Applied Machine Learning: Algorithms, Practice and Theory

Online Learning

Koby Crammer
Technion – Israel Institute of Technology

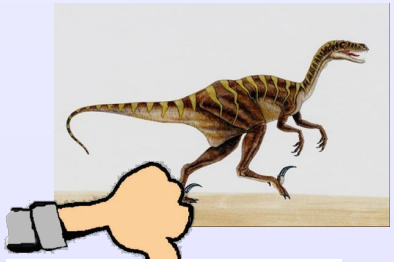
Online Learning

Tyrannosaurus rex

Online Learning

Triceratops

Online Learning



Vel

Tyrannosaurus rex

Formal Setting – Binary Classification

- Instances
  - Images, Sentences
- Labels
  - Parse tree, Names
- Prediction rule
  - Linear predictions rules
- Loss
  - No. of mistakes

$$\mathbf{x} \in \mathcal{X}$$

$$y \in \mathcal{Y} = \{-1 \; ; \; 1\}$$

$$f(\mathbf{x}) = \hat{y}$$

$$\ell(\hat{y}, y) \in \mathbb{R}_+$$

Online Framework

- Initialize Classifier $f_1(\mathbf{x})$
- Algorithm works in rounds $t = 1 \dots T \dots$
- On round t the online algorithm :
  - Receives an input instance $\mathbf{x}_t$
  - Outputs a prediction $f_t(\mathbf{x}_t) = \hat{y}_t$
  - Receives a feedback label y_t
  - Computes loss $\ell(\hat{y}_t, y_t)$
  - Updates the prediction rule $f_t \rightarrow f_{t+1}$

- Goal :
  - Suffer small cumulative loss $\sum_t \ell(\hat{y}_t, y_t)$

Why Online Learning?

- Fast
- Memory efficient - process one example at a time
- Simple to implement
- Formal guarantees – Mistake bounds
- Online to Batch conversions
- No statistical assumptions
- Adaptive

- Not as good as a well designed batch algorithms

Update Rules

- Online algorithms are based on an update rule which defines f_{t+1} from f_t (and possibly other information)
- Linear Classifiers : find $\mathbf{w}_{t+1}$ from $\mathbf{w}_t$ based on the input $(\mathbf{x}_t, y_t)$
- Some Update Rules :
  - Perceptron (Rosenblat)
  - ALMA (Gentile)
  - ROMMA (Li & Long)
  - NORMA (Kivinen et. al)
  - MIRA (Crammer & Singer)
  - EG (Littlestown and Warmuth)
  - Bregman Based (Warmuth)

Three Update Rules

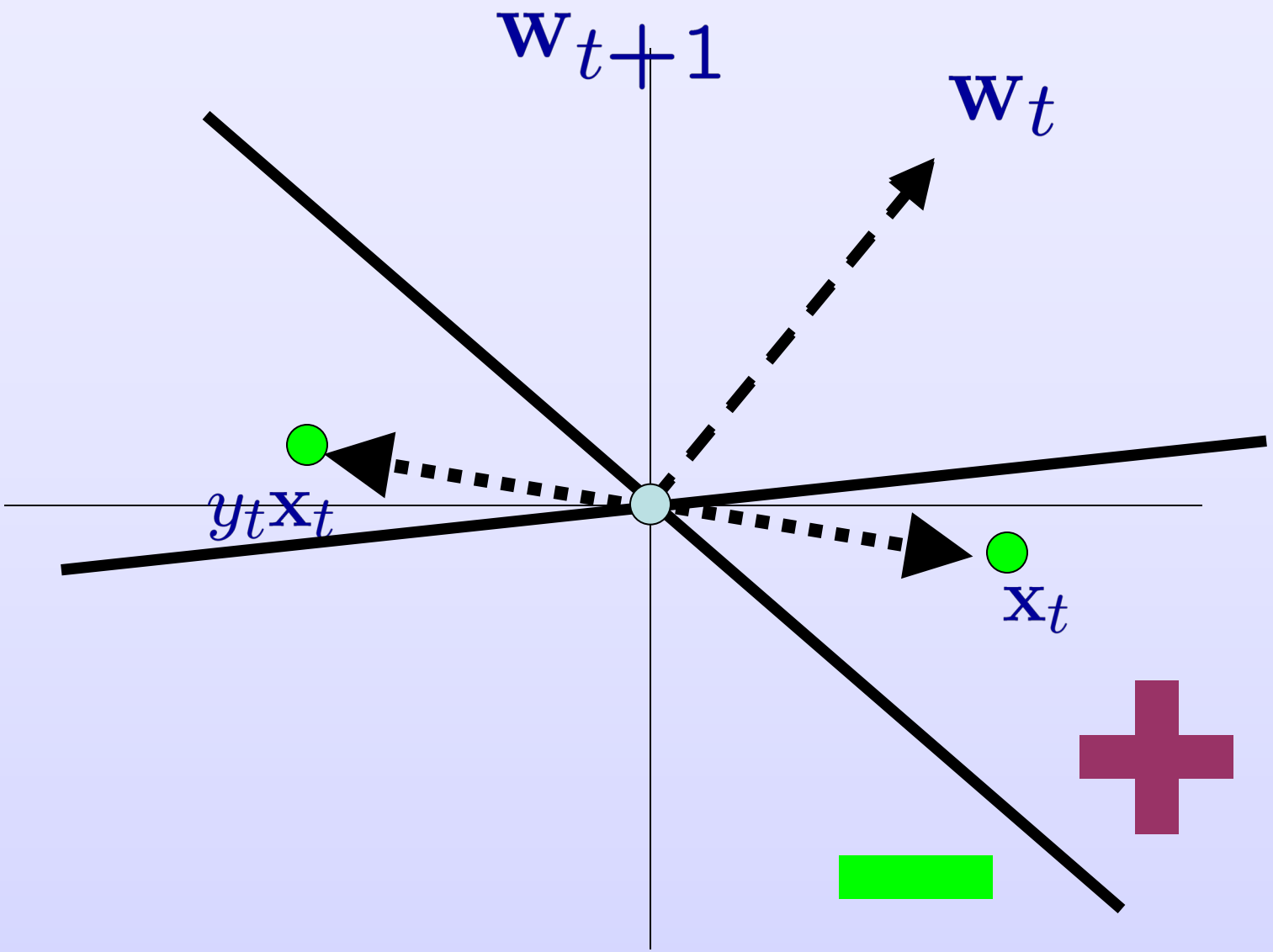
- The Perceptron Algorithm :
  - Agmon 1954; Rosenblatt 1952-1962, Block 1962, Novikoff 1962, Minsky & Papert 1969, Freund & Schapire 1999, Blum & Dunagan 2002
- Hildreth's Algorithm :
  - Hildreth 1957
  - Censor & Zenios 1997
  - Herbster 2002
- Loss Scaled :
  - Crammer & Singer 2001,
  - Crammer & Singer 2002

The Perceptron Algorithm

- If No-Mistake $y_t(\mathbf{w}_t \cdot \mathbf{x}_t) > 0$
  - Do nothing $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t$
- If Mistake $y_t(\mathbf{w}_t \cdot \mathbf{x}_t) \leq 0$
  - Update $\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + y_t \mathbf{x}_t$
- Margin after update :

$$y_t(\mathbf{w}_{t+1} \cdot \mathbf{x}_t) \geq y_t(\mathbf{w}_t \cdot \mathbf{x}_t) + \|\mathbf{x}_t\|^2 (\mathbf{x}_t \cdot \mathbf{x}_t)$$

Geometrical Interpretation



Relative Loss Bound

- For any competitor prediction function f^*
- We bound the loss suffered by the algorithm with the loss suffered by f^*

$$\sum_{t=1}^{T} \ell(f_t(\mathbf{x}_t), y_t) \leq \mathcal{R} + C \sum_{t=1}^{T} \ell(f^*(\mathbf{x}_t), y_t)$$

Cumulative Loss Suffered by the Algorithm

Sequence of Prediction Functions

Cumulative Loss of Competitor

Relative Loss Bound

- For any competitor prediction function
- We bound the loss suffered by the algorithm with the loss suffered by f^*

$$\sum_{t=1}^{T} \ell(f_t(\mathbf{x}_t), y_t) \leq \mathcal{R} + \mathcal{C} \sum_{t=1}^{T} \ell(f^*(\mathbf{x}_t), y_t)$$

Inequality
Possibly Large Gap

Regret
Extra Loss

Competitiveness
Ratio

Relative Loss Bound

- For any competitor prediction function
- We bound the loss suffered by the algorithm with the loss suffered by f^*

$$\sum_{t=1}^{T} \ell(f_t(\mathbf{x}_t), y_t) \leq \mathcal{R} + C \sum_{t=1}^{T} \ell(f^*(\mathbf{x}_t), y_t)$$

Grows With T

Constant

Relative Loss Bound

- For any competitor prediction function
- We bound the loss suffered by the algorithm with the loss suffered by f^*

$$\sum_{t=1}^{T} \ell(f_t(\mathbf{x}_t), y_t) \leq \mathcal{R} + \mathcal{C} \min_{f} \left[\sum_{t=1}^{T} \ell(f(\mathbf{x}_t), y_t) \right]$$

Best Prediction Function in hindsight for the data sequence

Remarks

- If the input is inseparable, then the problem of finding a separating hyperplane which attains less then M errors is NP-hard (Open hemisphere)
- Obtaining a zero-one loss bound with a unit competitiveness ratio is as hard as finding a constant approximating error for the Open Hemesphere problem.
- Bound of the *number of mistakes* the perceptron makes with the *hinge loss* of any compitetor

$$\sum_{t=1}^{T} \ell_{01}(f_t(\mathbf{x}_t), y_t) \leq \mathcal{R} + C \sum_{t=1}^{T} \ell_{\text{hinge}}(f(\mathbf{x}_t), y_t)$$

