UCB - CS189
Introduction to Machine Learning
Fall 2015
Lecture 5: Shrinkage

Isabelle Guyon

ChaLearn

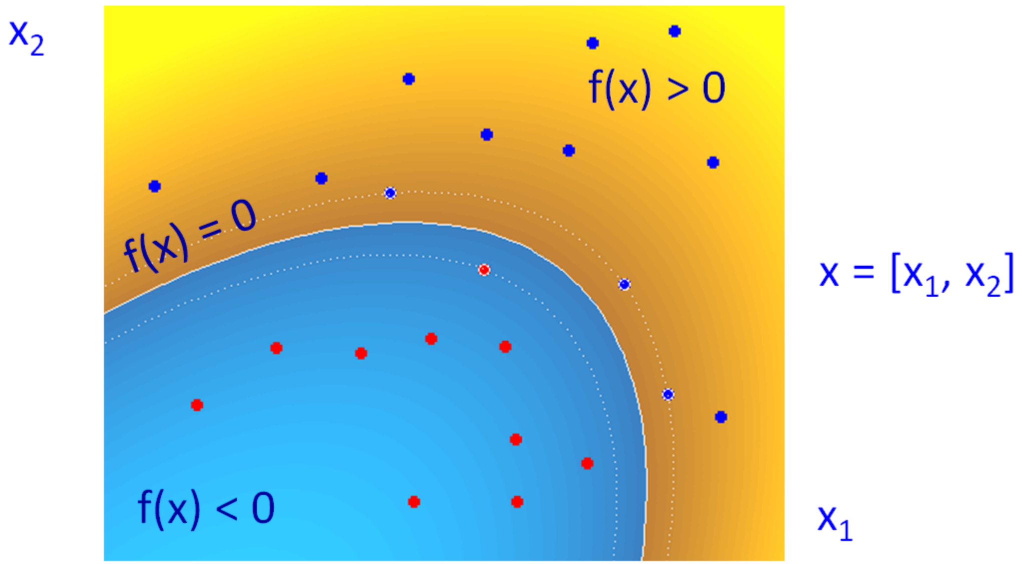
Come to my office hours…

Wed 2:30-4:30 Soda 329

Last time

(HARD)

Non-linear optimum margin classifier



$f(x) = \sum_k \alpha_k\, k(x_k, x) + b$

SVM, Boser-Guyon-Vapnik, 1992

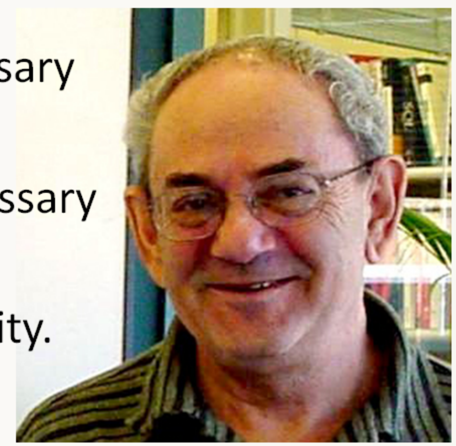
Come to my office hours…
Wed 2:30-4:30 Soda 329

Today

For Ockham to Vapnik

(→ SOFT margin)

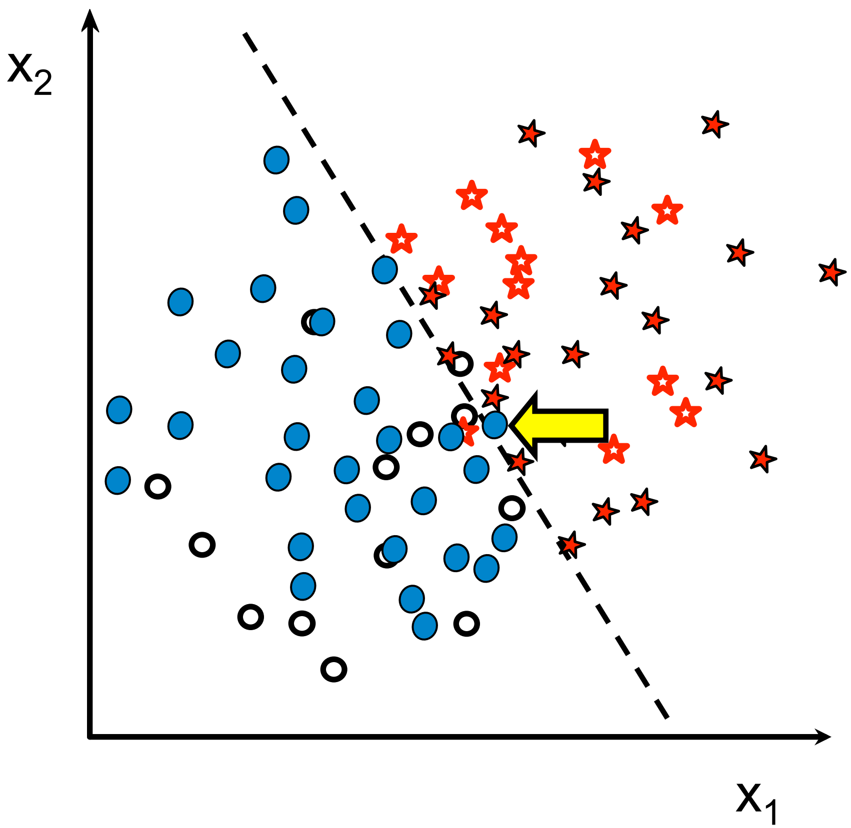
- Shave off unnecessary parameters.
- Forget the unnecessary memories.
- Minimize complexity.
- Minimize ||**w**||.

