Classification and Pattern Recognition

Léon Bottou

NEC Labs America

COS 424 – 2/23/2010

The machine learning mix and match

Goals	Classification, clustering, regression, other.
Representation	Parametric vs. kernels vs. nonparametric Probabilistic vs. nonprobabilistic Linear vs. nonlinear Deep vs. shallow
Capacity Control	Explicit: architecture, feature selection Explicit: regularization, priors Implicit: approximate optimization Implicit: bayesian averaging, ensembles
Operational Considerations	Loss functions Budget constraints Online vs. offline
Computational Considerations	Exact algorithms for small datasets. Stochastic algorithms for big datasets. Parallel algorithms.

Topics for today's lecture

Goals	**Classification**, clustering, regression, other.
Representation	Parametric vs. kernels vs. nonparametric Probabilistic vs. nonprobabilistic Linear vs. nonlinear Deep vs. shallow
Capacity Control	Explicit: architecture, feature selection Explicit: regularization, priors Implicit: approximate optimization Implicit: bayesian averaging, ensembles
Operational Considerations	**Loss functions** Budget constraints Online vs. offline
Computational Considerations	Exact algorithms for small datasets. Stochastic algorithms for big datasets. Parallel algorithms.

Summary

1. Bayesian decision theory
2. Nearest neigbours
3. Parametric classifiers
4. Surrogate loss functions
5. ROC curve.
6. Multiclass and multilabel problems

Classification a.k.a. Pattern recognition

Association between patterns $x \in \mathcal{X}$ and classes $y \in \mathcal{Y}$.

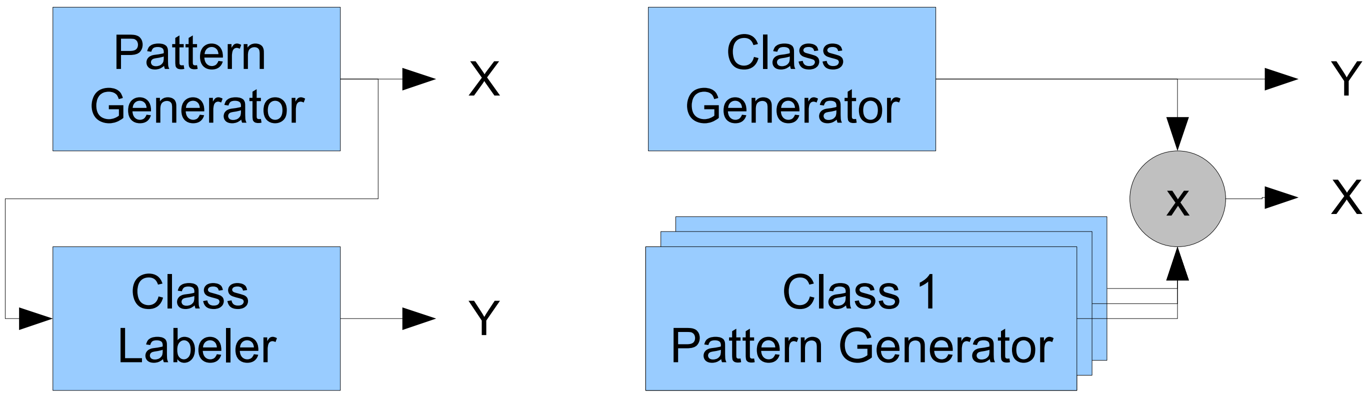
- The pattern space $\mathcal{X}$ is unspecified. For instance, $\mathcal{X} = \mathbb{R}^d$.
- The class space $\mathcal{Y}$ is an unordered finite set.

Examples:

- Binary classification ($\mathcal{Y} = \{\pm 1\}$).
  Fraud detection, anomaly detection,. . .
- Multiclass classification: ($\mathcal{Y} = \{C_1, C_2, \dots C_M\}$)
  Object recognition, speaker identification, face recognition,. . .
- Multilabel classification: ($\mathcal{Y}$ is a power set).
  Document topic recognition,. . .
- Sequence recognition: ($\mathcal{Y}$ contains sequences).
  Speech recognition, signal identification, ...

Probabilistic model

Patterns and classes are represented by random variables X and Y.



$$P(X,Y) = P(X)\,P(Y|X) = P(Y)\,P(X|Y)$$

Bayes decision theory

Consider a classifier $x \in \mathcal{X} \mapsto f(x) \in \mathcal{Y}$.