Definitions

- Any Competitor $f^*(\mathbf{x}) = \mathbf{u} \cdot \mathbf{x}$
- The parameters vector $\mathbf{u}$ can be chosen using the input data

$$(\mathbf{x}_1, y_1) \ , \ldots , \ (\mathbf{x}_T, y_T)$$

- The *parameterized* hinge loss of $f^*(\mathbf{x})$ on $(\mathbf{x}_t, y_t)$

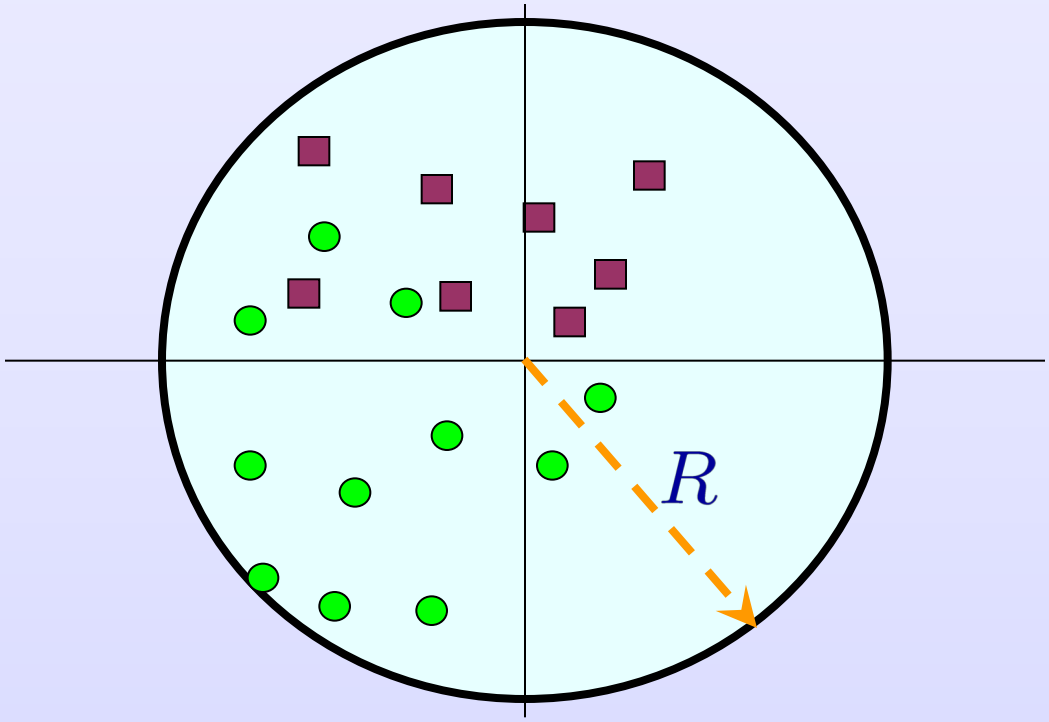
$$\ell^*_\gamma(\mathbf{x}_t, y_t) = \max\{0, \gamma - y_t(\mathbf{u} \cdot \mathbf{x})\}$$

- True hinge loss $\gamma = 1$
- 1-norm and 2-norm of hinge loss

$$\mathcal{L}_1 = \sum_t \ell^*(\mathbf{x}_t, y_t) \qquad \mathcal{L}_2 = \sqrt{\sum_t \left(\ell^*(\mathbf{x}_t, y_t)\right)^2}$$

Geometrical Assumption

- All examples are bounded in a ball of radius R



$$\|\mathbf{x}_t\|^2 \leq R^2 \quad \forall t$$

Perceptron's Mistake Bound

- Bounds : $M \leq (R\|\mathbf{u}\| + \mathcal{L}_2)^2$

$$M \leq \left(R\|\mathbf{u}\| + \sqrt{\mathcal{L}_1}\right)^2$$

$$M \leq R^2\|\mathbf{u}\|^2 + 2\mathcal{L}_1$$

- If the sample is separable then

$$M \leq R^2\|\mathbf{u}\|^2$$

[FS99, SS05]

Proof - Intuition

- Two views :
  - The angle between $\mathbf{u}$ and $\mathbf{w}_t$ decreases with t

$$\frac{\mathbf{u} \cdot \mathbf{w}_t}{\|\mathbf{w}_t\| \, \|\mathbf{u}\|}$$

  - The following sum is fixed

$$M\, R^2 - 2 \sum_t^T \ell^*(\mathbf{x}_t, y_t) + \|\mathbf{w}_{T+1} - \mathbf{u}\|^2$$

as we make more mistakes, our solution is better

Proof

- Define the potential :

$$\Delta_t = \|\mathbf{u} - \mathbf{w}_t\|^2 - \|\mathbf{u} - \mathbf{w}_{t+1}\|^2$$

- Bound it's cumulative sum $\sum_t \Delta_t$ from above and below

Proof

- Bound from above :

$$\begin{aligned}\sum_t \Delta_t &= \sum_t \left[\|\mathbf{u}-\mathbf{w}_t\|^2 - \|\mathbf{u}-\mathbf{w}_{t+1}\|^2\right] \\ &= \|\mathbf{u}-\mathbf{w}_1\|^2 - \|\mathbf{u}-\mathbf{w}_{T+1}\|^2 \\ &\leq \|\mathbf{u}\|^2\end{aligned}$$

Telescopic Sum

Zero Vector

Non-Negative

Proof

- Bound From Below :
  - No error on tth round

$$\begin{aligned}\Delta_t &= \|\mathbf{u}-\mathbf{w}_t\|^2 - \|\mathbf{u}-\mathbf{w}_{t+1}\|^2 \\ &= 0 \geq -2\ell^*_\gamma(\mathbf{x}_t, y_t)\end{aligned}$$

  - Error on tth round

$$\begin{aligned}\Delta_t &= \|\mathbf{u}-\mathbf{w}_t\|^2 - \|\mathbf{u}-\mathbf{w}_t - y_t\mathbf{x}_t\|^2 \\ &= -\|\mathbf{x}_t\|^2 + 2y_t(\mathbf{u}\cdot\mathbf{x}_t) - 2y_t(\mathbf{w}_t\cdot\mathbf{x}_t)\end{aligned}$$

Proof

- We bound each term :

$$\Delta_t = -\|\mathbf{x}_t\|^2 + 2y_t(\mathbf{u}\cdot\mathbf{x}_t) - 2y_t(\mathbf{w}_t\cdot\mathbf{x}_t)$$

$-\|\mathbf{x}_t\|^2 \geq -R^2$

$-2y_t(\mathbf{w}_t\cdot\mathbf{x}_t) \geq 0$

$$\begin{aligned} y_t(\mathbf{u}\cdot\mathbf{x}_t) &\geq \gamma - \max\{0, \gamma - y_t(\mathbf{u}\cdot\mathbf{x}_t)\} \\ &= \gamma - \ell^*_\gamma(\mathbf{x}_t, y_t) \end{aligned}$$

Proof

- Bound From Below :
  - No error on t[th] round

$$\Delta_t \geq -2\ell^*_\gamma(\mathbf{x}_t, y_t)$$

  - Error on t[th] round

$$\Delta_t \geq -2\ell^*_\gamma(\mathbf{x}_t, y_t) - R^2 + 2\gamma$$

- Cumulative bound :

$$\sum_t \Delta_t \geq -2\sum_t \ell^*_\gamma(\mathbf{x}_t, y_t) + M(2\gamma - R^2)$$

Proof

- Putting both bounds together :

$$-2\sum_t \ell^*_\gamma(\mathbf{x}_t, y_t) + M(2\gamma - R^2) \leq \sum_t \Delta_t \leq \|\mathbf{u}\|^2$$

- We use first degree of freedom (and scale) :

$$\mathbf{u} \leftarrow c\mathbf{u} \qquad \gamma \leftarrow c\gamma \qquad \ell^*_\gamma(\mathbf{x}_t, y_t) \leftarrow c\ell^*_\gamma(\mathbf{x}_t, y_t)$$

- Bound :

$$-2c\sum_t \ell^*_\gamma(\mathbf{x}_t, y_t) + M(2c\gamma - R^2) \leq c^2\|\mathbf{u}\|^2$$

Proof

- General Bound :

$$M \leq \frac{c^2\|\mathbf{u}\|^2 + 2c\sum_t \ell_\gamma^*(\mathbf{x}_t, y_t)}{2c\gamma - R^2}$$

- Choose :

$$c = \frac{R^2}{\gamma} \quad , \quad \gamma = 1$$

Objective of SVM

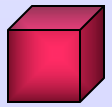
- Simple Bound :

$$M \leq R^2\|\mathbf{u}\|^2 + 2\sum_t \ell^*(\mathbf{x}_t, y_t)$$

Proof

- Better bound : optimize the value of

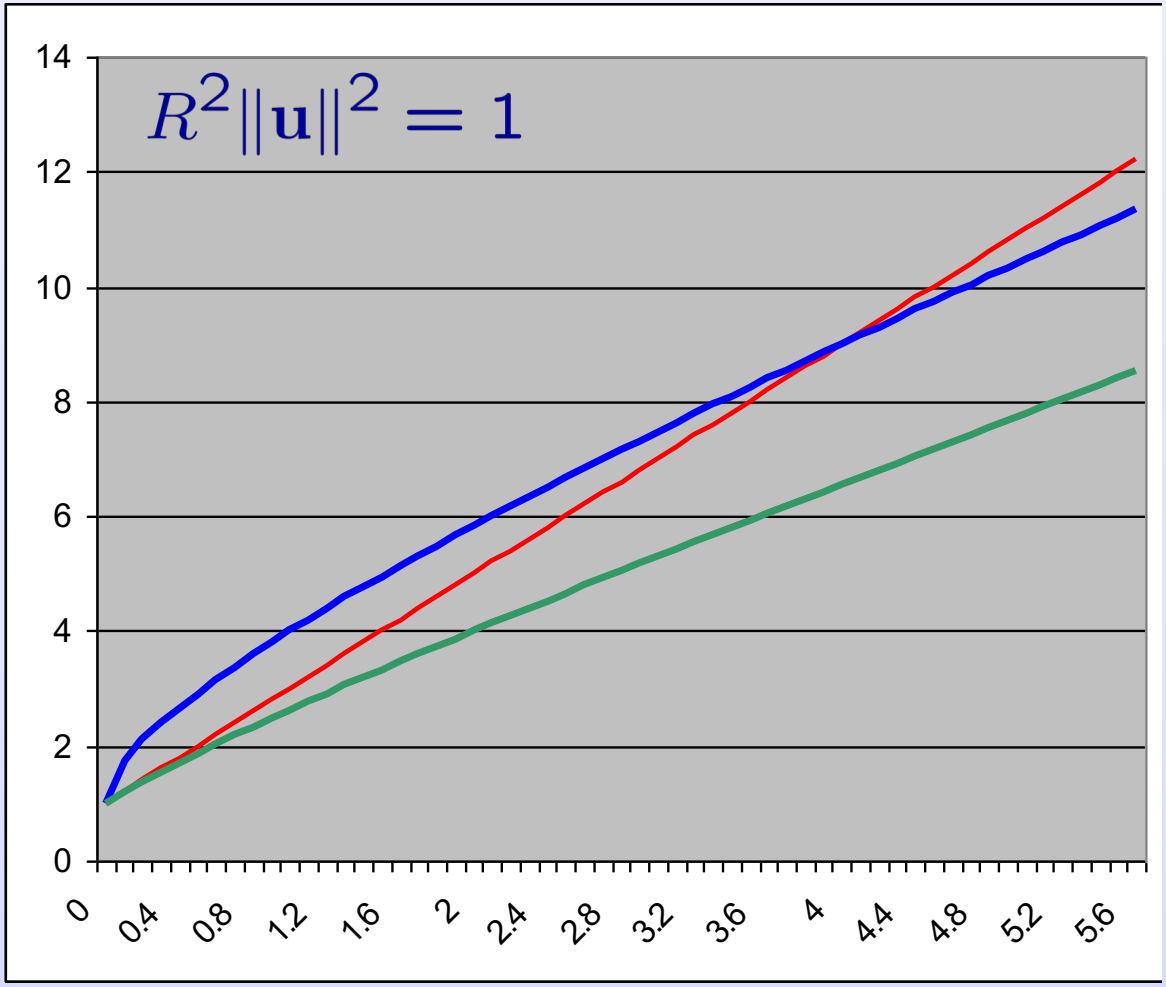
$$M \leq \min_{c} \left\{ \frac{c^2 \|\mathbf{u}\|^2 + 2c \sum_t \ell_\gamma^*(\mathbf{x}_t, y_t)}{2c\gamma - R^2} \right\}$$