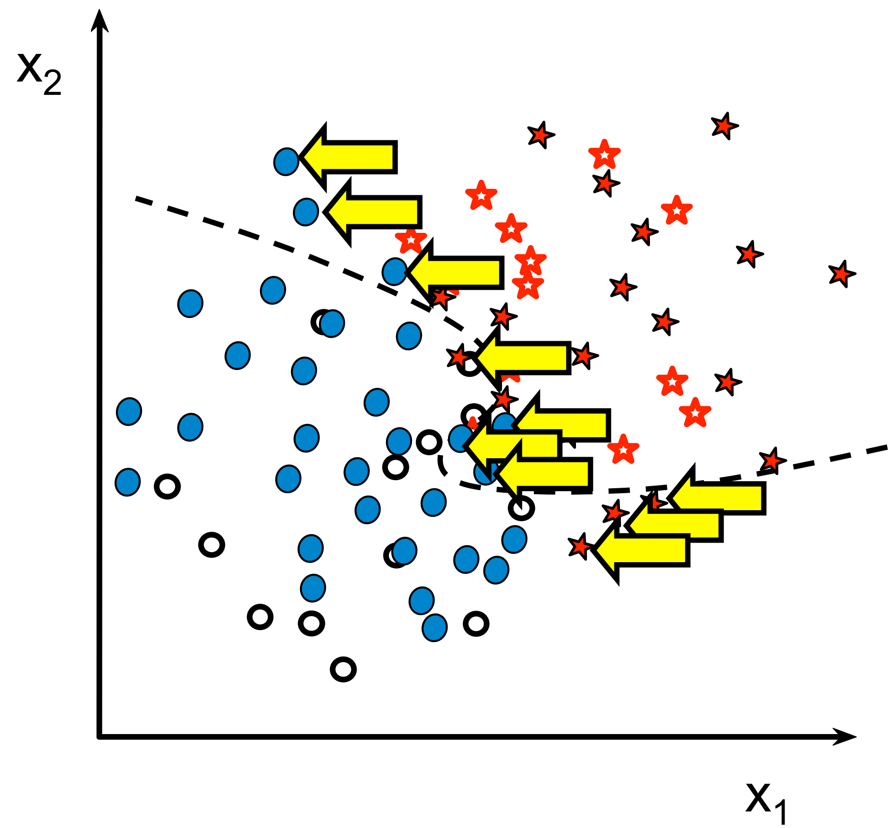


Math prerequisites

- Derivative
- Chain rule
- Lagrange multiplier

Fit / Robustness Tradeoff





Reasons for non-linear separability

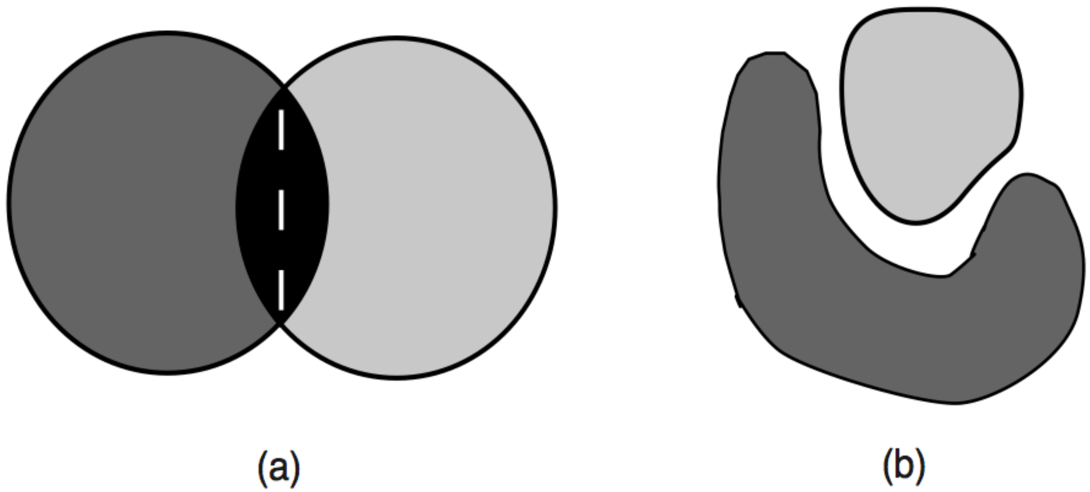


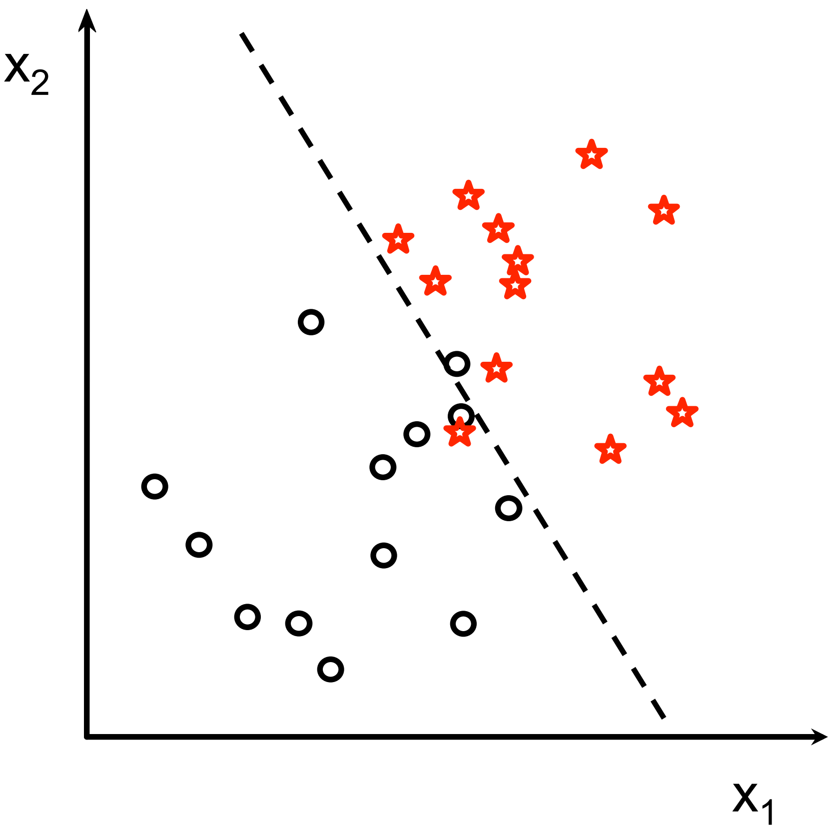
Figure 9.1 Non-linear separability. (a) Overlapping classes. The optimum decision boundary may still be linear. (b) Non overlapping classes. In the case shown, the optimum decision boundary is not linear.

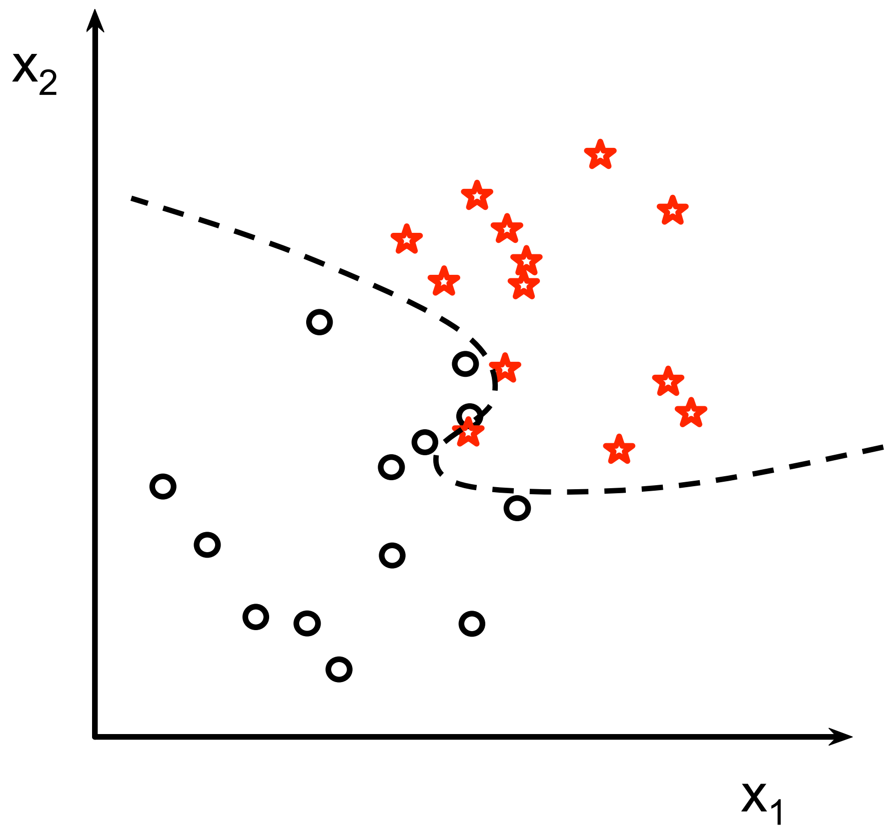
I. Guyon and D. Stork (2000). Linear Discriminant and Support Vector Classifiers.

Ockham's Razor

- Principle proposed by William of Ockham in the fourteenth century: "**Pluralitas non est ponenda sine neccesitate**".
- Of two theories providing similarly good predictions, prefer *the simplest one*.
- Shave off unnecessary parameters of your models.

Fit / Robustness Tradeoff

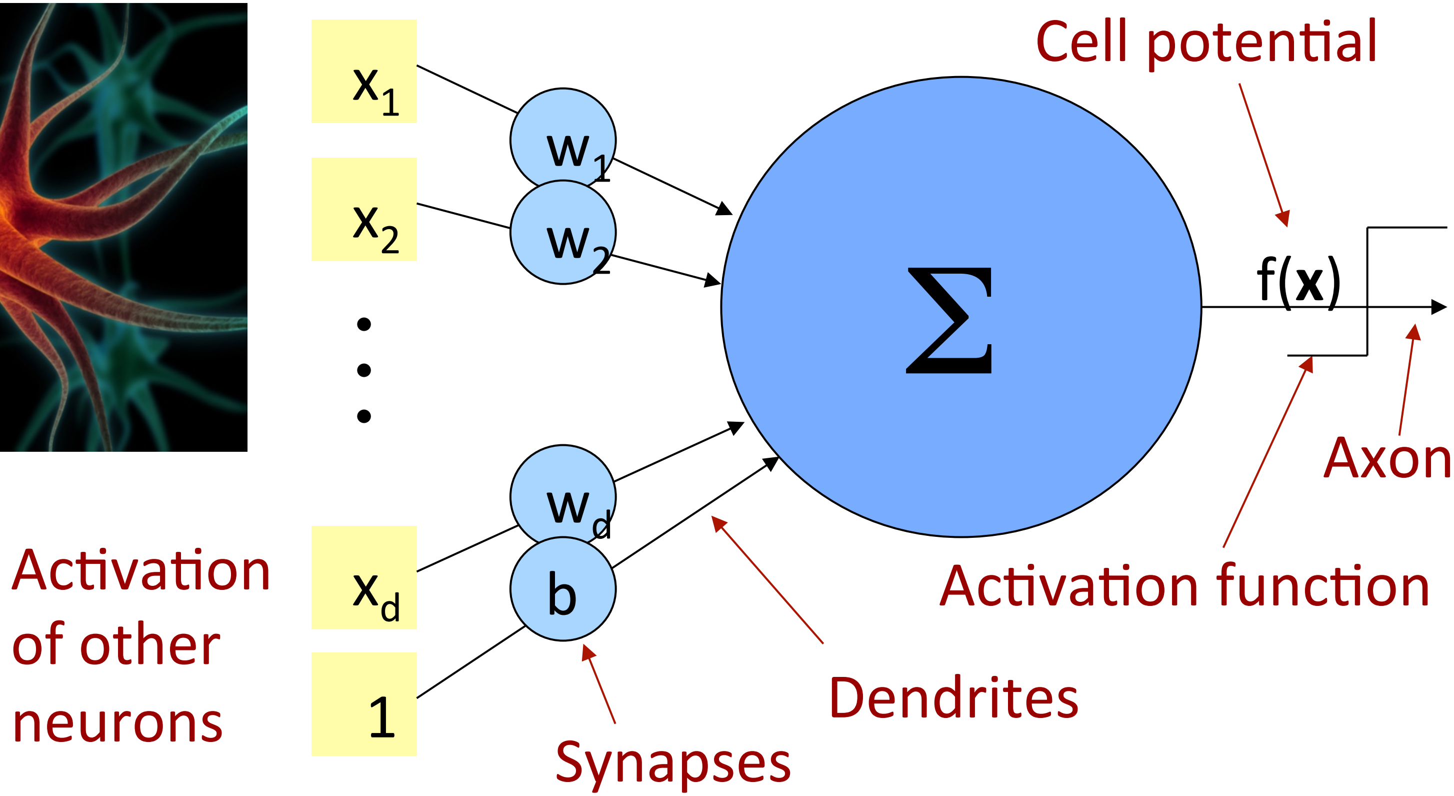




The Power of Amnesia

- The human **brain** is made out of billions of cells or Neurons, which are highly interconnected by synapses.
- Exposure to enriched environments with extra sensory and social stimulation enhances the **connectivity** of the synapses, but children and adolescents can lose them up to 20 million per day.