Maximixe the probability of correct answer:

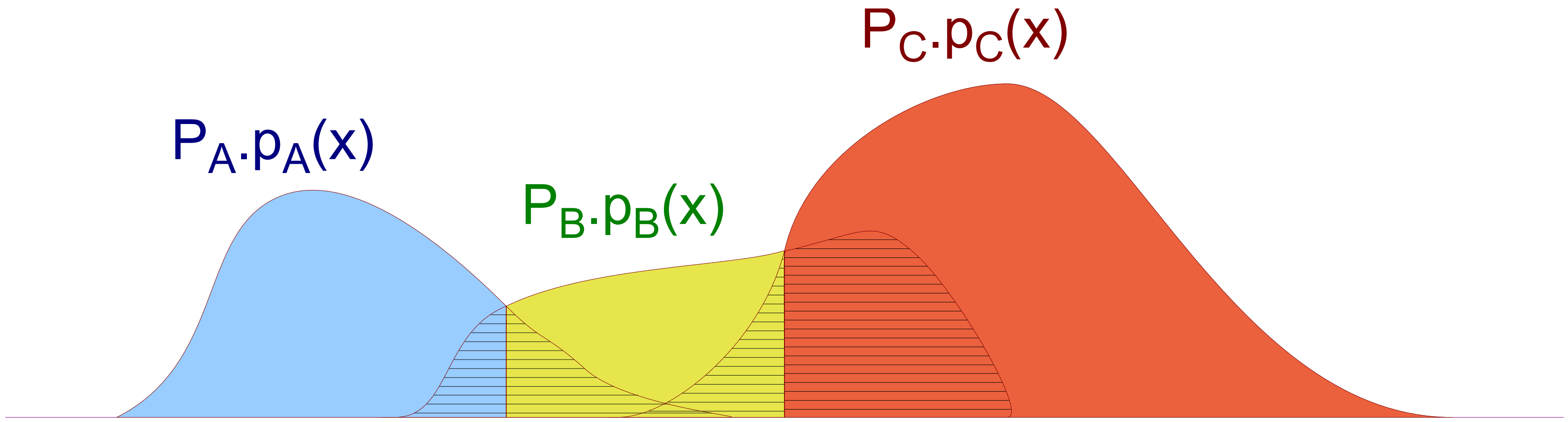
$$\begin{aligned}
\mathbb{P}\{f(X) = Y\} &= \int \mathbb{1}(f(x) = y)\, dP(x, y) \\
&= \int \sum_{y \in \mathcal{Y}} \mathbb{1}(f(x) = y)\, \mathbb{P}\{Y = y | X = x\}\; dP(x) \\
&= \int \mathbb{P}\{Y = f(x) | X = x\}\; dP(x)
\end{aligned}$$

Bayes optimal decision rule: $\boldsymbol{f^*(x) = \arg\max_{y \in \mathcal{Y}} \mathbb{P}\{Y = y | X = x\}}$

Bayes optimal error rate: $\boldsymbol{\mathcal{B} = 1 - \int \max_{y \in \mathcal{Y}} \mathbb{P}\{Y = y | X = x\}\;\; dP(x)}$.

Bayes optimal decision rule

Comparing class densities $p_y(x)$ scaled by the class priors $P_y = \mathbb{P}\{Y = y\}$:



Hatched area represents the Bayes optimal error rate.

How to build a classifier from data

Given a finite set of training examples $\{(x_1, y_1), \ldots, (x_n, y_m)\}$?

- **Estimating probabilities**:
  - Find a plausible probability distribution (next lecture).
  - Compute or approximate the optimal Bayes classifier.
- **Minimize empirical error**:
  - Choose a parametrized family of classification functions a priori.
  - Pick one that minimize the observed error rate.
- **Nearest neighbours**:
  - Determine class of x on the basis of the closest example(s).

Nearest neighbours

Let $d(x, x')$ be a distance on the patterns.

Nearest neighbour rule (1NN)

– Give x the class of the closest training example.

– $f_{\mathsf{nn}}(x) = y_{\mathsf{nn}(x)}$ with $\mathsf{nn}(x) = \arg\min_i d(x, x_i)$.

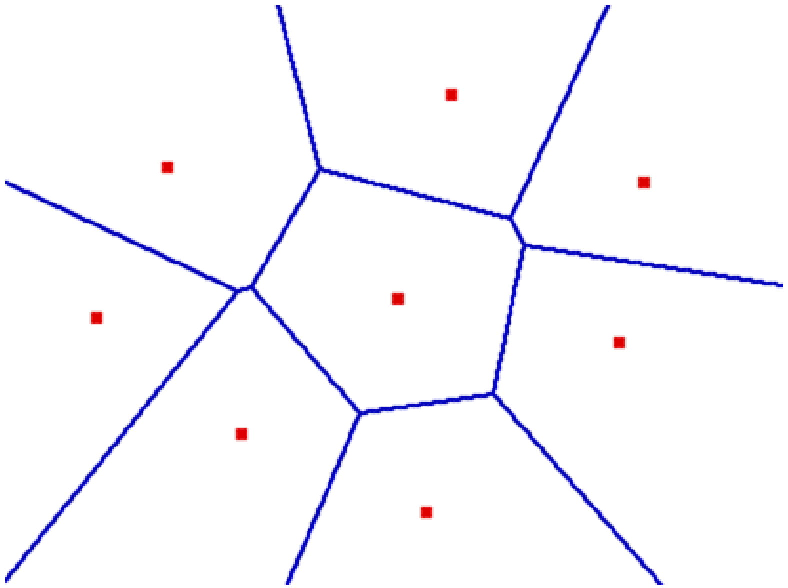
***K*-Nearest neighbours rule (kNN)**

– Give x the most frequent class among the K closest training examples.