Remarks

- Bound does not depend on dimension of the feature vector
- The bound holds for *all* sequences. It is not tight for most real world data
- But, there exists a setting for which it is tight – worst

Three Bounds



$$M \leq R^2\|\mathbf{u}\|^2 + 2\mathcal{L}_1$$

$$M \leq \left(R\|\mathbf{u}\| + \sqrt{\mathcal{L}_1}\right)^2$$

$$M \leq \frac{1}{4}\left(R\|\mathbf{u}\| + \sqrt{\|\mathbf{u}\|^2R^2 + 4\mathcal{L}_1}\right)^2$$

Separable Case

- Assume there exists $\mathbf{u}$ such that $y_t(\mathbf{u} \cdot \mathbf{x}_t) > 0$ for all examples $t = 1 \ldots T$
  Then all bounds are equivalent

$$M \leq R^2 \|\mathbf{u}\|^2$$

- Perceptron makes finite number of mistakes until convergence (not necessarily to $\mathbf{u}$)

Separable Case – Other Quantities

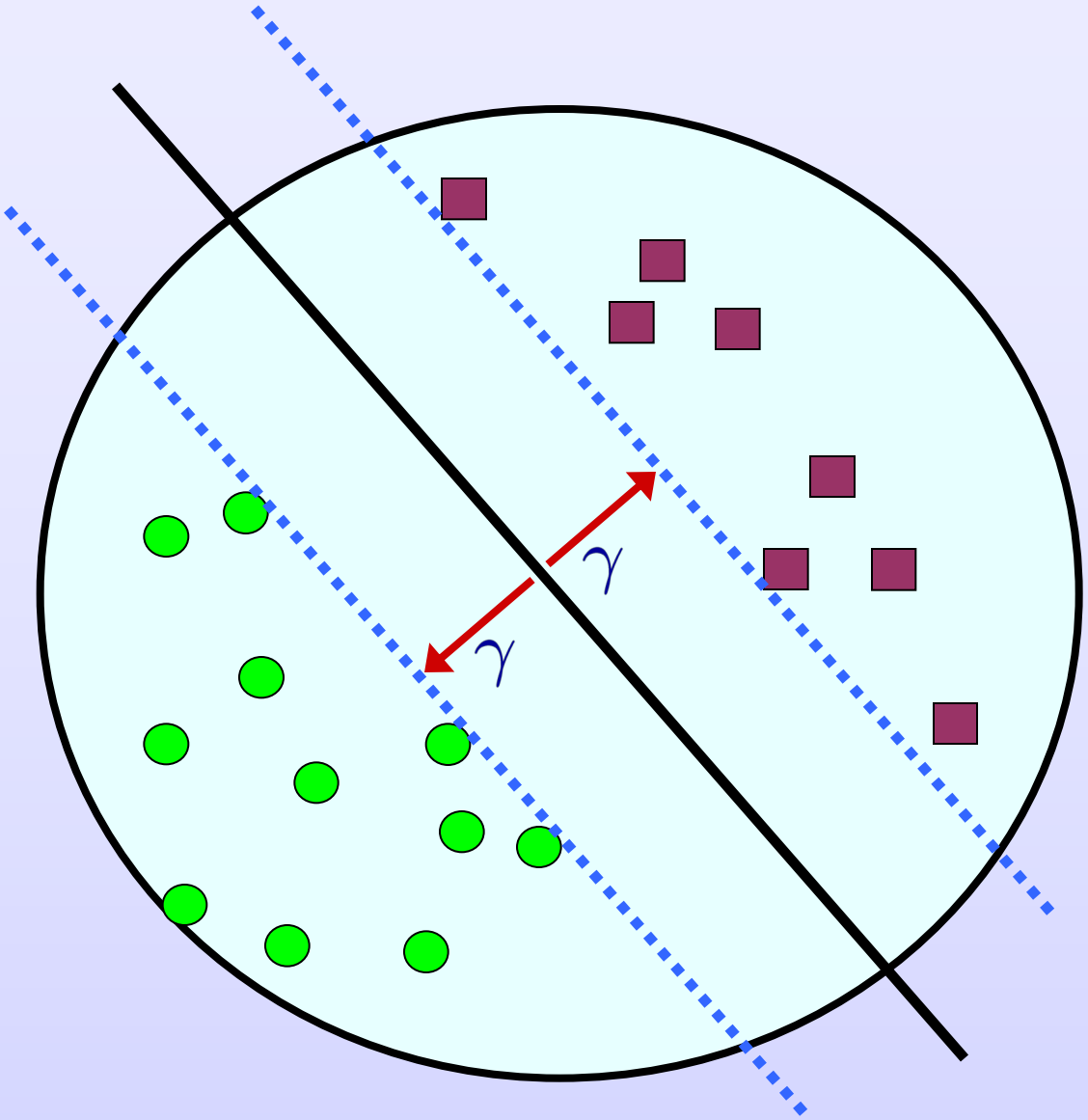
- Use 1st (parameterization) degree of freedom
- Scale the $\mathbf{u}$ such that $\|\mathbf{u}\| = 1$
- Define

$$\gamma = \min_t y_t(\mathbf{u} \cdot \mathbf{x}_t)$$

- The bound becomes

$$M \leq \frac{R^2}{\gamma^2}$$

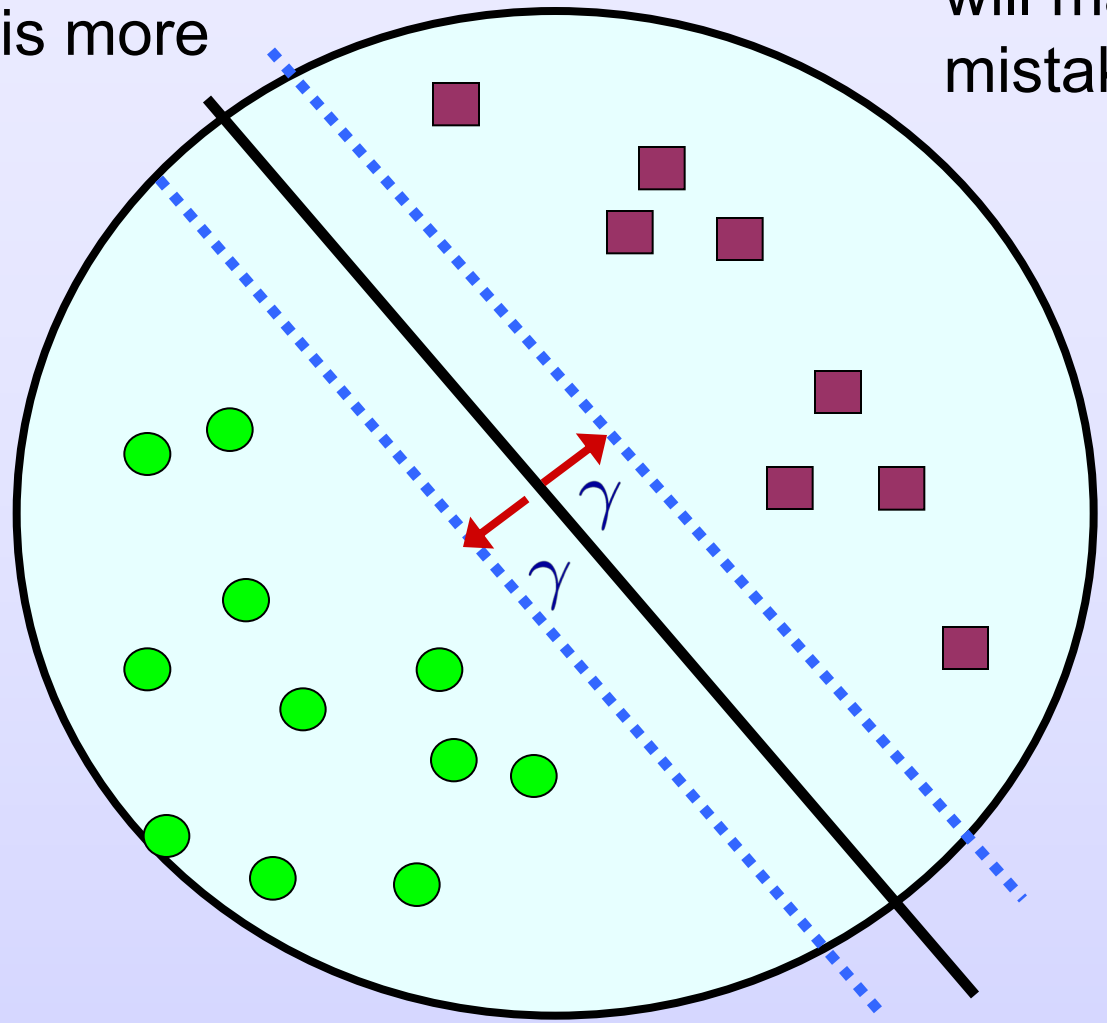
Separable Case - Illustration



separable Case – Illustration

Finding a separating hyperplance is more difficult

The Perceptron will make more mistakes



Inseparable Case

$$
\begin{aligned}
M &\leq \frac{1}{4}\left(R\|\mathbf{u}\| + \sqrt{\|\mathbf{u}\|^2 R^2 + 4\mathcal{L}_1}\right)^2 \\
M &\leq (R\|\mathbf{u}\| + \mathcal{L}_2)^2 \\
M &\leq \left(R\|\mathbf{u}\| + \sqrt{\mathcal{L}_1}\right)^2
\end{aligned}
$$

- Difficult problem implies a large value of $\mathcal{L}_1$, $\mathcal{L}_2$
- In this case the Perceptron will make a large number of mistakes

Perceptron Algorithm

- Extremely easy to implement
- Relative loss bounds for separable and inseparable cases. Minimal assumptions (not iid)
- Easy to convert to a well-performing batch algorithm (under iid assumptions)
- Quantities in bound are not compatible : no. of mistakes vs. hinge-loss.
- Margin of examples is ignored by update
- Same update for separable case and inseparable case.