Artificial Neurons

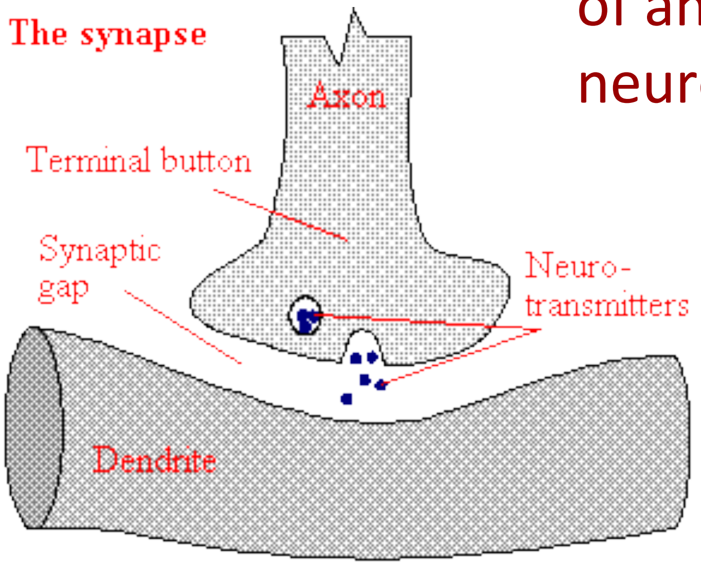


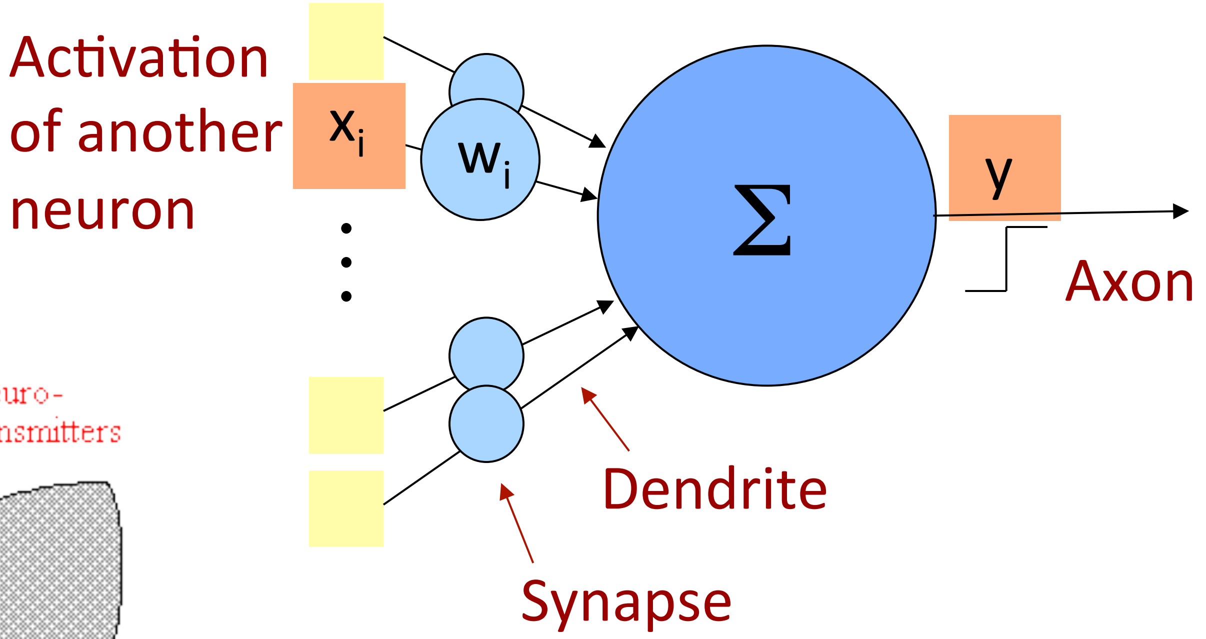
McCulloch and Pitts, 1943

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

Hebb's Rule

$$\boxed{w_i \leftarrow w_i + y^k x^k_i}$$





Weight Decay

$$w_i \leftarrow w_i + y^k x^k_i$$

Hebb's rule

$$\boxed{w_i \leftarrow (1-\gamma)\, w_i + y^k x^k_i}$$

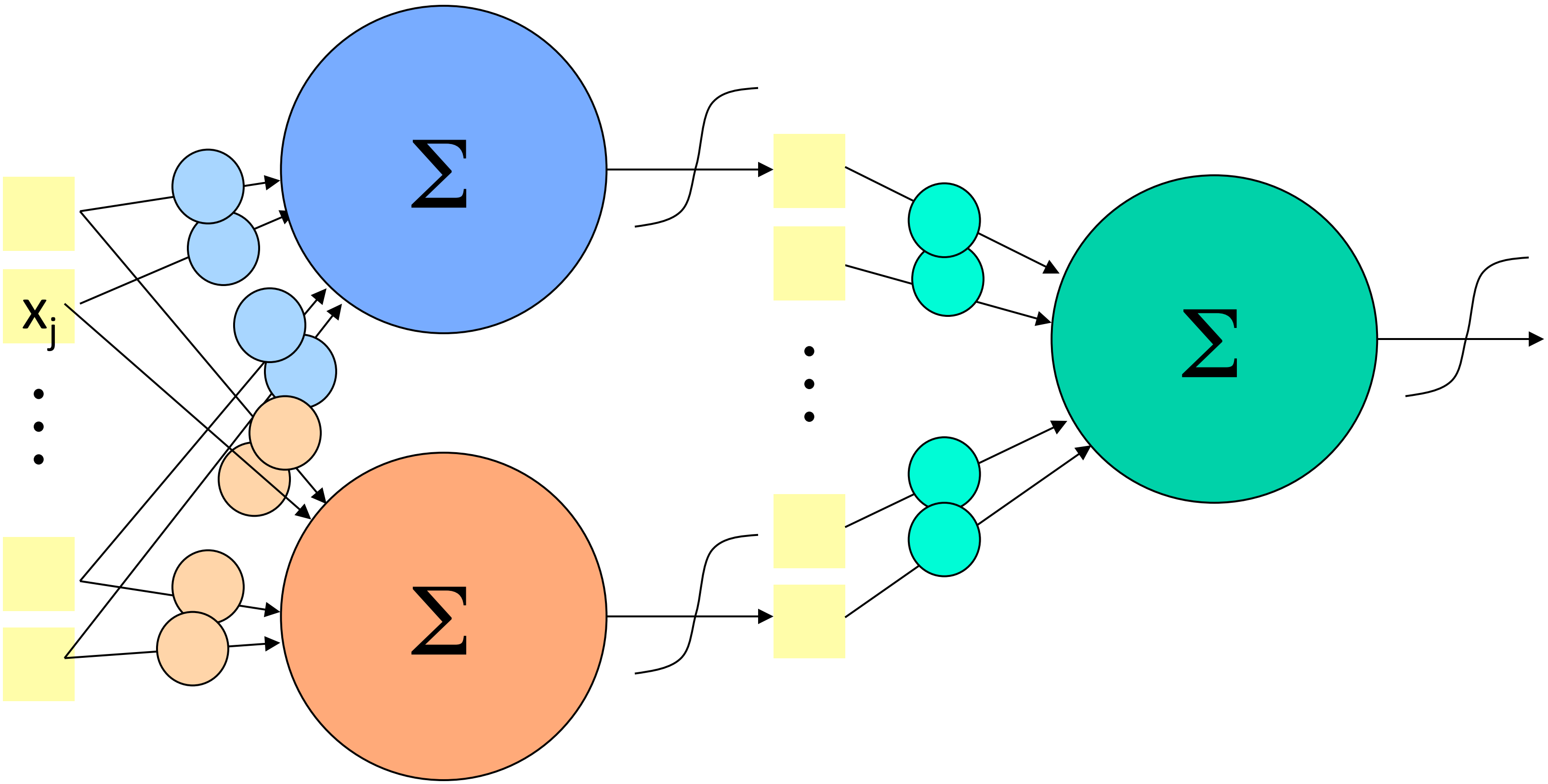
Weight decay

$\gamma \in [0, 1]$, decay parameter

Weight Decay for MLP

Replace: $w_i \leftarrow w_i + \text{back\_prop}(i)$

by: $w_i \leftarrow (1-\gamma)\, w_i + \text{back\_prop}(i)$



Notion of "Risk"

Last time: The risk is the sum of losses

$$R[f] = (1/N) \Sigma_{k=1:N} L(f(\mathbf{x}^k), y^k)$$

- $L(f(\mathbf{x}), y) = \mathbf{1}(f(\mathbf{x})\neq y) = \mathbf{1}(yf(\mathbf{x})<0)$ zero-one loss
- $L(f(\mathbf{x}), y) = (f(\mathbf{x}) - y)^2 = (yf(\mathbf{x}) - 1)^2$ square loss

with $y=\pm 1$

Today: Expected risk

$$R[f] = \int L(f(\mathbf{x}, \mathbf{w}), y) \, dP(\mathbf{x}, y)$$

Empirical risk

$$R_{train}[f] = (1/N) \Sigma_{k=1:N} L(f(\mathbf{x}^k), y^k)$$

Notion of "Risk"

Last time: The risk is the sum of losses

$$R[f] = (1/N) \Sigma_{k=1:N} L(f(\mathbf{x}^k), y^k)$$

- $L(f(\mathbf{x}), y) = \mathbf{1}(f(\mathbf{x}) \neq y)$ zero-one loss
- $L(f(\mathbf{x}), y) = (f(\mathbf{x}) - y)^2$ square loss

with $y=\pm 1$

Today: Expected risk

$$R[f] = \int L(f(\mathbf{x}, \mathbf{w}), y) \, dP(\mathbf{x}, y)$$

Empirical risk

$$R_{train}[f] = (1/N) \Sigma_{k=1:N} L(f(\mathbf{x}^k), y^k)$$

Risk Minimization

- Learning problem: find the best function $f(\mathbf{x};\, \mathbf{w})$ minimizing a risk functional

$$R[f] = \int \underbrace{L(f(\mathbf{x};\, w),\, y)}_{\text{loss function}} \, d\underbrace{P(\mathbf{x},\, y)}_{\text{unknown distribution}}$$

- **Examples are given**:

$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots (\mathbf{x}_N, y_N)$

Approximations of R[f]

(generalization error)

- **Empirical risk**: $R_{train}[f] = (1/N)\ \Sigma_{\mathbf{k=1:N}}\ L(f(\mathbf{x}^k, \mathbf{w}), y^k)$
  - 0/1 loss $\mathbf{1}(f(\mathbf{x}_k) \neq y_k)$: $R_{train}[f]$ = error rate
  - square loss $(f(\mathbf{x}_k)-y_k)^2$: $R_{train}[f]$ = mean square error

- **Guaranteed risk**:

With *high* probability $(1-\delta)$, $R[f] \leq R_{gua}[f]$

$$\boxed{R_{gua}[f] = R_{train}[f] + \varepsilon(\delta, C/N)}$$

Approximations of R[f]

(generalization error)

- **Empirical risk**: $R_{train}[f] = (1/N)\ \Sigma_{\mathbf{k=1:N}}\ L(f(\mathbf{x}^k, \mathbf{w}), y^k)$

  – 0/1 loss $\mathbf{1}(f(\mathbf{x}_k) \neq y_k)$: $R_{train}[f]$ = error rate

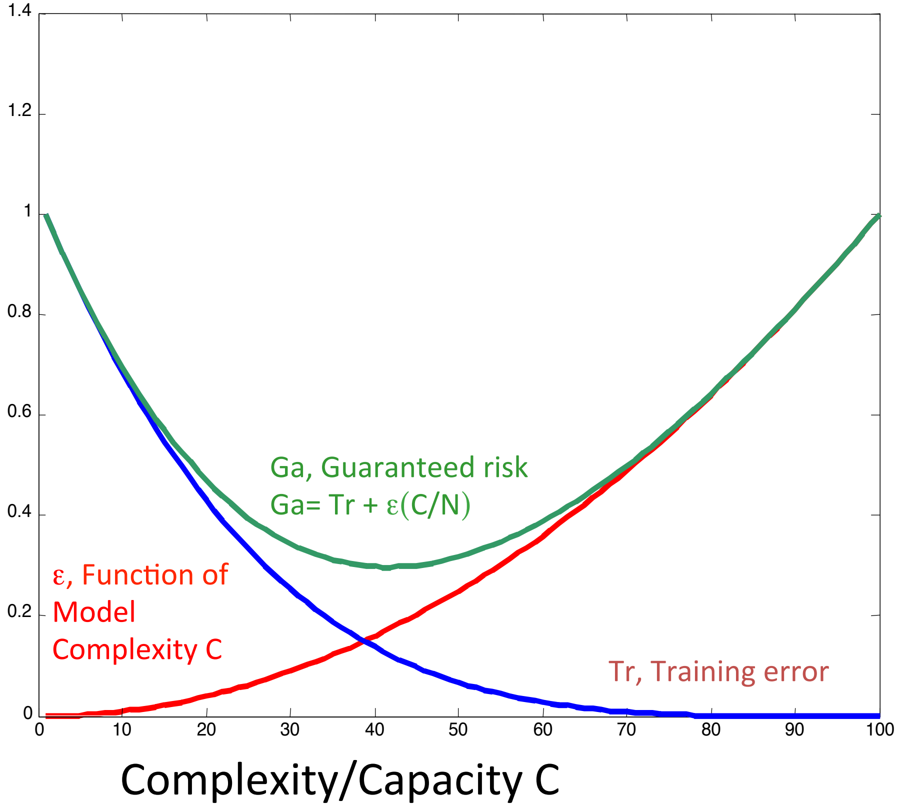
  – square loss $(f(\mathbf{x}_k)-y_k)^2$: $R_{train}[f]$ = mean square error

- **Guaranteed risk**:

  With *high* probability $(1-\delta)$, $R[f] \leq R_{gua}[f]$

$$\boxed{R_{gua}[f] = R_{train}[f] + \varepsilon(\delta, C/N)}$$

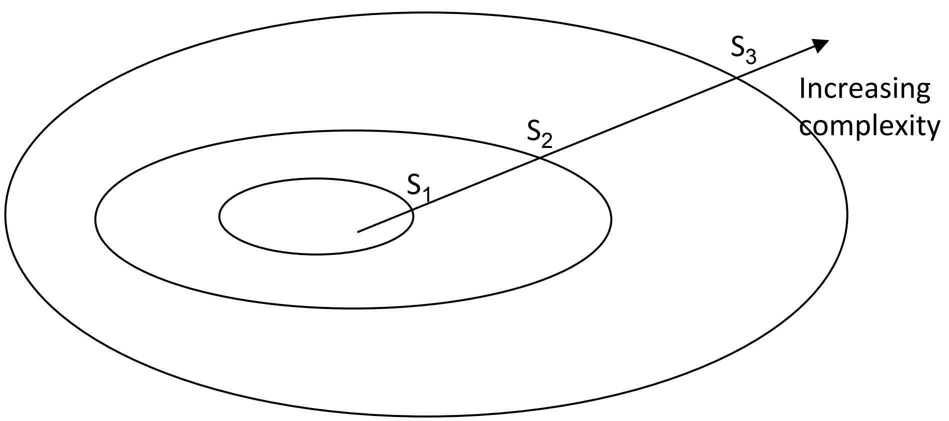
Structural Risk Minimization



Vapnik, 1974

Nested subsets of models, increasing complexity/capacity:

$S_1 \subset S_2 \subset \ldots S_N$



SRM Example (linear model)

$$S_1 \subset S_2 \subset \dots S_N$$

- **Rank with $\|w\|^2 = \sum_i w_i^2$**

  $S_k = \{ \mathbf{w} \mid \|\mathbf{w}\|^2 < \omega_k^2 \}$, $\omega_1 < \omega_2 < \dots < \omega_n$

- **Minimization under constraint:**

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

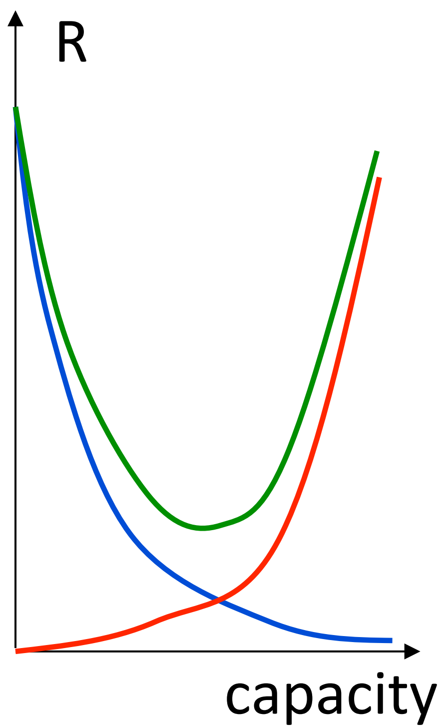
- **Lagrangian:**

  $R_{reg}[f,\lambda] = R_{train}[f] + \lambda \left(\|\mathbf{w}\|^2 - \omega_k^2\right)$, $\lambda > 0$

- **Equivalent problems:**

$\min R_{train}[f]$ s.t. $\|\mathbf{w}\|^2 < \omega_k^2$, $\omega_1 < \omega_2 < \dots < \omega_n$

$\min R_{reg}[f] = R_{train}[f] + \lambda_k \|\mathbf{w}\|^2$, $0 < \lambda_1 < \dots < \lambda_n$



Gradient Descent

$R_{reg}[f] = R_{train}[f] + \lambda \, \|\mathbf{w}\|^2$ **SRM/regularization**

$w_j \leftarrow w_j - \eta \, \partial R_{reg}/\partial w_j$

$w_j \leftarrow w_j - \eta \, \partial R_{train}/\partial w_j - 2 \, \eta \, \lambda \, w_j$ $\gamma = 2 \, \eta \, \lambda$

$w_j \leftarrow (1 - \gamma) \, w_j - \eta \, \partial R_{train}/\partial w_j$ **Weight decay**

Example: **Mean square error:** $(1/N) \underbrace{\Sigma_{k=1:N} (f(\mathbf{x}^k) - y^k)^2}_{\text{RSS}}$