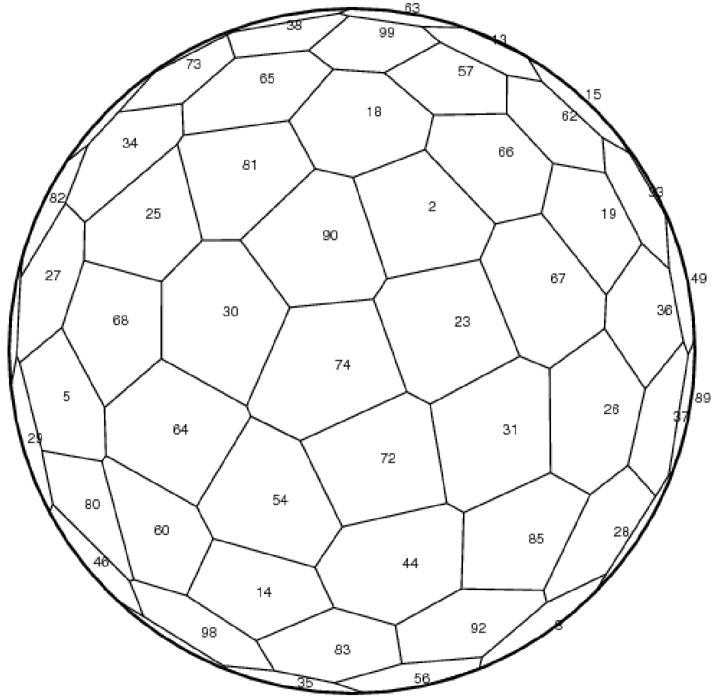
***K*-Nearest neighbours variants**

– Weighted votes (according the the distances)

Voronoi tesselation



Euclian distance in the plane



Cosine distance on the sphere

– 1NN: Piecewise constant classifier defined on the Voronoi cells.
– kNN: Same, but with smaller cells and additional constraints.

1NN and Optimal Bayes Error

Theorem (Cover & Hart, 1967) :

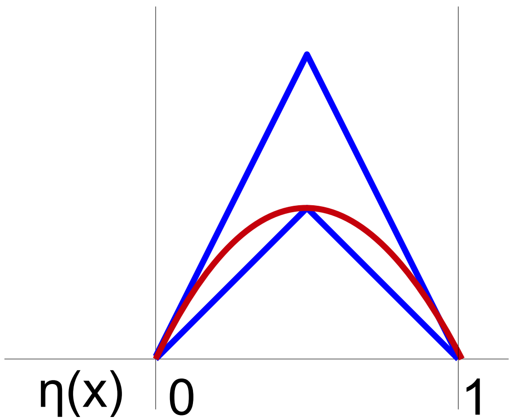
Assume $\boldsymbol{\eta_y(x)} = \mathbb{P}\left\{\boldsymbol{Y = y | X = x}\right\}$ is continuous.

When $\boldsymbol{n} \rightarrow \infty$, $\boldsymbol{\mathcal{B}} \leq \mathbb{P}\left\{\boldsymbol{f_{\mathsf{nn}}(X) \neq Y}\right\} \leq \boldsymbol{2\mathcal{B}}$.

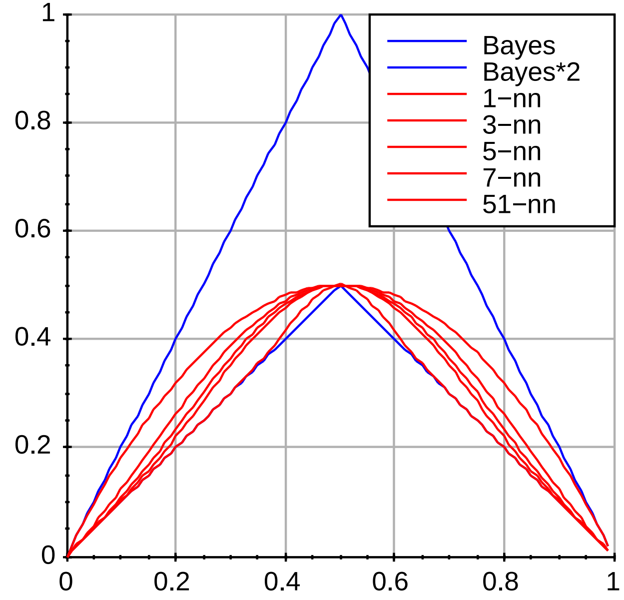
Easy proof when there are only two classes

Let $\eta(x) = \mathbb{P}\left\{Y = +1 | X = x\right\}$.