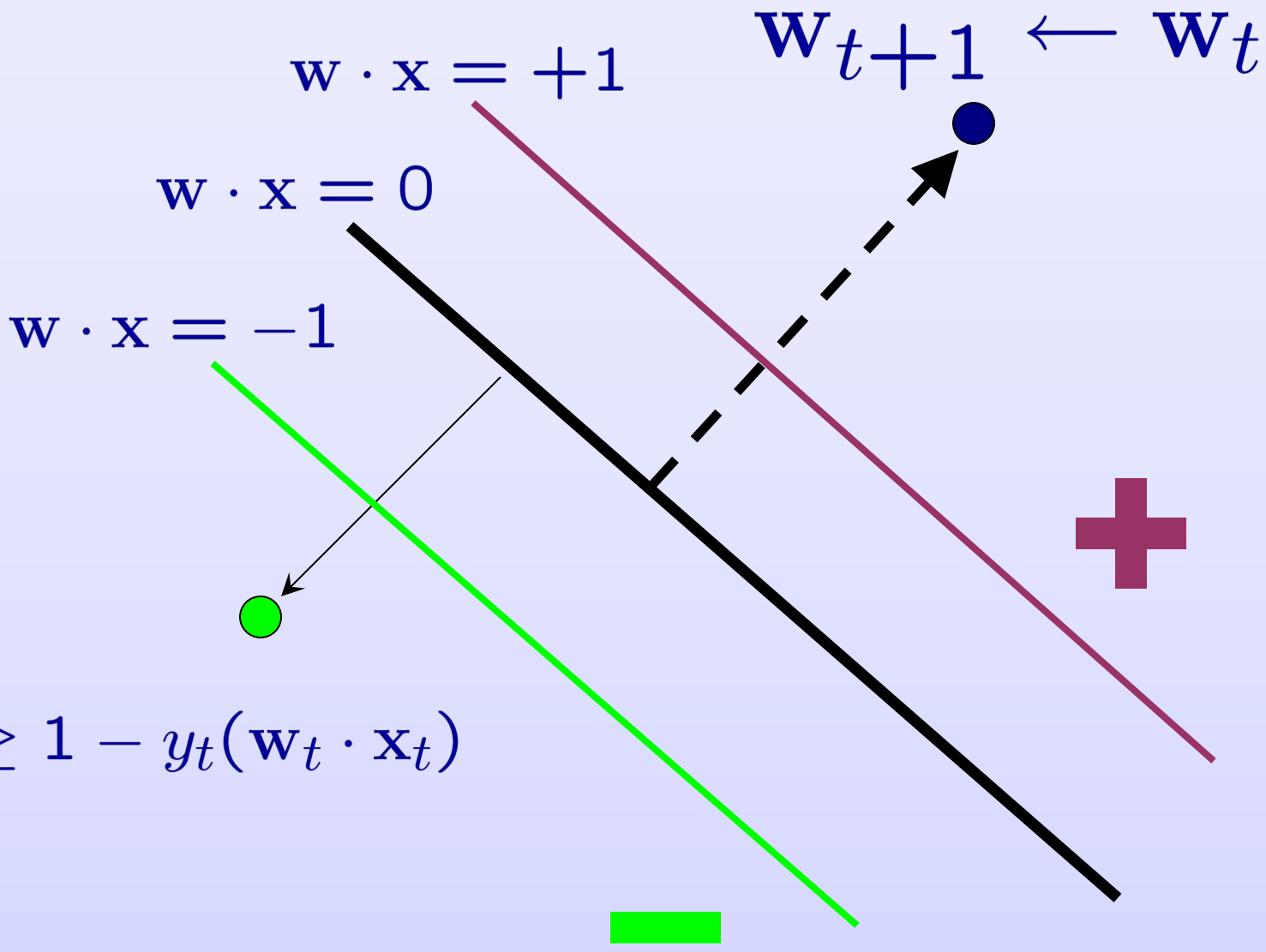
Passive – Aggressive Approach

- The basis for a well-known algorithm in convex optimization due to Hildreth (1957)
  - Asymptotic analysis
  - Does not work in the inseparable case
- Three versions :
  - PA separable case
  - PA-I PA-II inseparable case
- Beyond classification
  - Regression, one class, structured learning
- Relative loss bounds

Motivation

- Perceptron: No guaranties of margin *after* the update
- PA :Enforce a minimal non-zero margin after the update
- In particular :
  - If the margin is large enough (1), then do nothing
  - If the margin is less then unit, update such that the margin *after* the update is *enforced* to be unit

Input Space


$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t$
$\mathbf{w} \cdot \mathbf{x} = +1$
$\mathbf{w} \cdot \mathbf{x} = 0$
$\mathbf{w} \cdot \mathbf{x} = -1$
$0 \geq 1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)$

Input Space vs. Version Space

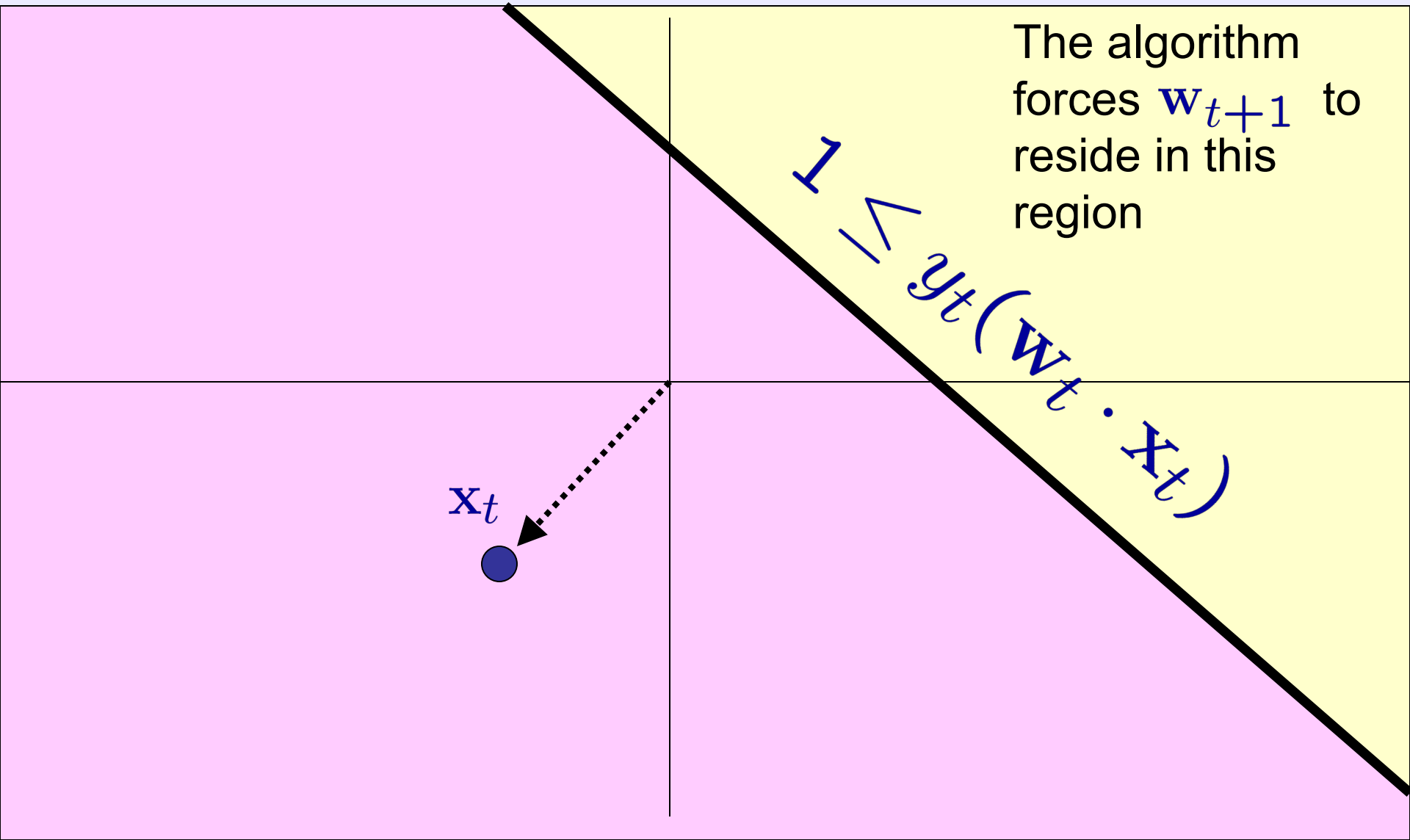
- Input Space :
  - Points are input data $y_t \mathbf{x}_t$
  - One constraint is induced by weight vector $\mathbf{w}$
  - Primal space
  - Half space = all input examples that are classified correctly by a given predictor (weight vector)

$$\{y\mathbf{x} \ : \ \mathbf{w} \cdot (y\mathbf{x}) \geq 0\}$$

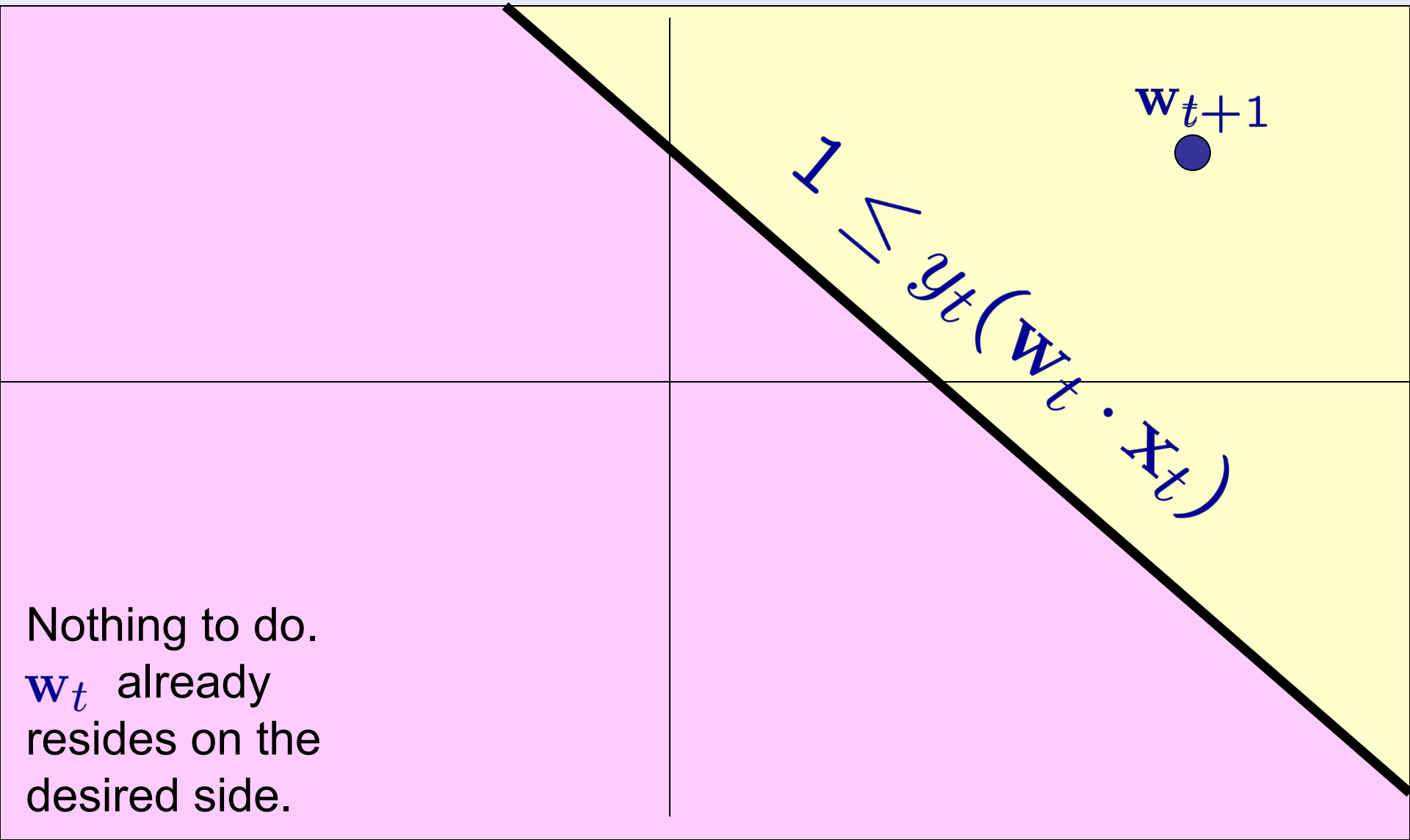
- Version Space :
  - Points are weight vectors $\mathbf{w}$
  - One constraints is induced by input data $y_t \mathbf{x}_t$
  - Dual space
  - Half space = all predictors (weight vectors) that classify correctly a given input example