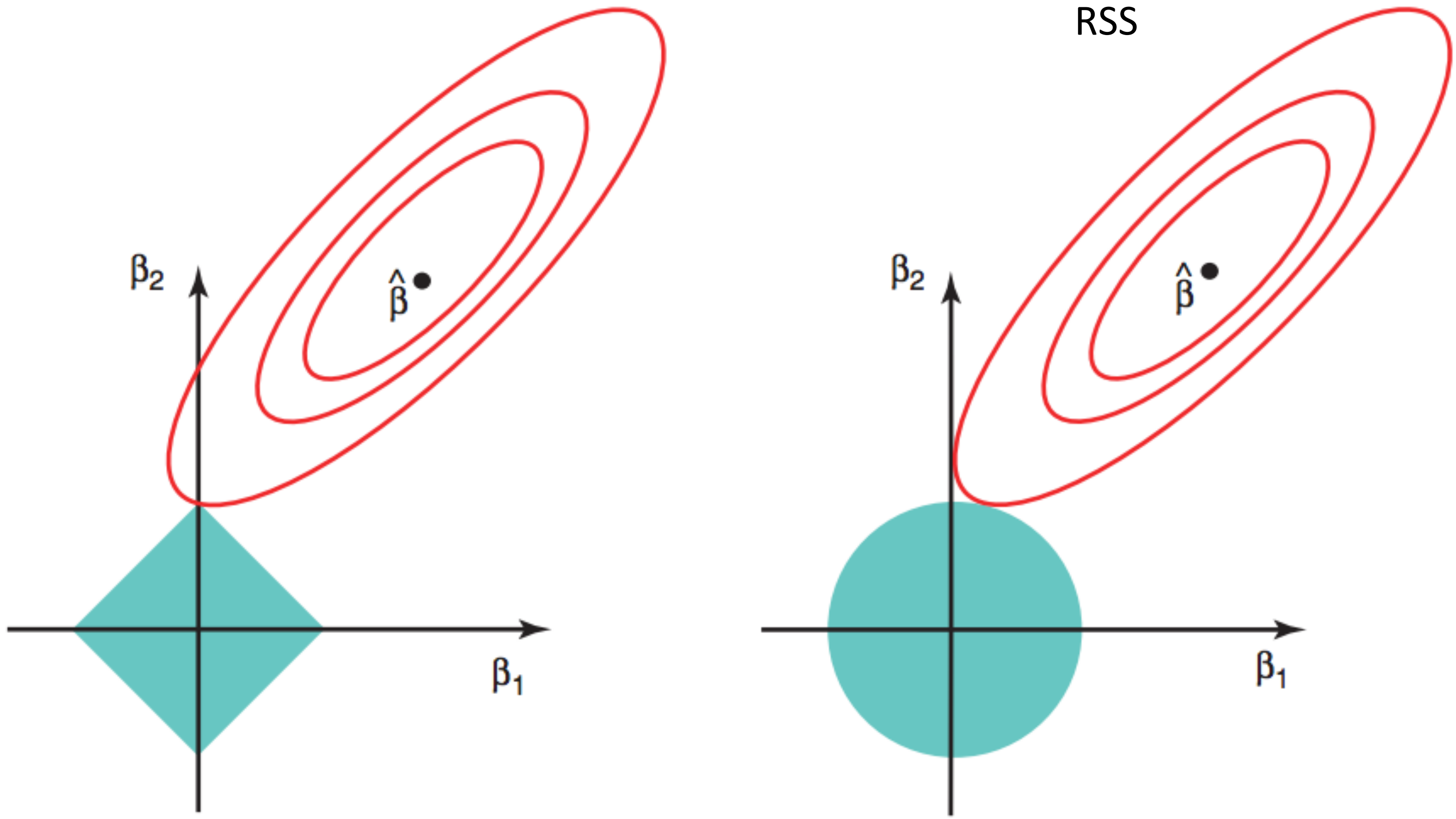
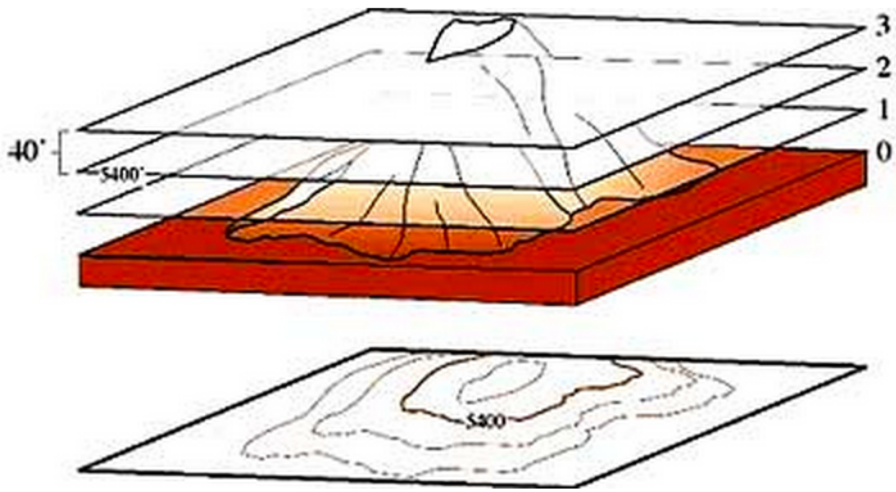


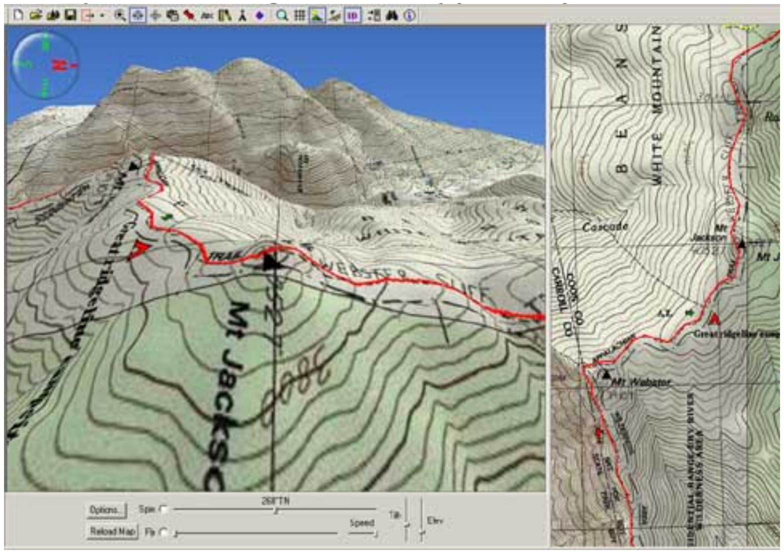
FIGURE 6.7. *Contours of the error and constraint functions for the lasso (left) and ridge regression (right). The solid blue areas are the constraint regions, $|\beta_1| + |\beta_2| \leq s$ and $\beta_1^2 + \beta_2^2 \leq s$, while the red ellipses are the contours of the RSS.*

An Introduction to Statistical Learning: with Applications in R, Gareth James Daniela Witten Trevor Hastie Robert Tibshirani

Contour maps



http://geology.isu.edu/geostac/Field_Exercise/topomaps/topo_map.htm



http://www.nationalgeographic.com/adventure/images/02_06/Appalachian_TOPO_5.jpg

Shrinkage justified

Why do we want "simple" models?

Everything is about mitigating risk

Guaranteed risk:

With *high* probability $(1-\delta)$, $R[f] \leq R_{gua}[f]$

$$R_{gua}[f] = R_{train}[f] + \varepsilon(\delta, C/N)$$

Regularized risk:

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

FIT **ROBUSTNESS**

Multiple Structures

- **Shrinkage (weight decay, ridge regression, SVM):**

  $S_k = \{ \mathbf{w} \mid \|\mathbf{w}\|_2 < \omega_k \}$, $\omega_1 < \omega_2 < \ldots < \omega_k$

  $\gamma_1 > \gamma_2 > \gamma_3 > \ldots > \gamma_k$ *(γ is the ridge)*

- **Feature selection:**

  $S_k = \{ \mathbf{w} \mid \|\mathbf{w}\|_0 < \nu_k \}$,

  $\nu_1 < \nu_2 < \ldots < \nu_k$ *(ν is the number of features)*

- **Kernel parameters** $k(\mathbf{s}, \mathbf{t}) = (\mathbf{s} \bullet \mathbf{t} + 1)^q$ **:**

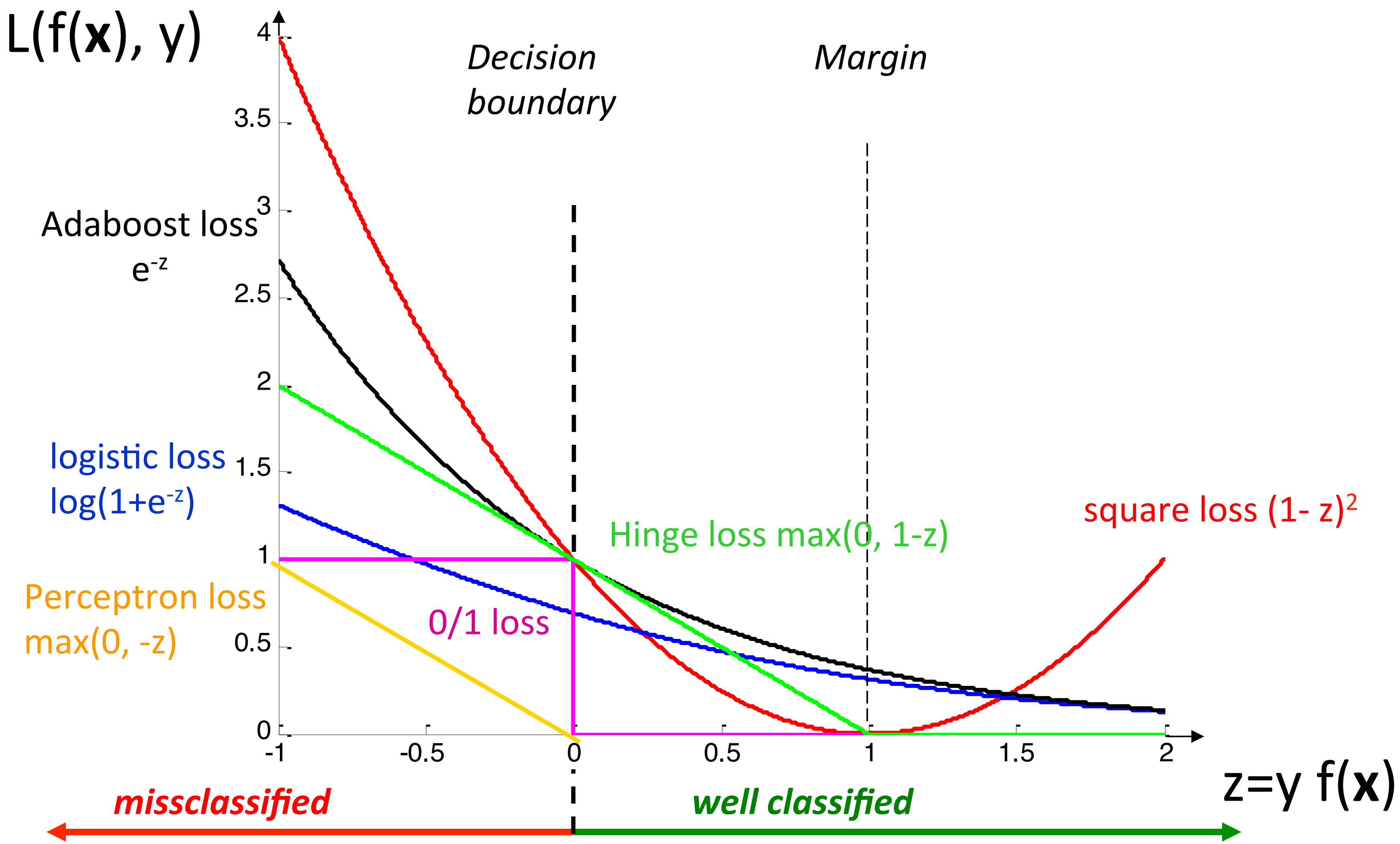
  $q_1 < q_2 < \ldots < q_k$ *(q is the polynomial degree)*