- $\mathcal{B} \;=\; \int \min(\eta(x), 1 - \eta(x))\, dP(x)$
- $\mathbb{P}\left\{f_{\mathsf{nn}}(X) \neq Y\right\}$
  $= \int \eta(x)(1 - \eta(x^*)) + (1 - \eta(x))\eta(x^*)\, dP(x)$
  $\rightarrow \int 2\,\eta(x)(1 - \eta(x))\, dP(x)$



1NN versus kNN



Using more neighbours

– Is to Bayes rule in the limit.

– Needs more examples to approach the condition $\eta(x_{k\mathrm{nn}(x)}) \approx \eta(x)$

K is a capacity parameter

– to be determined using a validation set.

Computation

Straightforward implementation
– Computing $f(x)$ requires n distance computations.
– $(-)$ Grows with the number of examples.
– $(+)$ Embarrassingly parallelizable.

Data structures to speedup the search: K-D trees
– $(+)$ Very effective in low dimension
– $(-)$ Nearly useless in high dimension

Shortcutting the computation of distances
– Stop computing as soon as a distance gets non-competitive.

Use the triangular inequality $\boldsymbol{d(x, x_i) \geq |d(x, x') - d(x_i, x')|}$
– Pick r well spread patterns $x_{(1)} \ldots x_{(r)}$.
– Precompute $d(x_i, x_{(j)})$ for $i = 1 \ldots n$ and $j = 1 \ldots r$.
– Lower bound $d(x, x_i) \geq \max_{j=1\ldots r} |d(x, x_{(j)}) - d(x_i, x_{(j)})|$.
– Shortcut if lower bound is not competitive.

Distances

Nearest Neighbour performance is sensitive to distance.

Euclidian distance: $d(x, x') = (x - x')^2$
– do not take the square root!

Mahalanobis distance: $d(x, x') = (x - x')^\top A\,(x - x')$
– Mahalanobis distance: $A = \Sigma^{-1}$
– Safe variant: $A = (\Sigma + \epsilon I)^{-1}$

Dimensionality reduction:
– Diagonalize $\Sigma = Q^\top \Lambda Q$.
– Drop the low eigenvalues and corresponding eigenvector.
– Define $\tilde{x} = \Lambda^{-1/2} Q\, x$. Precompute all the $\tilde{x}_i$.
– Compute $d(x, x_i) = (\tilde{x} - \tilde{x}_i)^2$.

Discriminant function

Binary classification: $y = \pm 1$

Discriminant function: $f_w(x)$

– Assigns class $\mathbf{sign}(\boldsymbol{f_w}(\boldsymbol{x}))$ to pattern $\boldsymbol{x}$.

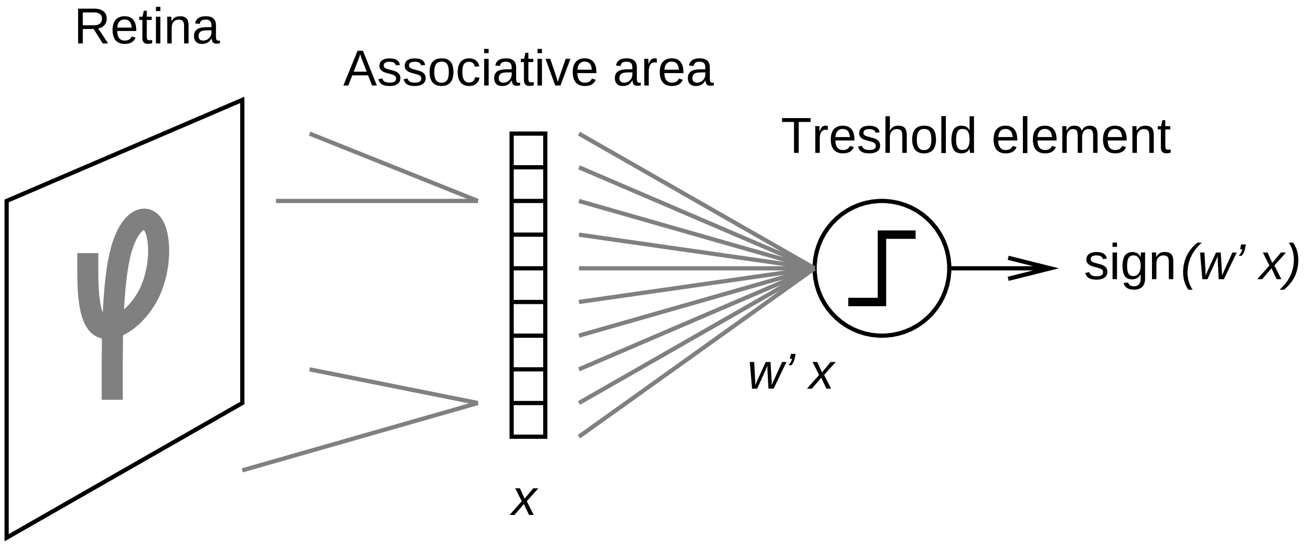
– Symbol $\boldsymbol{x}$ represents parameters to be learnt.