$$\{\mathbf{w} \ : \ \mathbf{w} \cdot (y\mathbf{x}) \geq 0\}$$

Weight Vector (Version) Space

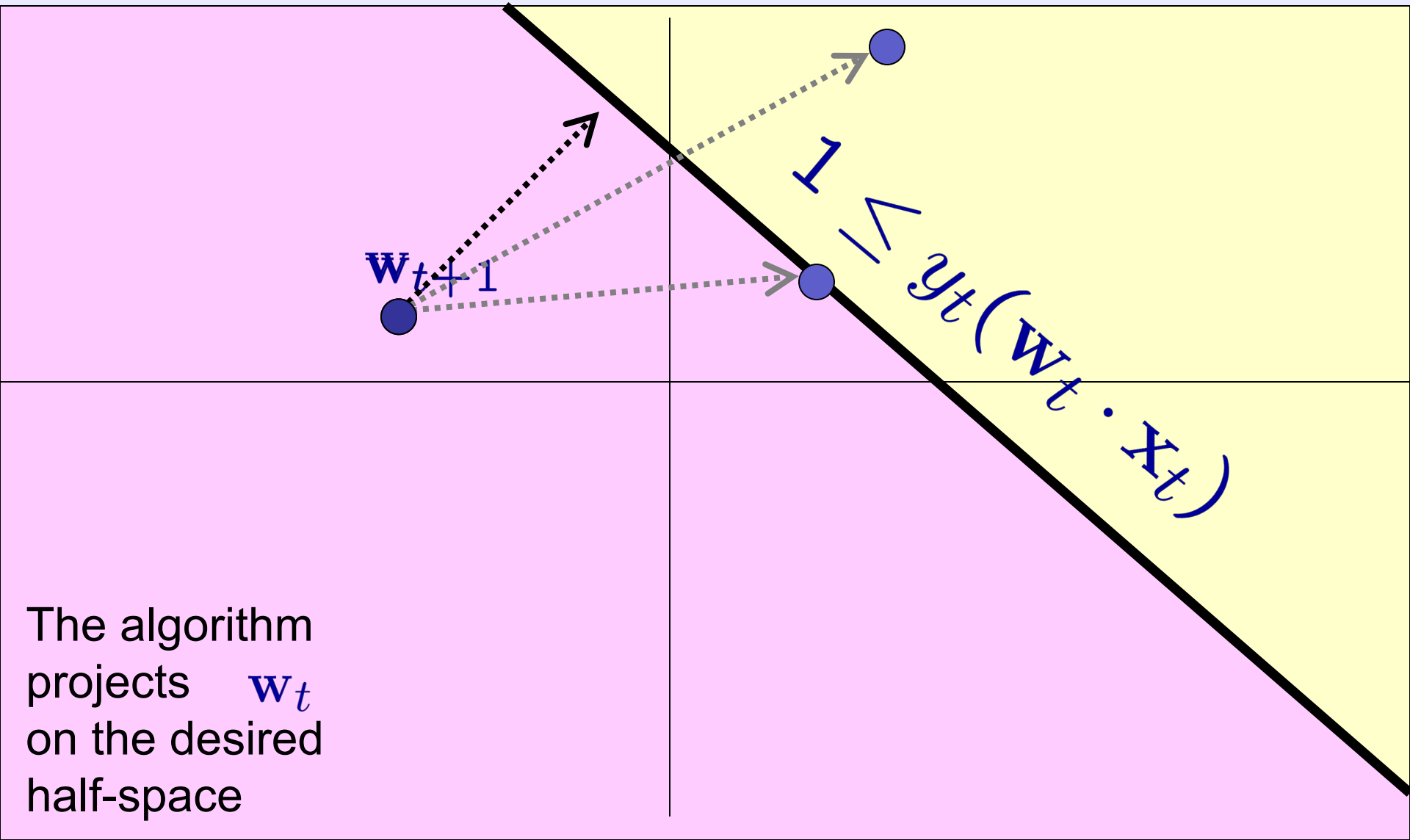


Passive Step



Nothing to do.
$\mathbf{w}_t$ already resides on the desired side.

Aggressive Step



The algorithm projects $\mathbf{w}_t$ on the desired half-space

Aggressive Update Step

- Set $\mathbf{w}_{t+1}$ to be the solution of the following optimization problem :

$$\mathbf{w}_{t+1} = \min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w} - \mathbf{w}_t\|^2$$
$$\text{s.t.} \quad y_t(\mathbf{w} \cdot \mathbf{x}_t) \geq 1$$

- The Lagrangian :

$$\mathcal{L}(\mathbf{w}, \tau) = \frac{1}{2}\|\mathbf{w} - \mathbf{w}_t\|^2 + \tau(1 - y_t(\mathbf{w} \cdot \mathbf{x}_t))$$

- Solve for the dual :

$$\max_{\tau \geq 0} \min_{\mathbf{w}} \mathcal{L}(\mathbf{w}, \tau)$$

Aggressive Update Step

- Optimize for $\mathbf{w}$:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \mathbf{w} - \mathbf{w}_t + \tau y_t \mathbf{x}_t$$

- Set the derivative to zero
- Substitute back into the Lagrangian :

$$\mathcal{L}(\max_{\tau \geq 0}) = -\frac{1}{2}\|\mathbf{x}_t\|^2 \tau^2 + \tau(1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t))$$

- Dual optimization problem

Aggressive Update Step

- Dual Problem :

$$\max_{\tau \geq 0} \quad -\frac{1}{2}\|\mathbf{x}_t\|^2\tau^2 + \tau(1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t))$$

- Solve it :

$$\tau = \max\left\{0, \frac{1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)}{\|\mathbf{x}_t\|^2}\right\}$$

- What about the constraint?

Alternative Derivation

- Additional Constraint (linear update) :

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \tau y_t \mathbf{x}_t$$

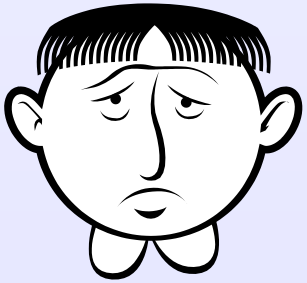
- Force the constraint to hold as equality

$$1 = y_t((\mathbf{w}_t + \tau y_t \mathbf{x}_t) \cdot \mathbf{x}_t) = y_t(\mathbf{w}_t \cdot \mathbf{x}_t) + \tau \|\mathbf{x}_t\|^2$$

- Solve :

$$\tau = \frac{1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)}{\|\mathbf{x}_t\|^2}$$

Passive-Aggressive Update



$$\mathbf{w}_{t+1} = \mathbf{w}_t + \tau y_t \mathbf{x}_t$$

$$y_t(\mathbf{w}_t \cdot \mathbf{x}_t) \geq 1$$

$$y_t(\mathbf{w}_t \cdot \mathbf{x}_t) < 1$$

$$\tau = 0$$

$$\tau = \frac{1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)}{\|\mathbf{x}_t\|^2}$$

Perceptron vs. PA

- Common Update :