  $k(\mathbf{s}, \mathbf{t}) = \exp(-\|\mathbf{s}-\mathbf{t}\|^2/\sigma^2)$

  $\sigma_1 > \sigma_2 > \sigma_3 > \ldots > \sigma_k$ *(σ is the kernel width)*

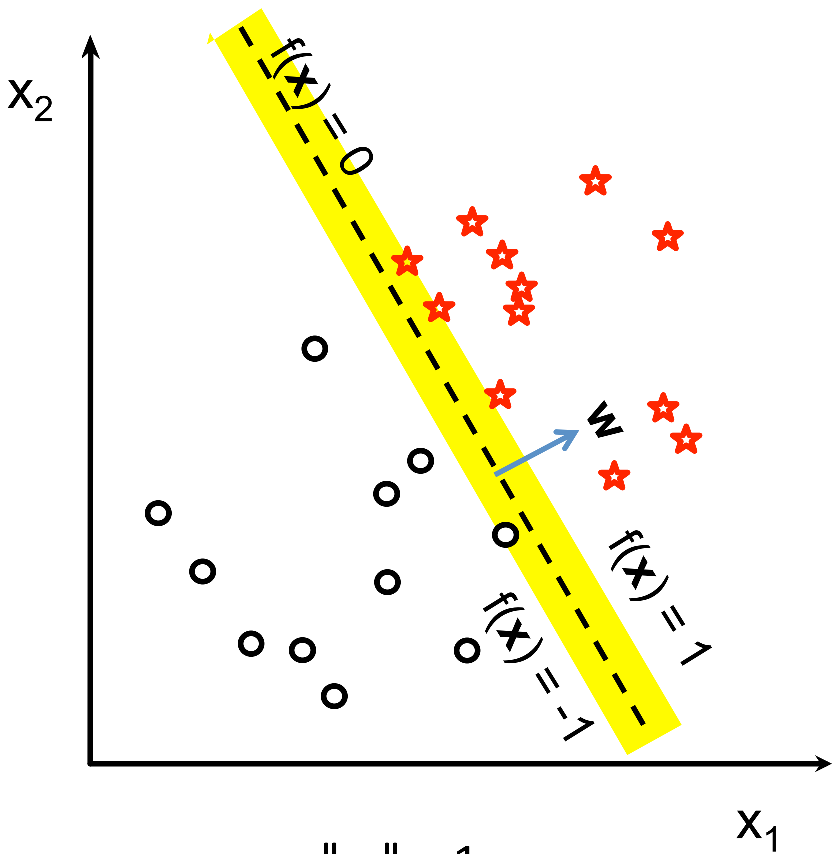
Loss Functions

The risk is the average of the loss.



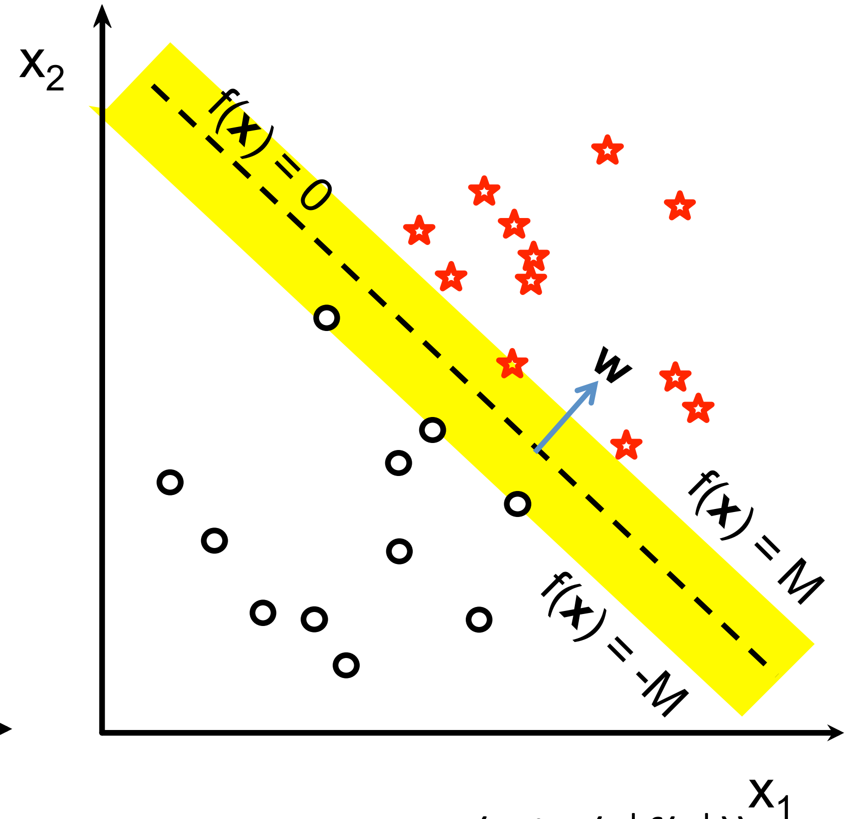
Maximizing the margin

Large margin



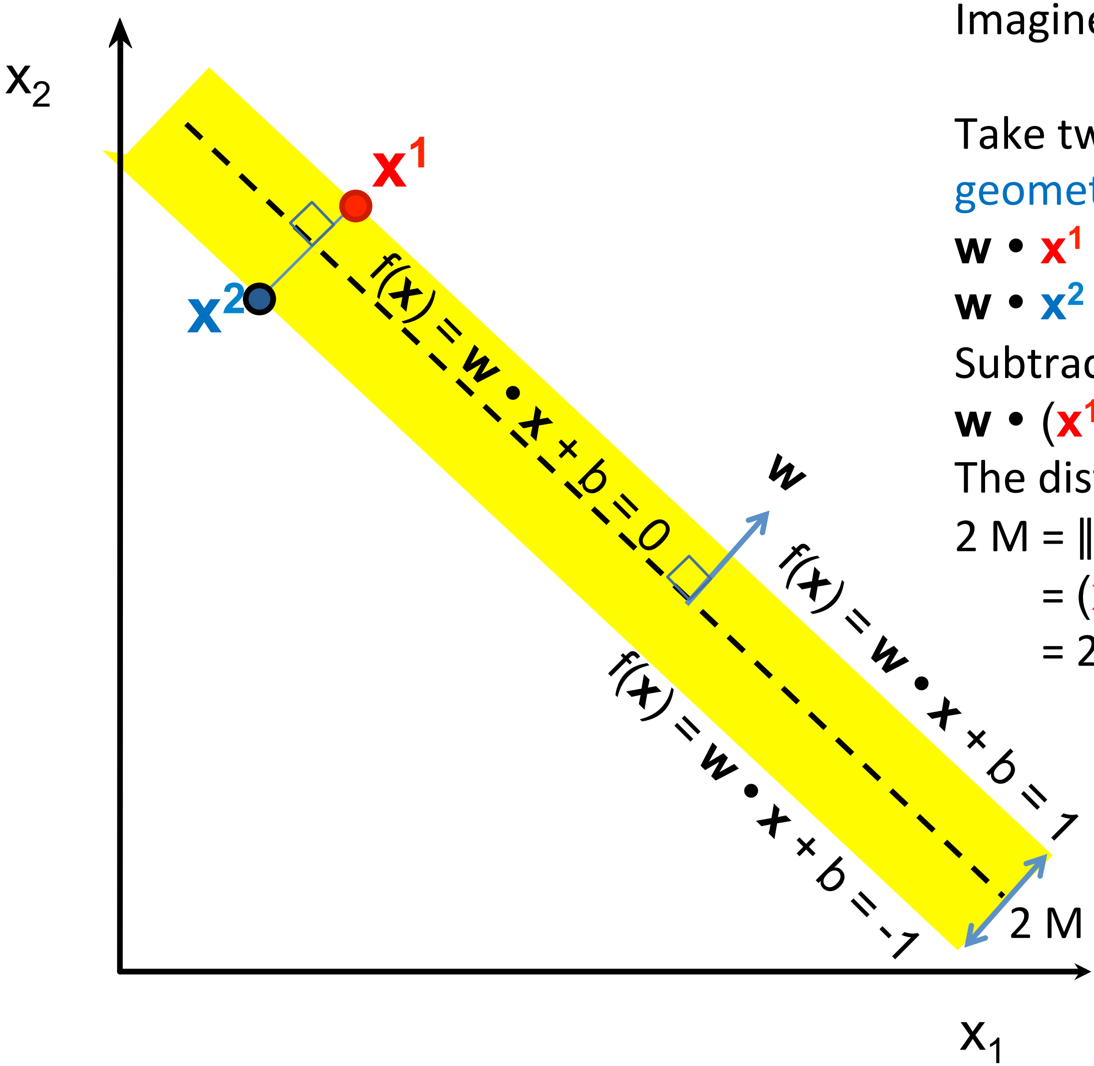
$\|\mathbf{w}\| = 1$

Optimum margin



$M_{opt} = \text{argmax}_w\,(\min_k(y^k f(\mathbf{x}^k)))$

Linking geometrical and functional margin



Imagine that we want to impose $y^k f(\mathbf{x}^k) \geq 1$

$y^k f(\mathbf{x}^k)$ is the functional margin

Take two points on opposite sides of geometrical margin:

$\mathbf{w} \bullet \mathbf{x}^1 + b = 1$

$\mathbf{w} \bullet \mathbf{x}^2 + b = -1$

Subtract:

$\mathbf{w} \bullet (\mathbf{x}^1 - \mathbf{x}^2) = 2$

The distance between $\mathbf{x}^1$ and $\mathbf{x}^2$ is:

$$2\,M = \| \mathbf{x}^1 - \mathbf{x}^2 \| = (\mathbf{x}^1 - \mathbf{x}^2) \bullet \mathbf{w}/\|\mathbf{w}\| = 2 / \|\mathbf{w}\|$$

$$\boxed{M = 1 / \|\mathbf{w}\|}$$

Equivalent formulations

$\|\mathbf{w}\| = 1$

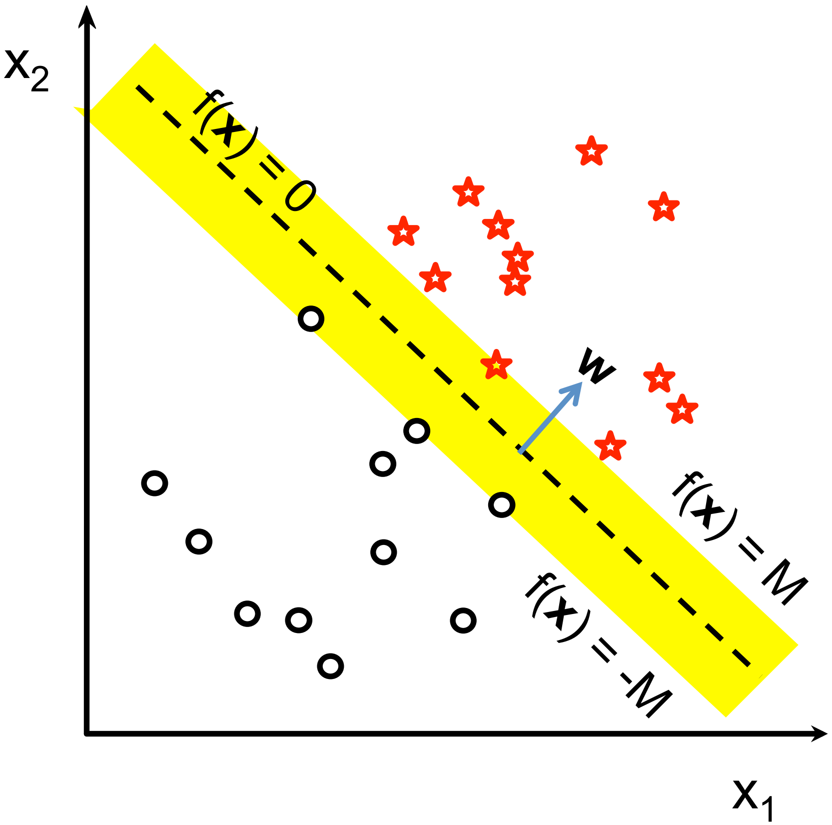
$M_{opt} = \text{argmax}_w (\min_k(y^k f(\mathbf{x}^k))$