Example: Linear discriminant function

– $\boldsymbol{f_w}(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{\Phi}(\boldsymbol{x})$.

Example: The Perceptron

The perceptron is a linear discriminant function



The Perceptron Algorithm

– Initialize $w \leftarrow 0$.
– Loop
  – Pick example x_i, y_i
  – If $y_i\, w^\top \Phi(x_i) \leq 0$ then $w \leftarrow w + y_i\, \Phi(x_i)$
– Until all examples are correctly classified

Perceptron theorem

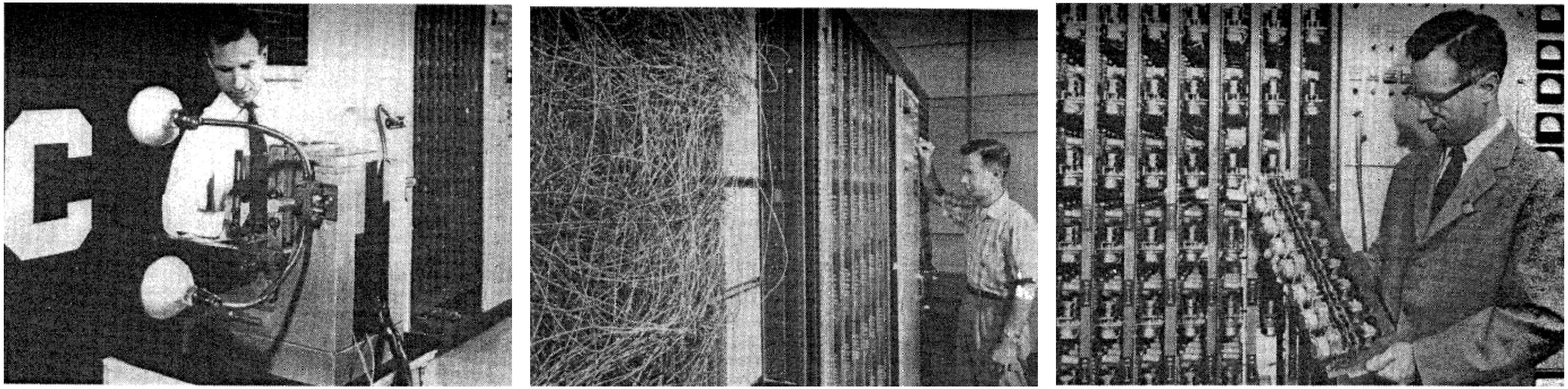
Guaranteed to stop if the training data is linearly separable

Perceptron via Stochastic Gradient Descent

SGD for minimizing $C(w) = \sum_i \max(0, -y_i\, w^\top \Phi(x_i))$ gives:

– If $y_i\, w^\top \Phi(x_i) \leq 0$ then $w \leftarrow w + \gamma\, y_i\, \Phi(x_i)$

The Perceptron Mark 1 (1957)



The Perceptron is not an algorithm.

The Perceptron is a machine!

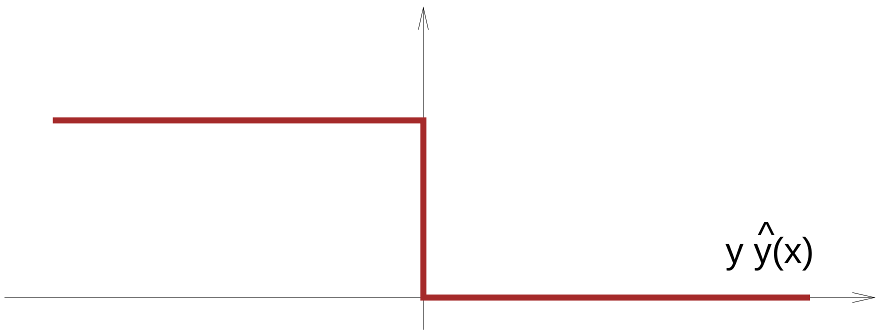
Minimize the empirical error rate

Empirical error rate

$$\min_{w} \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}\{y_i\, f(x_i, w) \leq 0\}$$