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \tau y_t \mathbf{x}_t$$

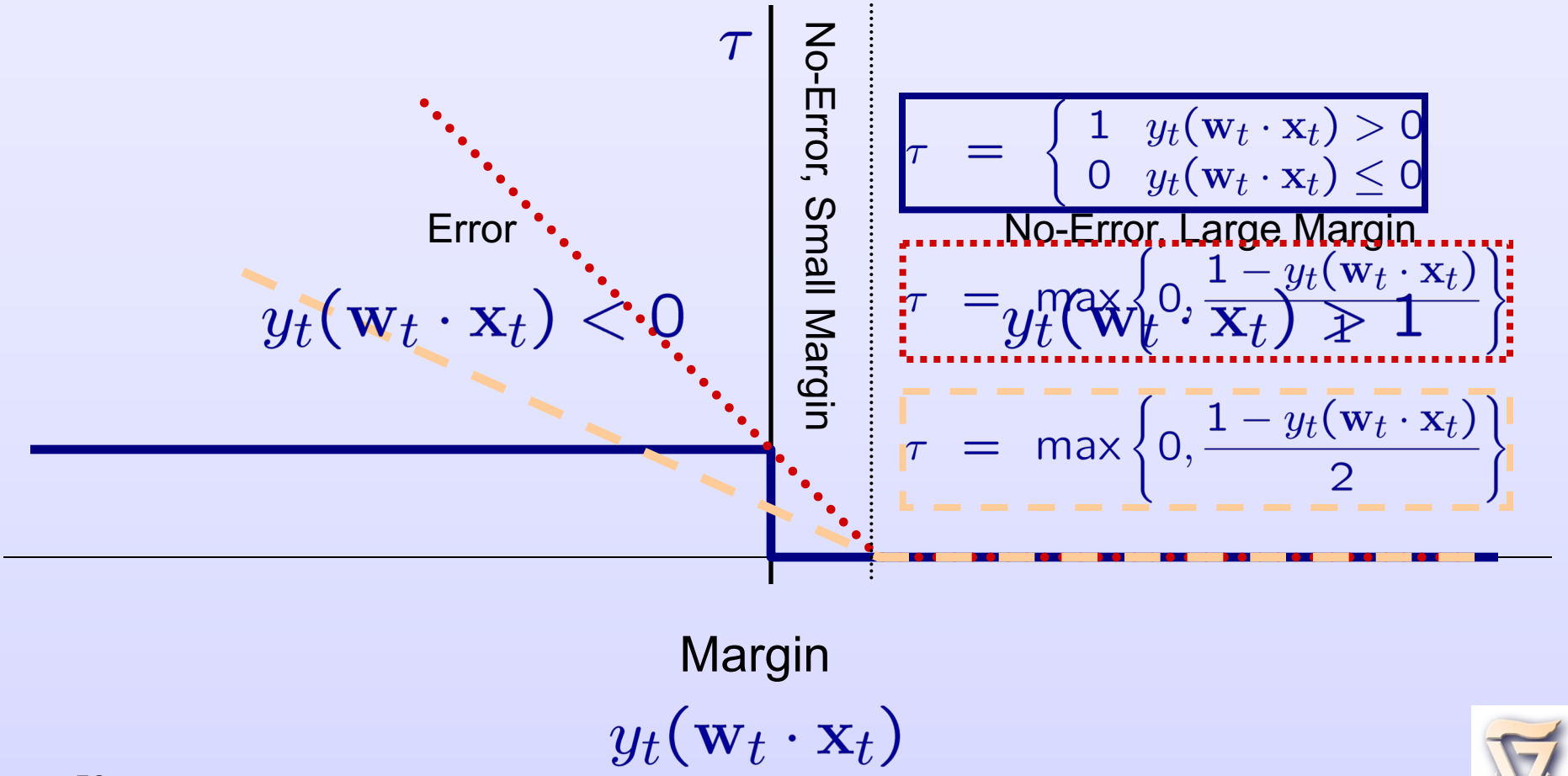
- Perceptron

$$\tau = \begin{cases} 1 & y_t(\mathbf{w}_t \cdot \mathbf{x}_t) > 0 \\ 0 & y_t(\mathbf{w}_t \cdot \mathbf{x}_t) \leq 0 \end{cases}$$

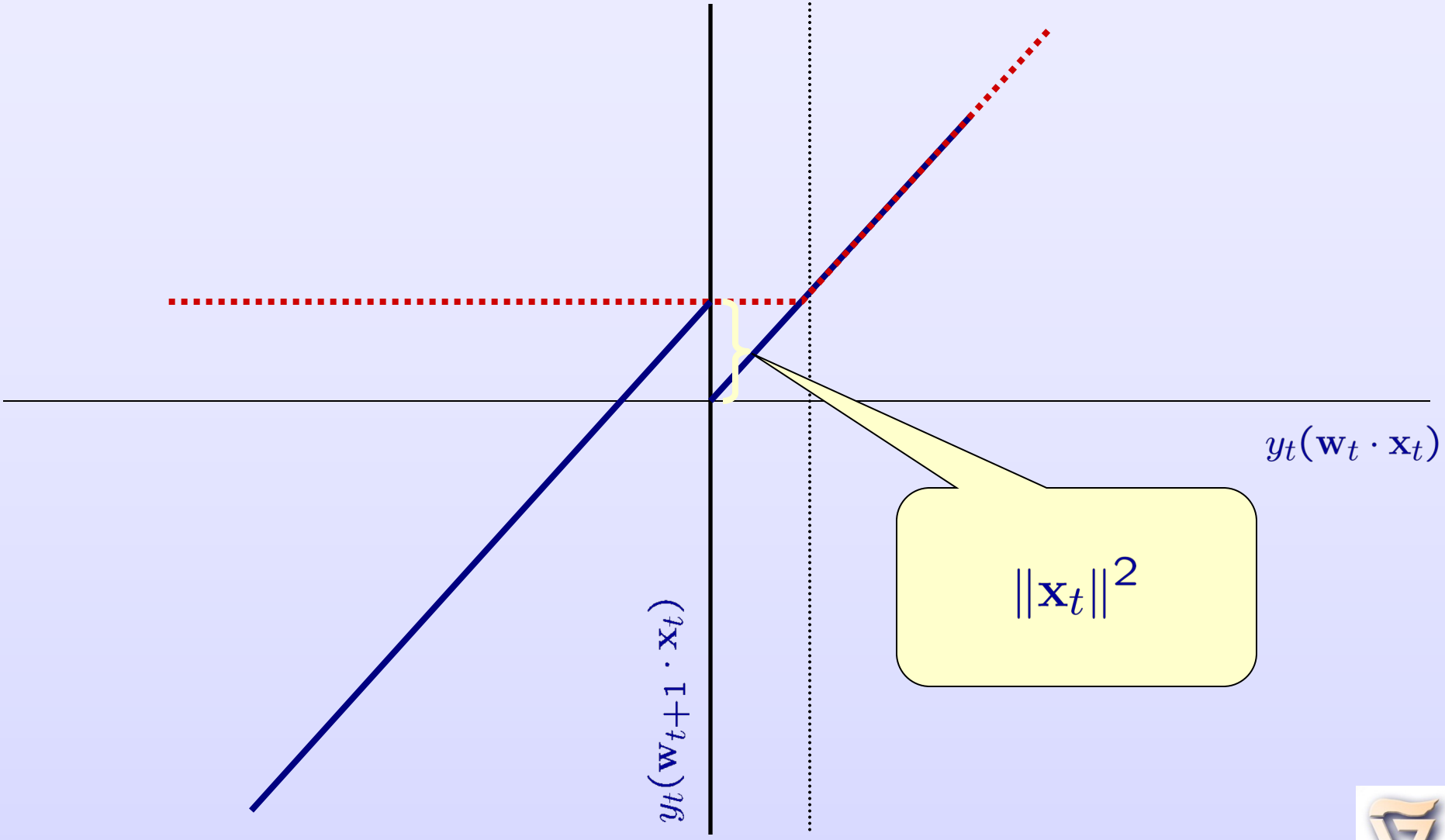
- Passive-Aggressive

$$\tau = \max\left\{0, \frac{1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)}{\|\mathbf{x}_t\|^2}\right\}$$

Perceptron vs. PA

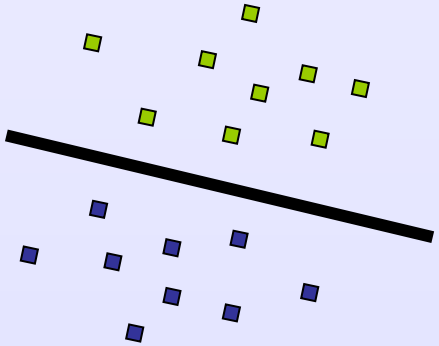


Perceptron vs. PA



Three Decision Problems

Classification

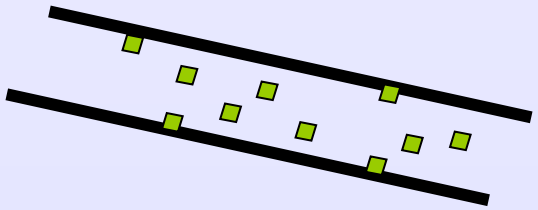


$$y_t \mathbf{w} \cdot \mathbf{x}_t \geq \widetilde{\epsilon}$$

$$\mathbf{z}_t = (\mathbf{x}_t, y_t)$$

$$(\mathbf{x}_t \in \mathcal{R}^n, y_t \in \{-1, 1\})$$

Regression

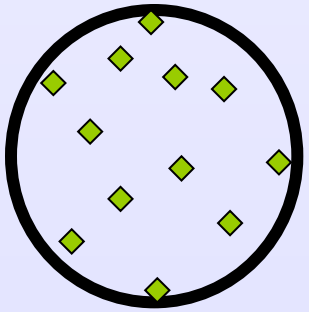


$$|\mathbf{w} \cdot \mathbf{x}_t - y_t| \leq \widetilde{\epsilon}$$

$$\mathbf{z}_t = (\mathbf{x}_t, y_t)$$

$$(\mathbf{x}_t \in \mathcal{R}^n, y_t \in \mathcal{R})$$

Uniclass



$$\|\mathbf{y}_t - \mathbf{w}\| \leq \widetilde{\epsilon}$$

$$\mathbf{z}_t = \mathbf{y}_t$$

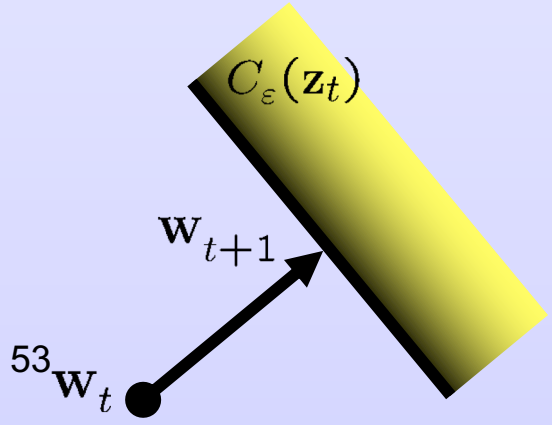
$$\mathbf{y}_t \in \mathcal{R}^n$$

The Passive-Aggressive Algorithm

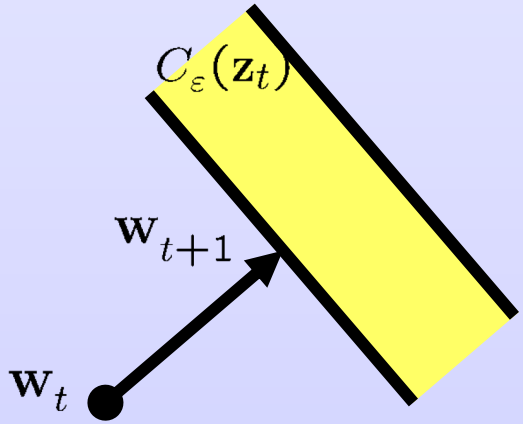
- Each example defines a set of consistent hypotheses: $C_\varepsilon(\mathbf{z}_t) = \{\mathbf{w} \mid \delta(\mathbf{w}; \mathbf{z}_t) \leq \varepsilon\}$
- The new vector $\mathbf{w}_{t+1}$ is set to be the projection of $\mathbf{w}_t$ onto $C_\varepsilon(\mathbf{z}_t)$

$$\mathbf{w}_{t+1} = \underset{\mathbf{w}}{\operatorname{argmin}} \|\mathbf{w} - \mathbf{w}_t\| \quad \text{s.t.} \quad \mathbf{w} \in C_\varepsilon(\mathbf{z}_t)$$

