study Case

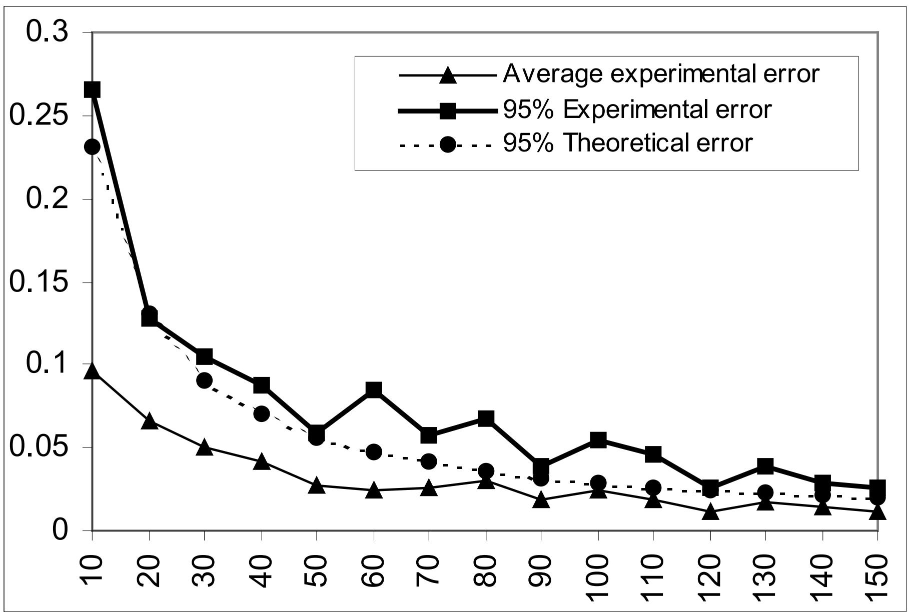
Two classes of points distributed in $[0, 1]^2$, linearly separable.



- Ideal hypothesis: $x_2 = x_1$
- Sampling distribution D : uniform distribution

Experiments with single perceptron (MLP2:1)

For each n=10, 20,...,150 value, 25 sets, X_n, are generated and the MLP2:1 solutions obtained. For each perceptron solution the exact error is computed.



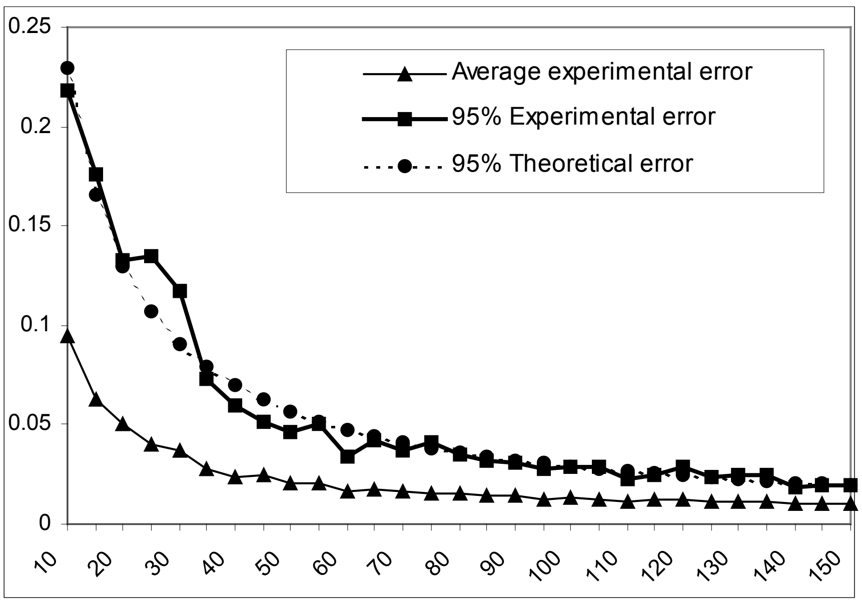
Average error in 25 experiments

95% percentile of the errors in 25 experiments

Error, ε, corresponding to δ=95% for $n_l = n$ and d_{VC}=3 (Blumer *et al.* formula)