Misclassification loss function

– Noncontinuous
– Nondifferentiable
– Nonconvex



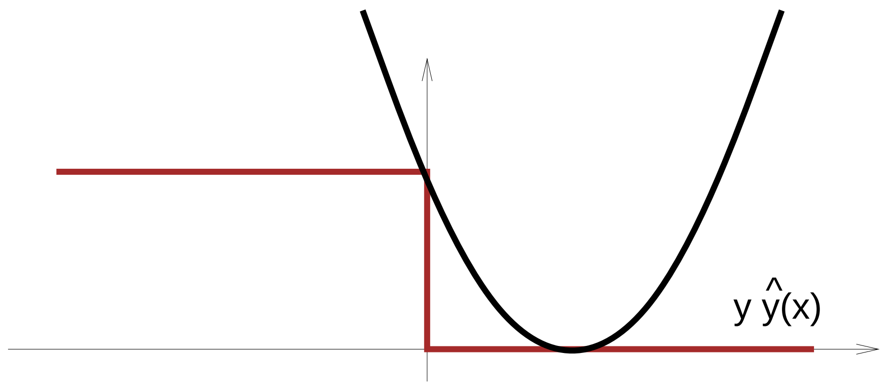
Surrogate loss function

Minimize instead

$$\min_{w} \frac{1}{n} \sum_{i=1}^{n} \ell(y_i\, f(x_i, w))$$

Quadratic surrogate loss

Quadratic:
$\ell(z) = (z - 1)^2$



Surrogate loss functions

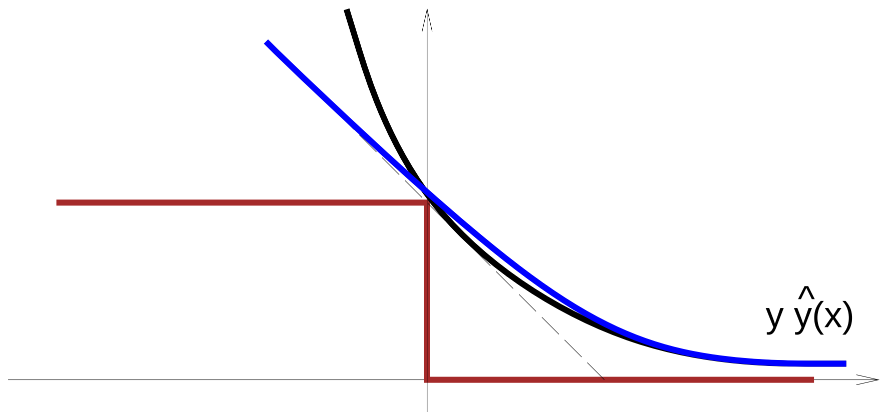
Exp loss and Log loss

Exp loss:

$$\ell(z) = \exp(-z)$$

Log loss:

$$\ell(z) = \log(1 + exp(-z))$$



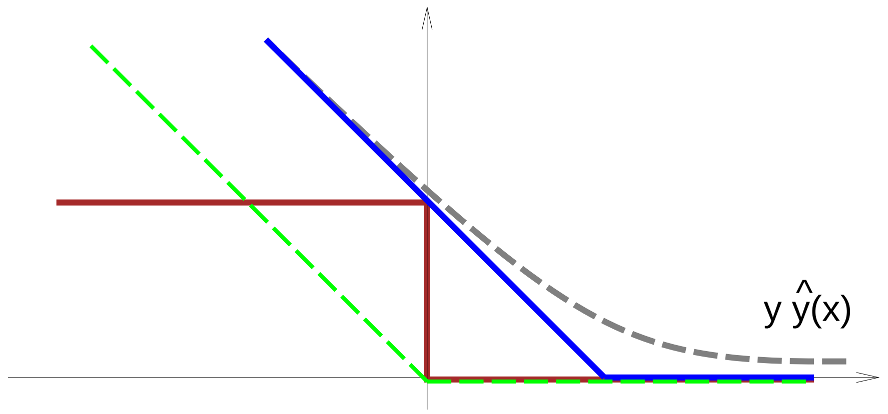
Hinges

Perceptron loss:

$$\ell(z) = \max(0, -z)$$

Hinge loss:

$$\ell(z) = \max(0, 1 - z)$$

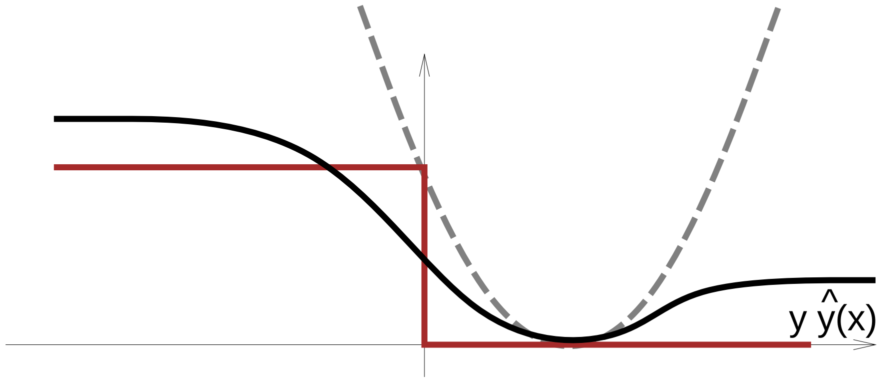


Surrogate loss function

Quadratic+Sigmoid

Let $\sigma(z) = \tanh(z)$.