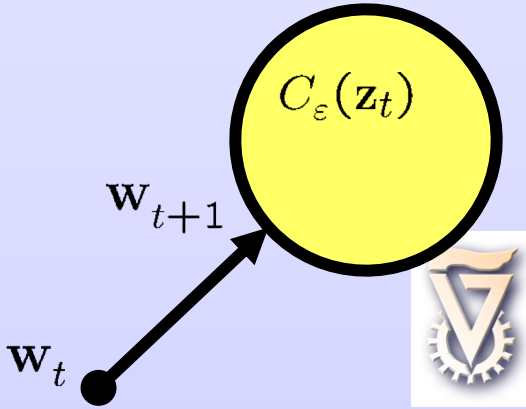
Classification



Regression



Uniclass



Loss Bound

- Assume there exists $\mathbf{u}$ such that

$$y_t(\mathbf{u} \cdot \mathbf{x}_t) \geq 1 \quad \forall t = 1 \ldots T$$

- Assume :

$$\|\mathbf{x}_t\| \leq R \quad \forall t = 1 \ldots T$$

- Then :

$$\sum_t \left(\ell(\mathbf{w}_t \cdot \mathbf{x}_t, y_t)^2 \right) \leq R^2 \|\mathbf{u}\|^2$$

- Note :

$$\ell_{01}(\mathbf{w}_t \cdot \mathbf{x}_t, y_t) \leq \ell(\mathbf{w}_t \cdot \mathbf{x}_t, y_t) \leq R^2 \|\mathbf{u}\|^2$$

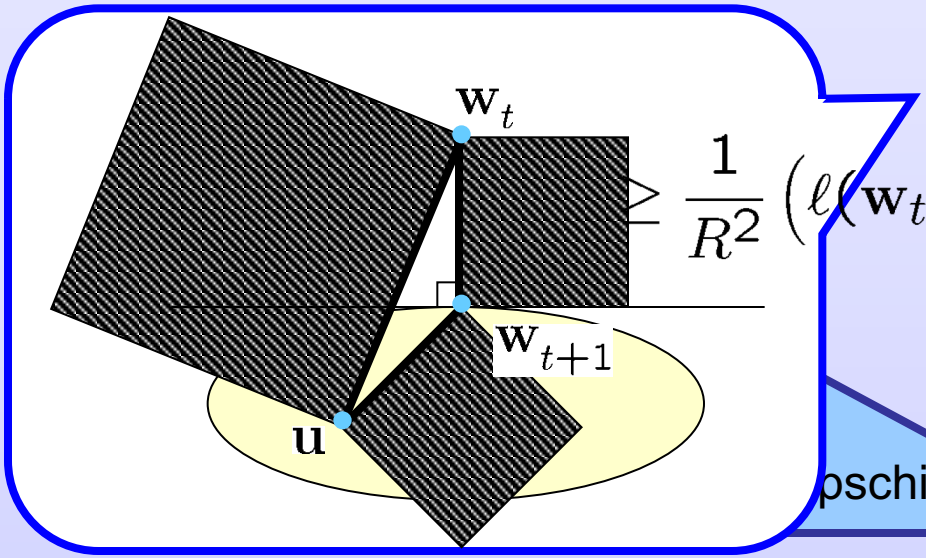
Proof Sketch

- Define:
- Upper bound:
- Lower bound:

$$\Delta_t = \|\mathbf{w}_t - \mathbf{u}\|^2 - \|\mathbf{w}_{t+1} - \mathbf{u}\|^2$$

$$\sum_{t=1}^{T} \Delta_t \leq \|\mathbf{w}_1 - \mathbf{u}\|^2$$

$$\Delta_t = \|\mathbf{w}_t - \mathbf{u}\|^2 - \|\mathbf{w}_{t+1} - \mathbf{u}\|^2$$



$$\geq \|\mathbf{w}_t - \mathbf{w}_{t+1}\|^2$$

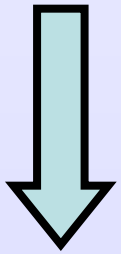
$$\geq \frac{1}{R^2}\left(\ell(\mathbf{w}_t \cdot \mathbf{x}_t\ ,\ y_t) - \underbrace{\ell(\mathbf{w}_{t+1} \cdot \mathbf{x}_t\ ,\ y_t)}_{=0}\right)^2$$

pschitz Condition

Proof Sketch (Cont.)

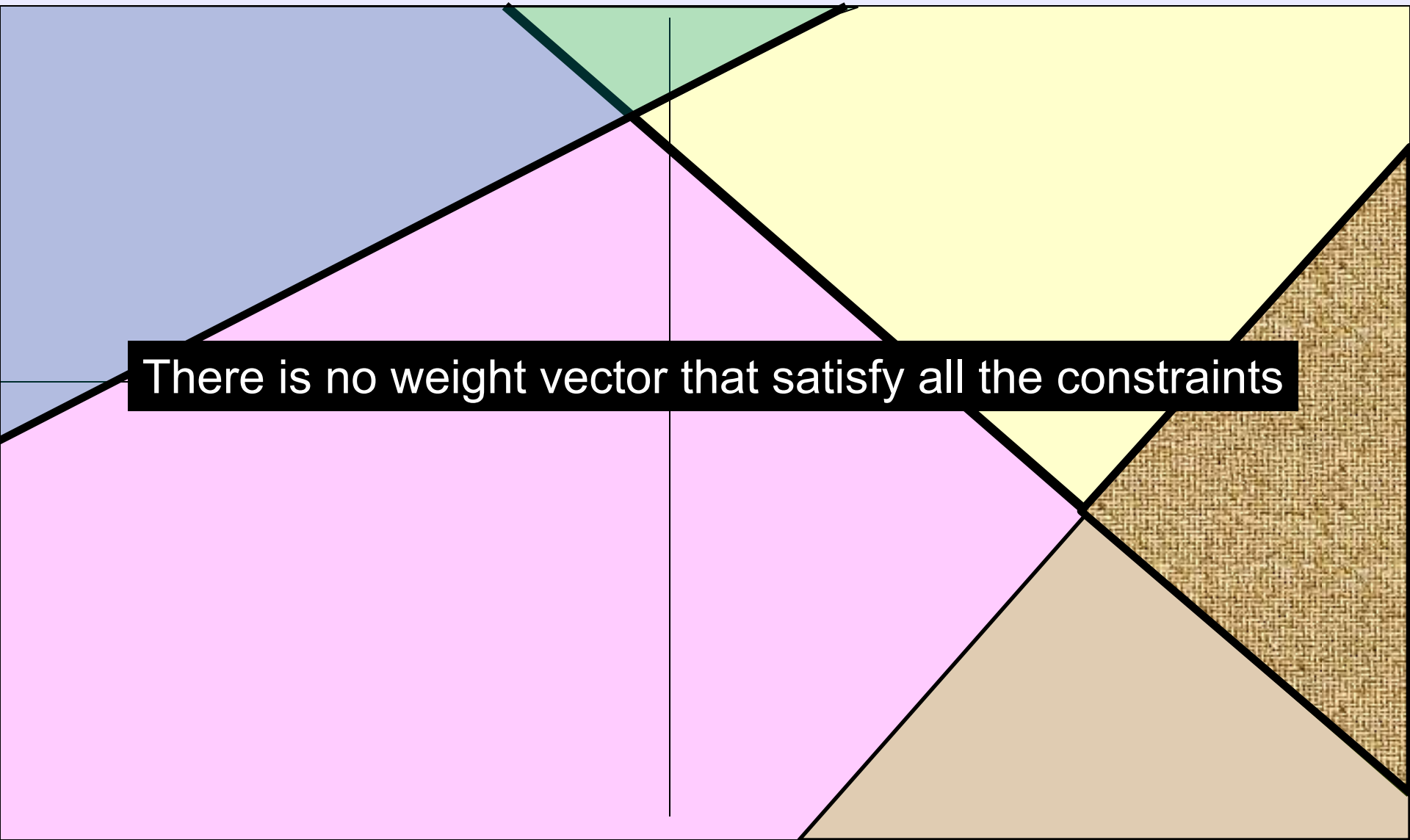
- Combining upper and lower bounds

$$\frac{1}{R^2}\sum_{t=1}^{T}\left(\ell(\mathbf{w}_t\cdot\mathbf{x}_t\ ,\ y_t)\right)^2\ \leq\ \sum_{t=1}^{T}\Delta_t\ \leq\ \left\|\mathbf{w}_1-\mathbf{u}\right\|^2$$

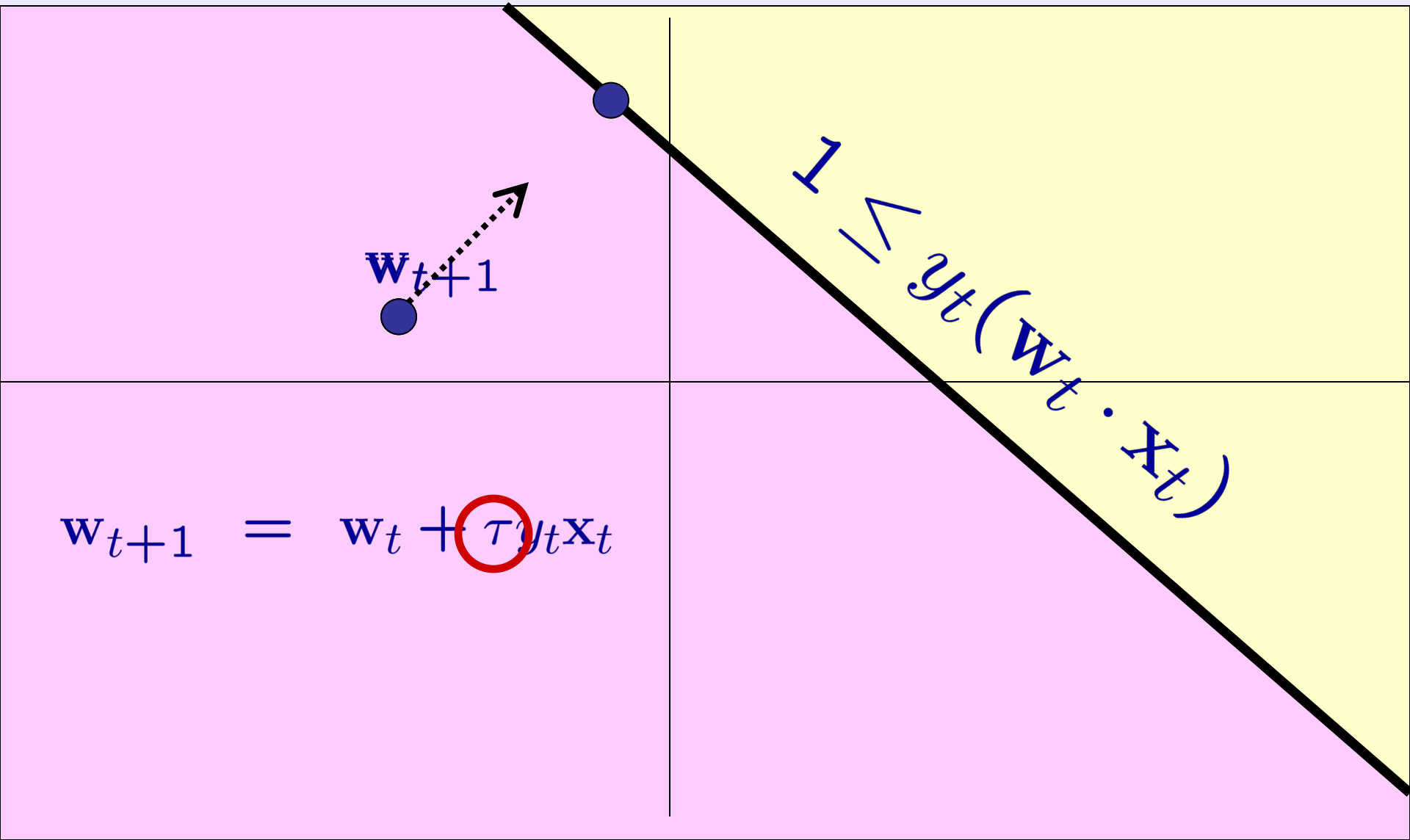


$$\sum_{t=1}^{T}\left(\ell(\mathbf{w}_t\cdot\mathbf{x}_t\ ,\ y_t)\right)^2\ \leq\ R^2\ \left\|\mathbf{w}_1-\mathbf{u}\right\|^2$$