Experiments with linear SVM

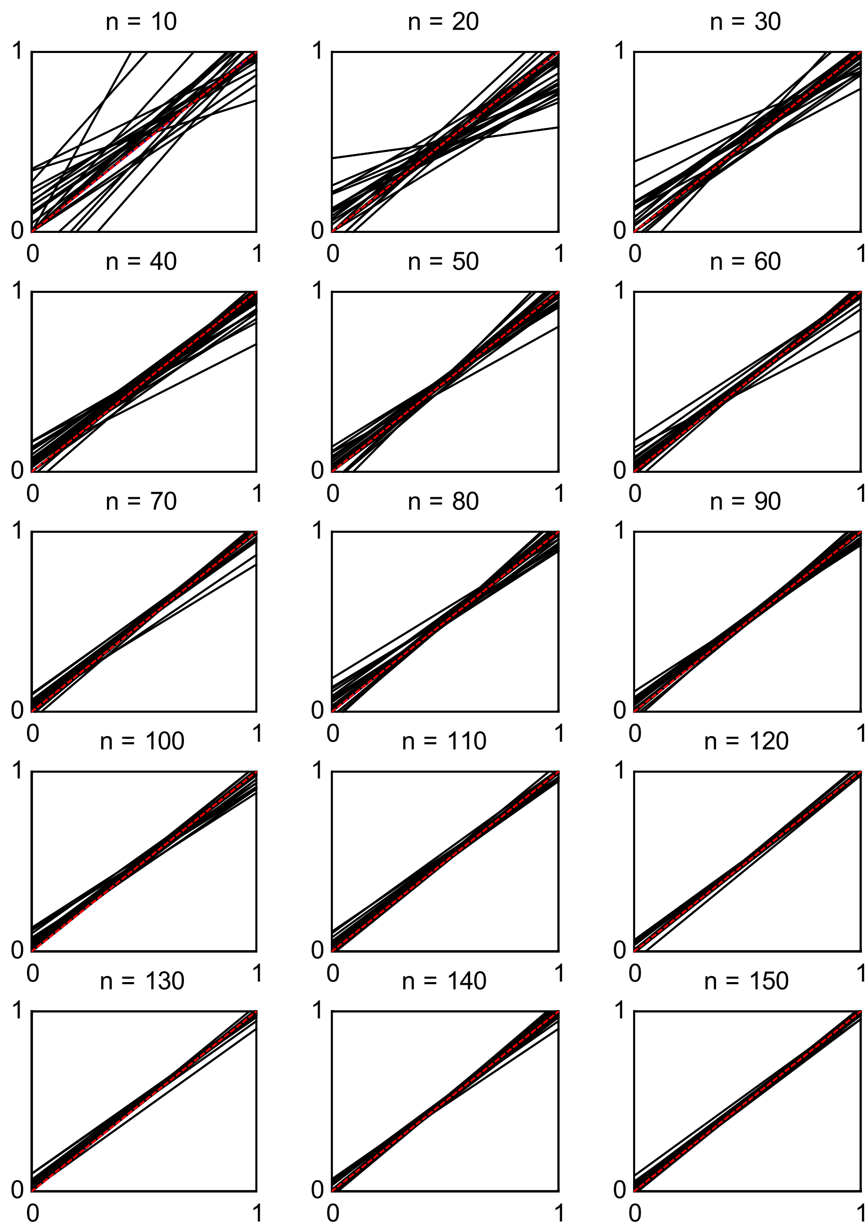
For each n=10, 15, ...,150 value, 200 sets, X_n, are generated and the respective SVM determined. For each SVM the exact error is computed.