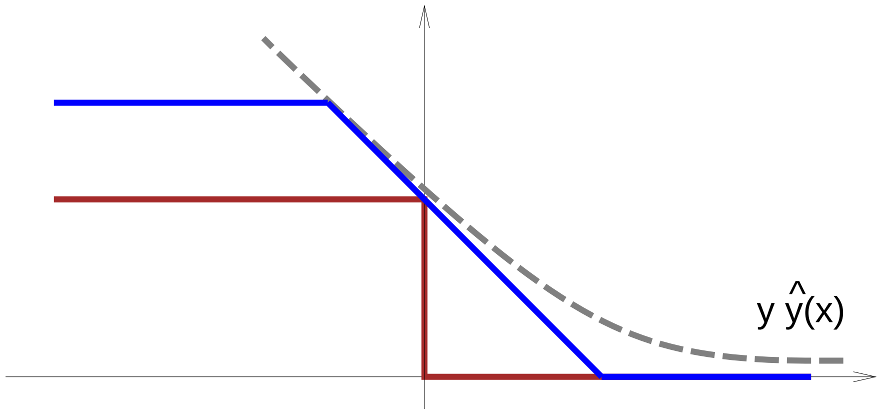
$$\boldsymbol{\ell(z) = (\sigma(\tfrac{3}{2}z) - 1)^2}$$



Ramp

Ramp loss:

$$\boldsymbol{\ell(z) = [1 - z]_+ - [s - z]_+}$$



Choice of a surrogate loss function

Constraints from the optimization algorithm

– A convex loss with a convex $f_w(x)$ ensures the unicity of the minimum.

– Optimization by gradient descent suggests differentiable losses.

– Dual optimizatoin methods work well with hinges.

Class calibrated loss

– In the limit $\min \int \left[\eta(x)\ell(f_w(x)) + (1-\eta(x))\ell(-f_w(x))\right] dP(x)$.

– Define $L(\eta, z) = \eta\ell(z) + (1-\eta)\ell(-z)$.

– If we had an infinite training set and a fully flexible $f_w(x)$,
we would have: $f(x) = \arg\min_z L(\mathbb{P}\left\{Y=+1|X=x\right\}, z)$.

– Examples.

Asymmetric cost problem

Binary classification.

– Positive class $y = +1$, negative class $y = -1$.

Examples of positive classes.

– fraudulent credit card transaction
– relevant document for a given query
– heart failure detection

Different kinds of errors have different costs.

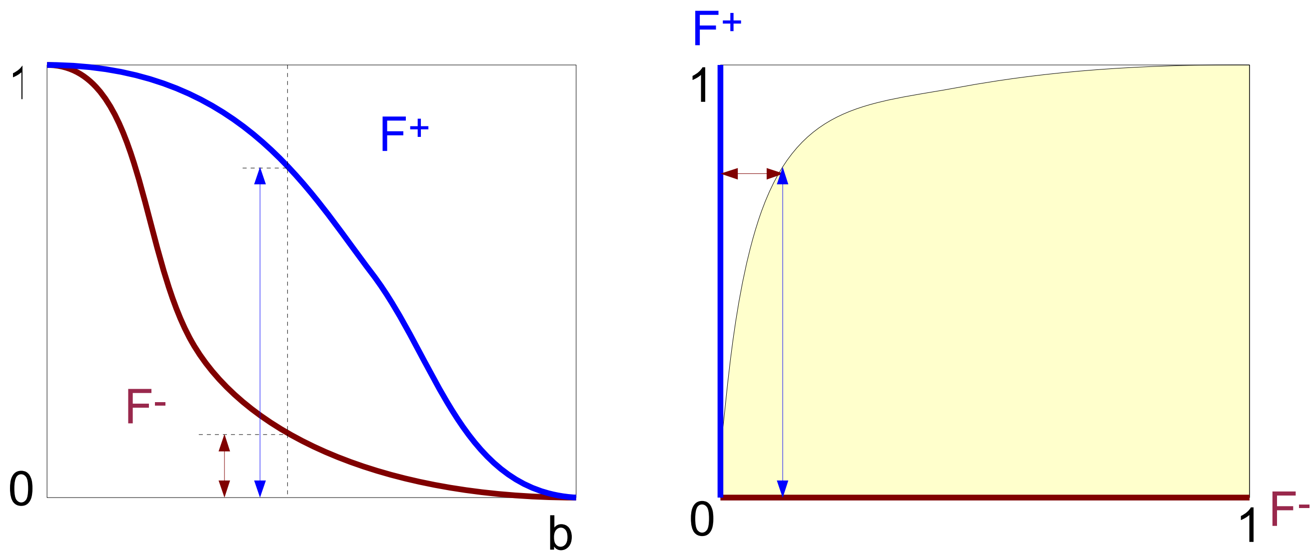
– False positive, false detection, false alarm.
– False negative, non detection.