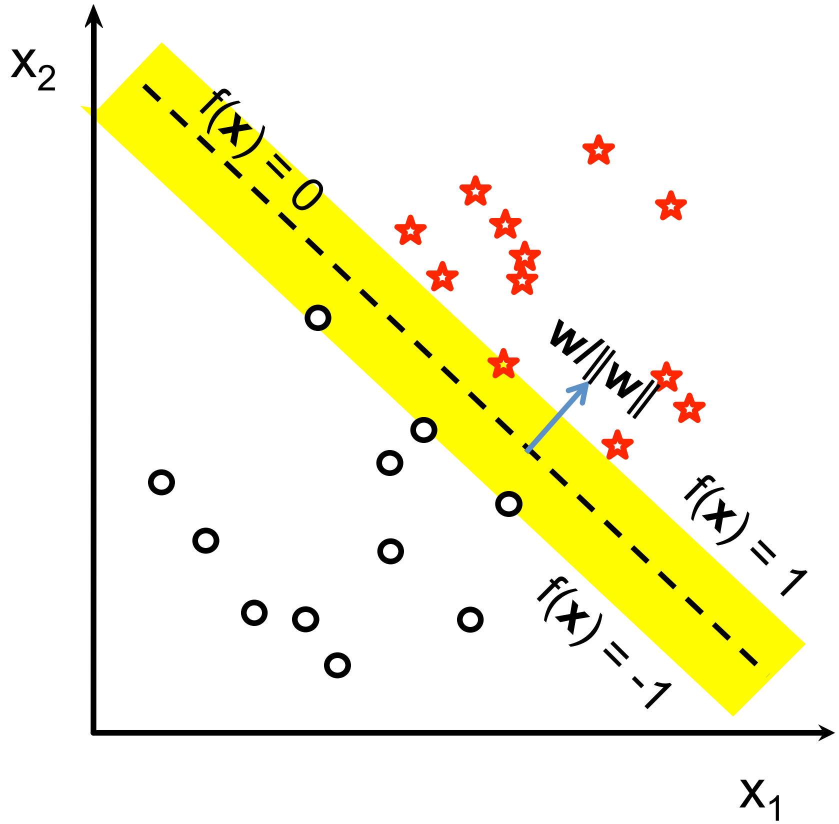
$\Leftrightarrow$

$M = 1/\ \|\mathbf{w}\|$

$M_{opt} = \max\ (1/\ \|\mathbf{w}\|)$

s.t. $\min_k(y^k f(\mathbf{x}^k)) = 1$





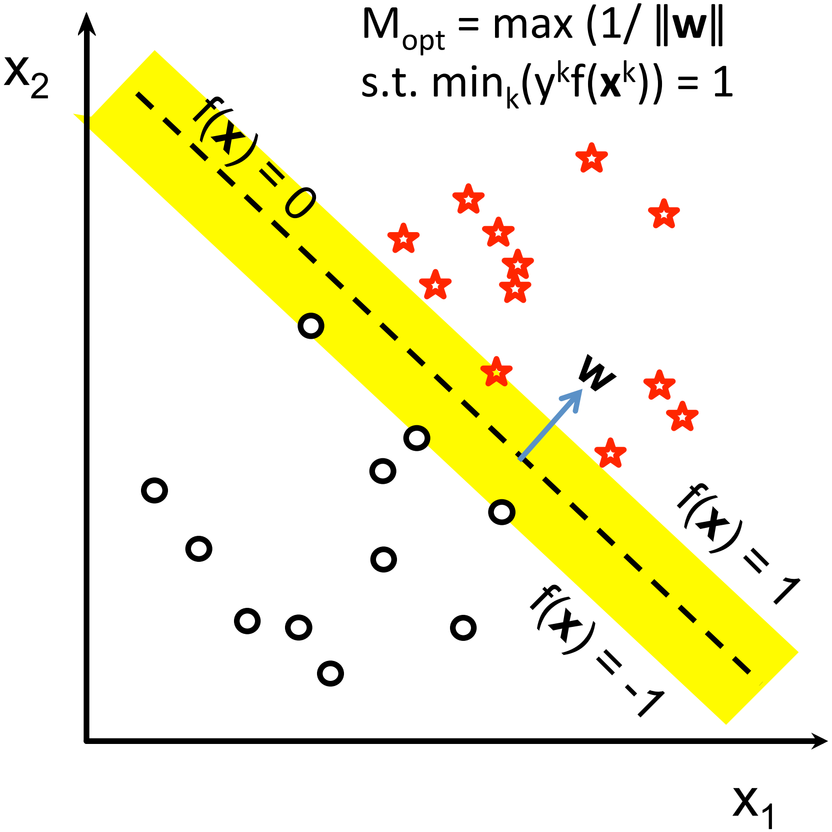
Optimum margin

Hard margin

$M = 1/\ \|\mathbf{w}\|$

$M_{opt} = \max\ (1/\ \|\mathbf{w}\|$

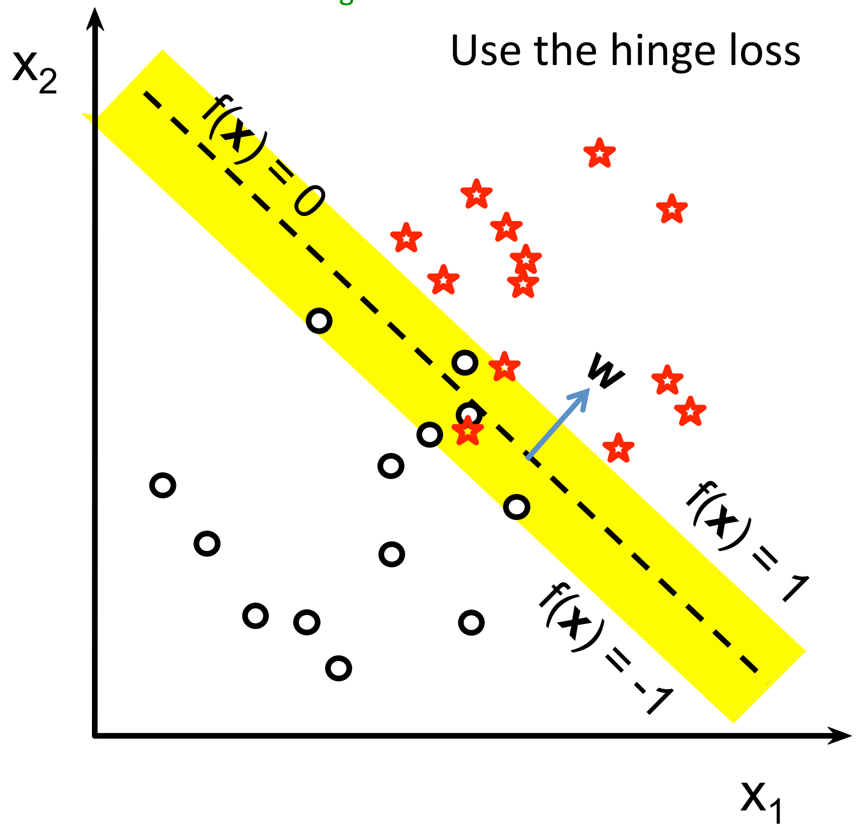
s.t. $\min_k(y^k f(\mathbf{x}^k)) = 1$



Soft margin

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

Use the hinge loss

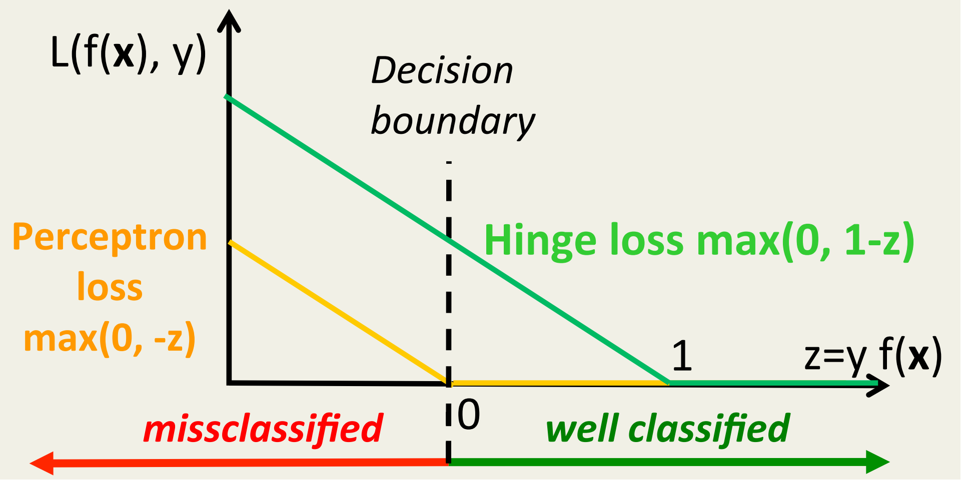


Large margin Perceptron with weight decay = soft margin

$L_{hinge} = \max(0, 1 - z)$

$z = y\, f(\mathbf{x})$

$\min_w \,(L_{hinge} + \lambda \,\|\mathbf{w}\|^2)$



$$w_i \leftarrow \begin{cases} (1-\gamma)\, w_i + \eta\, y\, x_i, & \text{if } z < 1 \text{ (misclassified or within margin)} \\ (1-\gamma)\, w_i & \text{otherwise} \end{cases} \qquad (\gamma = 2\,\eta\,\lambda)$$

Soft Margin **C**ompromise

Minimize

(1/Margin) + C Training error

Good robustness

Good fit

Soft Margin Compromise

Minimize

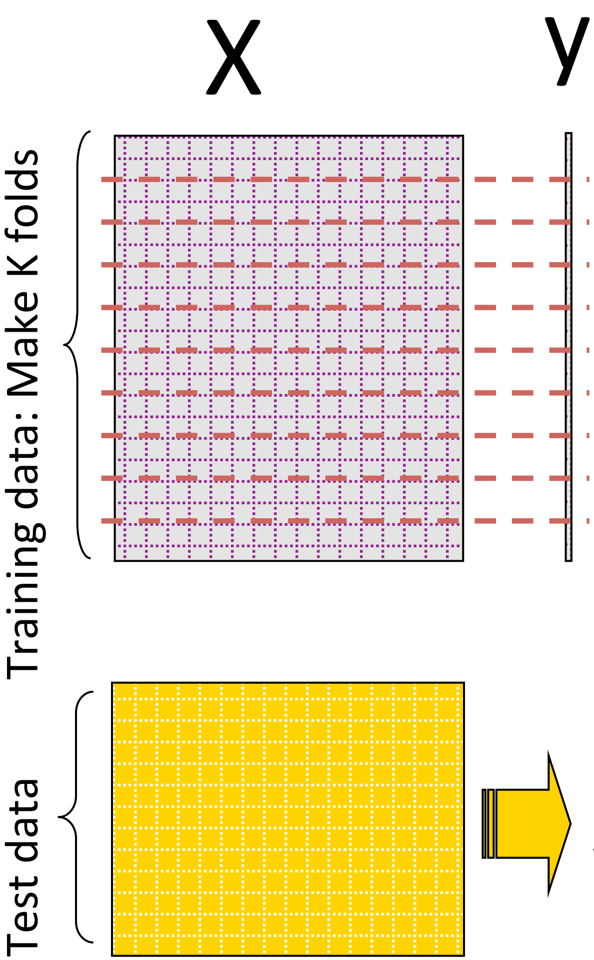
$\lambda \, \|\mathbf{w}\|^2$ **+** $R_{train}[f]$

$\Sigma_k \max(0, 1-z^k)$

Good robustness

Good fit

Hyper-parameter Selection



Prospective study / “real” validation

- **Learning = adjusting:**

  parameters ($\mathbf{w}$ vector).

  hyper-parameters (γ, ν, q, σ).

- ***Cross-validation* with K-folds:**

For various values of γ, ν, q, σ:

- Adjust $\mathbf{w}$ on a fraction (K-1)/K of training examples *e.g.* 9/10th.
- Test on 1/K remaining examples *e.g.* 1/10th.
- Rotate examples and average test results (CV error).
- Select γ, ν, q, σ to minimize CV error.
- Re-compute $\mathbf{w}$ on **all** training examples using optimal γ, ν, q, σ.

Summary

- High complexity models may "**overfit**":
  - Fit perfectly training examples
  - Generalize poorly to new cases
- **SRM solution**: organize the models in nested subsets such that in every structure element

  complexity < θ

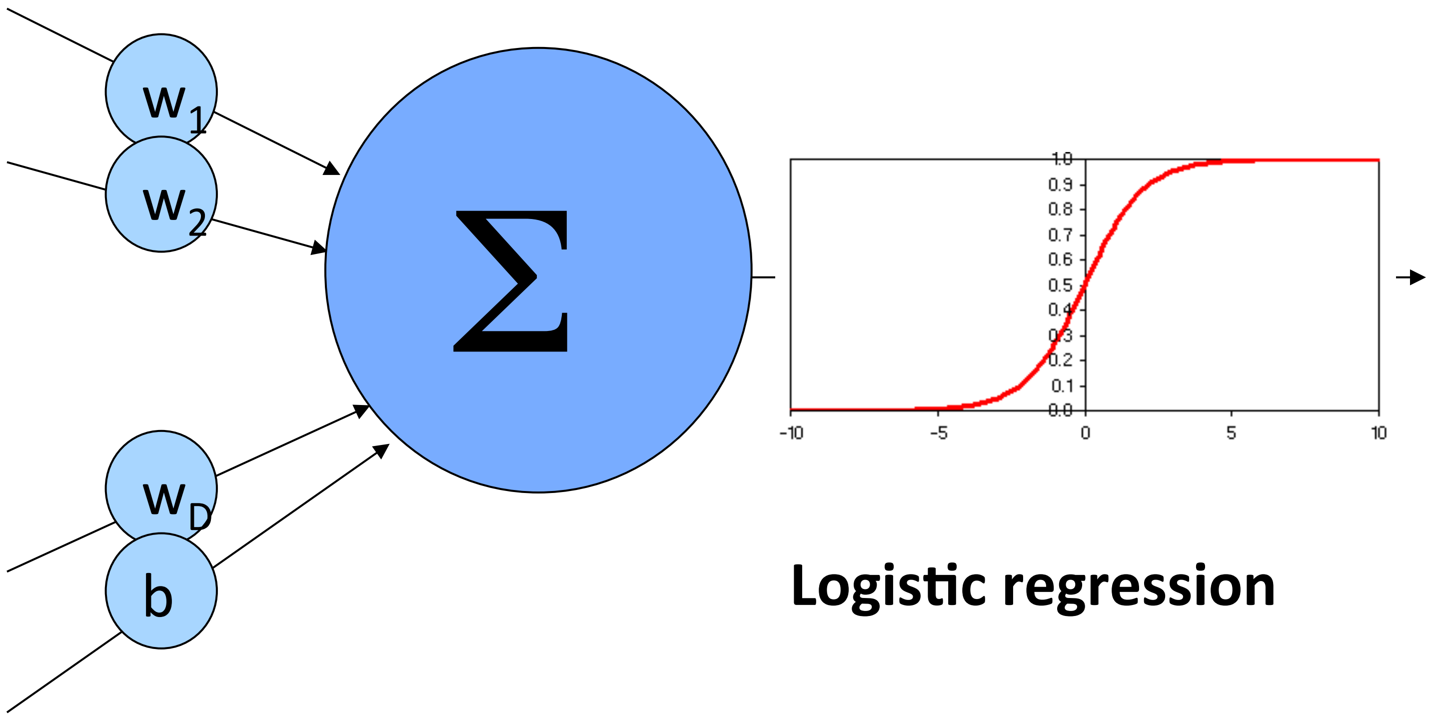
- **Regularization:** Formalize learning as a constrained optimization problem, minimize

  regularized risk = training error + λ complexity

- Both formulations are equivalent via the use of Lagrange multipliers.
- θ and λ are hyperparamenters, which can be optimized by cross-validation.

Come to my office hours…
Wed 2:30-4:30 Soda 329

Next time



Logistic regression