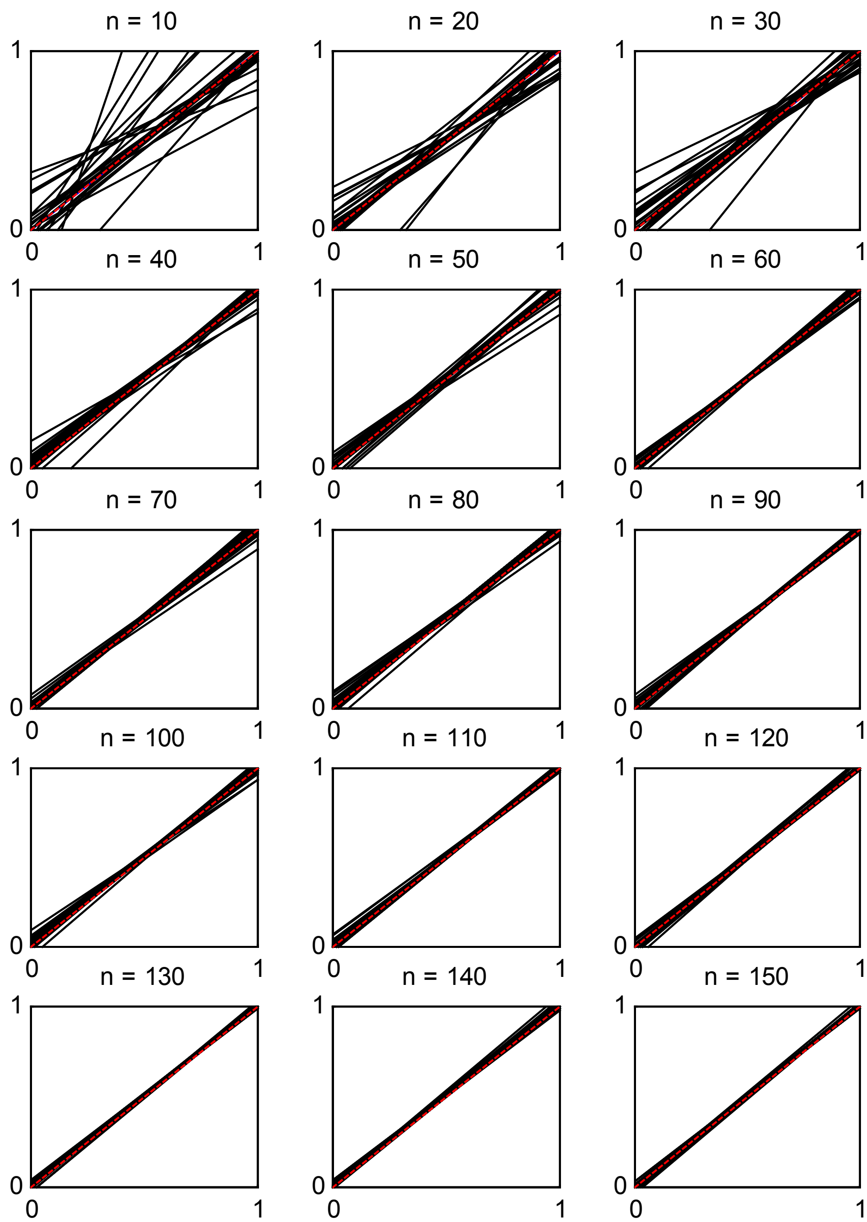
Average error in 200 experiments

95% percentile of the errors in 200 experiments

Error, ε, corresponding to δ=95% for $n_l = n$ and d_{VC}=3 (Blumer *et al.*formula)

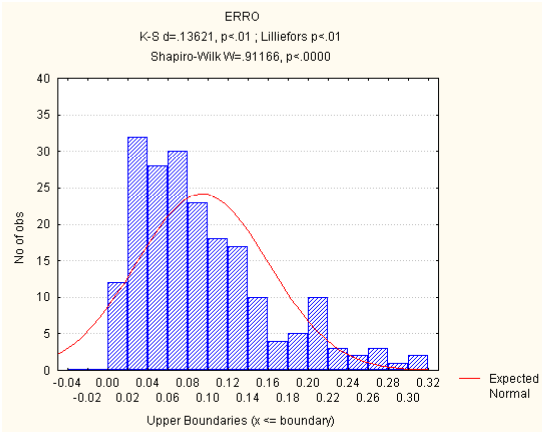


Linear discriminants produced by a Perceptron

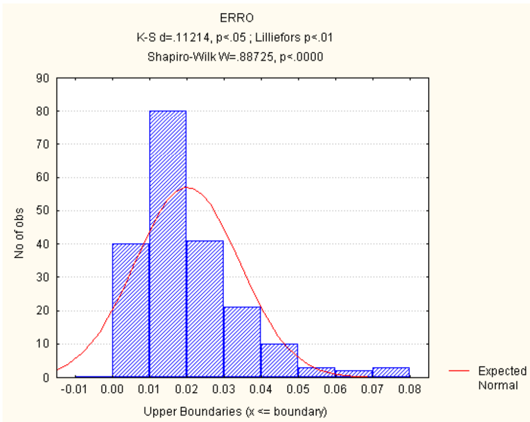


Linear discriminants produced by a Support Vector Machine

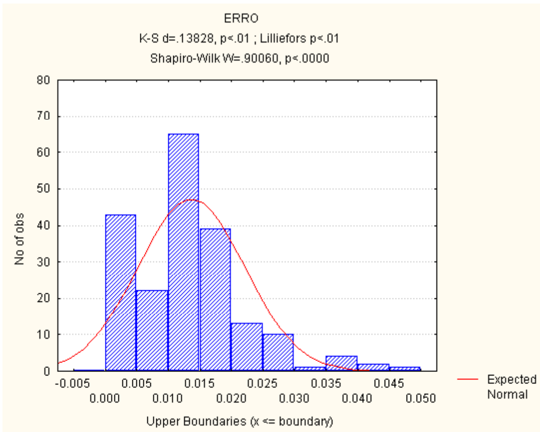
Error histograms (SVM):



n=10



n=50



n=100