Costs are difficult to assess.

Receiver Operating Curve (ROC)

Changing the threshold

– Assigned class is $\operatorname{sign}(f(x) - b)$.
– True positives: $F_+(b) = \mathbb{P}\{f(x) - b > 0 | Y = +1\}$
– False positives: $F_-(b) = \mathbb{P}\{f(x) - b > 0 | Y = -1\}$



Optimal decision rule with asymmetric costs

Optimal asymmetric decision rule

– Let C_y be the cost of erroneously assigning class y to an example.

– We want to minimize $\displaystyle\int \sum_{y=\pm 1} C_y \, \mathbb{I}(f(x)=y) \, \mathbb{P}\{Y \neq y | X=x\} \, dP(x)$.

– $\displaystyle f(x) \;=\; \arg\min_{y=\pm 1} C_y \, \mathbb{P}\{Y \neq y | X=x\} \;=\; \mathrm{sign}\left(\eta(x) - \frac{C_+}{C_+ + C_-}\right)$

Optimal ROC curve

– The optimal decision rules have the form $\mathrm{sign}(f(x) - b)$

– Therefore $f(x) = \eta(x) = \mathbb{P}\{Y = +1 | X\}$ gives the optimal ROC curve.

– Same for monotone transformations of $f(x)$.

Empirical ROC

Ranking

Find a function $f_w(x)$ with ROC close to the optimal ROC.

Maximize Area Under Curve (AUC)

– We would like $\min \sum_{i\in\mathcal{P}} \sum_{j\in\mathcal{N}} \mathbb{I}\{f(x_i,w) \leq f(x_j,w)\}$

– With a surrogate $\min \sum_{i\in\mathcal{P}} \sum_{j\in\mathcal{N}} \ell(f(x_i,w) - f(x_j,w))$

Ranking the best instances

– AUC often optimizes useless parts of the ROC curve.

– Various algorithms have been proposed to do better. . . .

What to do with more than two classes ?

Turning the problem into multiple binary classification problems.

- **One versus all** (M classifiers).
  - Classifier $f_k(x)$ detects class k.
  - Recognized class is $\arg\max_k f_x(x)$.
  - Each classifier is trained on the full dataset.
  - Dubious principle. Works well in practice.

- **One versus others** ($M(M-1)/2$ classifiers)
  - Classifier $f_{k,k'}$ separates class k from class k'.
  - Recognized class if $\arg\max_k \sum_{k'} f_{k,k'}(x)$.
  - Classifier $f_{k,k'}$ is trained on examples from classes k and k'.
  - Dubious principle. Often faster but sligtly worse.

What to do with more than two classes ?

Doing it right!

– Learn a function $S_w(x,y)$ that measures how well y goes with x.
– Recognized class $\arg\max_y S_w(x,y)$

Cost functions

Perceptron-like: $\displaystyle \min_w \frac{1}{n}\sum_{i=1}^{n} \; - S_w(x_i,y_i) + \max_y S_w(x_i,y)$

Hinge-like: $\displaystyle \min_w \frac{1}{n}\sum_{i=1}^{n} \; \max\left[1 - S_w(x_i,y_i) + \max_{y\neq y_i} S_w(x_i,y)\right]_+$

Logloss-like: $\displaystyle \min_w \frac{1}{n}\sum_{i=1}^{n} \; - S_w(x_i,y_i) + \log\left(\sum_y e^{S_w(x_i,y)}\right)$

Comments

– More costly than OVA.
– Not better than OVA in practice.

Multilabel Problems

Documents can treat multiple topics.

Therefore y is a subset of the set of topics.

Simple approach

– One binary classification for each topic.

– But labels are not independent: taxonomies, related topics.

Complex scoring functions

– $f_k(x)$ gives a score for document x and topic k.

– $R_w(y)$ measures the compatibility the topic set y.

– Recognized topics: $\displaystyle \arg\max_{y_1 \ldots y_k} \; R_w(\{y_1 \ldots y_k\}) + \sum_k f_k(x)$.

– Same loss functions as the multiclass problem.