Unrealizable case



Unrealizable Case



Unrealizable Case

$$\mathbf{w}_{t+1} = \min_{\mathbf{w}} \quad \frac{1}{2}\|\mathbf{w} - \mathbf{w}_t\|^2 + C\xi^2$$
$$\text{s.t.} \quad y_t(\mathbf{w} \cdot \mathbf{x}_t) \geq 1 - \xi \qquad \xi \geq 0$$

$$\mathbf{w}_{t+1} = \mathbf{w}_t + \tau y_t \mathbf{x}_t$$

ξ

$$\min\left\{C, \max\left\{0, \frac{1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)}{\|\mathbf{x}_t\|^2}\right\}\right\}$$

ξ^2

$$\max\left\{0, \frac{1 - y_t(\mathbf{w}_t \cdot \mathbf{x}_t)}{\|\mathbf{x}_t\|^2 + C}\right\}$$

Loss Bound for PA-I

- Mistake bound :

$$M \leq \max\left\{R^2, \frac{1}{C}\right\}\left(2\|\mathbf{u}\|^2 + 2C\sum_t \ell(\mathbf{x}_t \cdot \mathbf{u}\ ,\ y_t)\right)$$

- Optimal value, set $C = \frac{1}{R^2}$

$$M \leq \left(2R^2\|\mathbf{u}\|^2 + 2\sum_t \ell(\mathbf{x}_t \cdot \mathbf{u}\ ,\ y_t)\right)$$

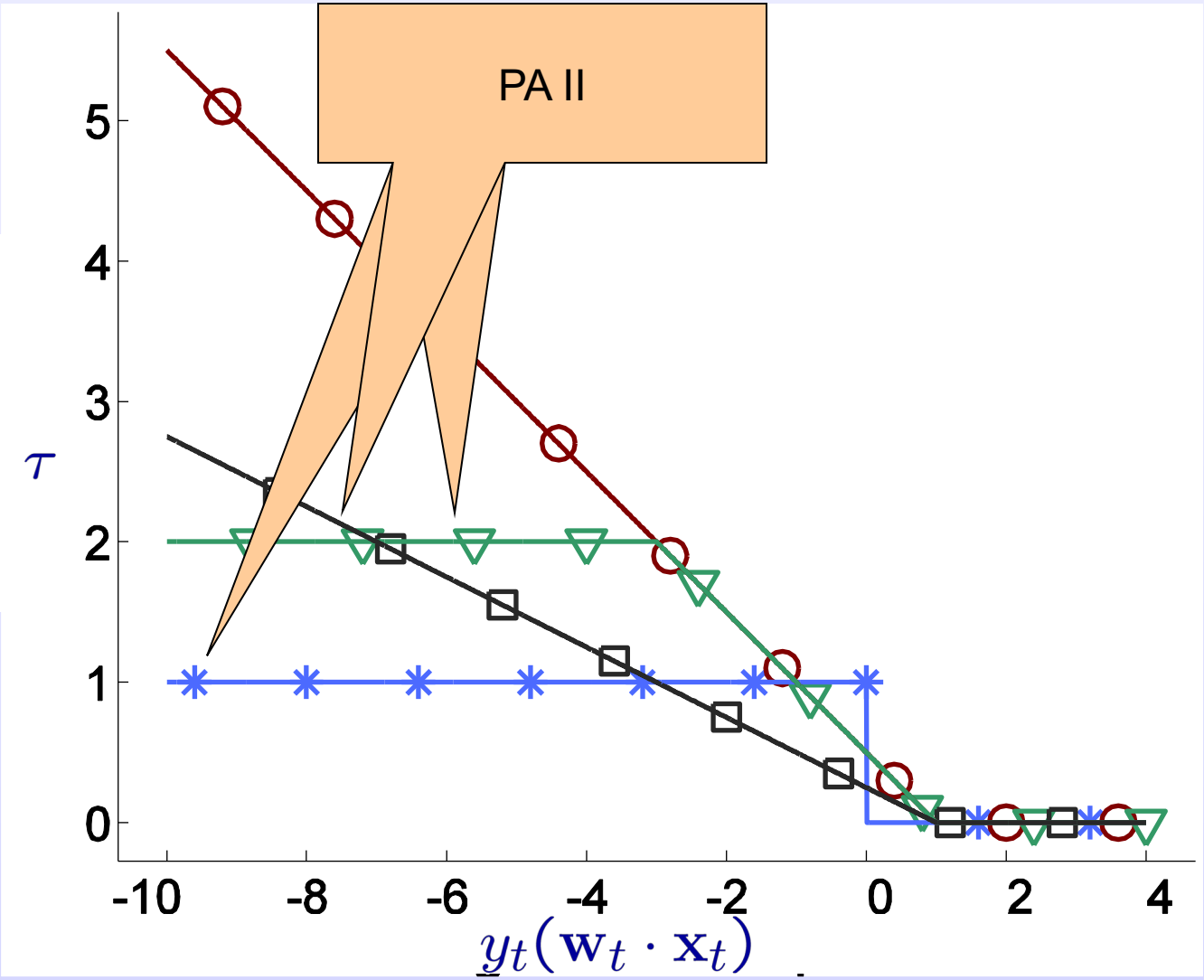
Loss Bound for PA-II

- Loss bound :

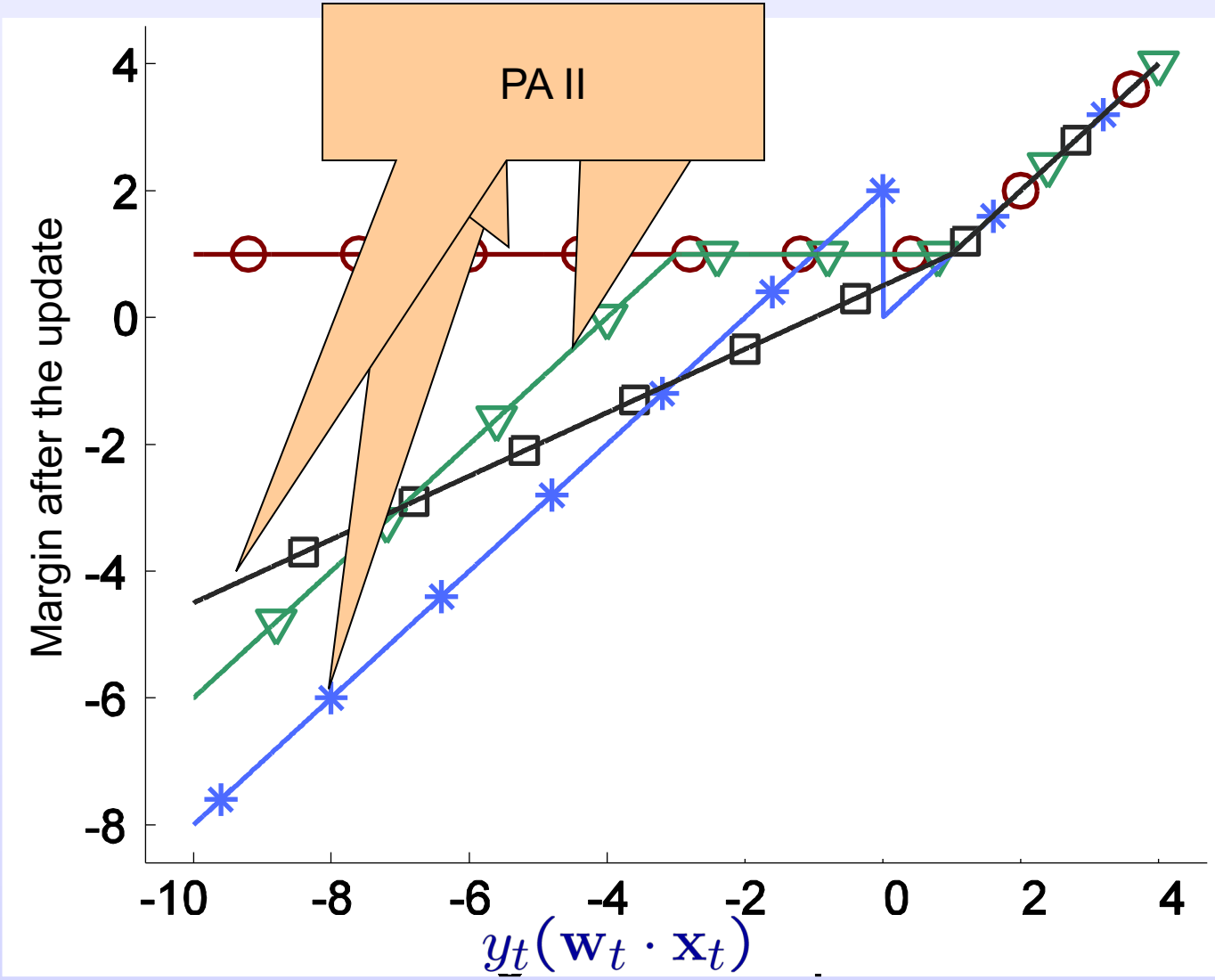
$$\sum_t \left(\ell(\mathbf{w}_t \cdot \mathbf{x}_t \ , \ y_t)^2\right)$$
$$\leq \ (C + R^2)\|\mathbf{u}\|^2 + \left(1 + \frac{R^2}{C}\right) \sum_t \left(\ell(\mathbf{u} \cdot \mathbf{x}_t \ , \ y_t)^2\right)$$

- Similar proof technique as of PA
- Bound can be improved similarly to the Perceptron

Four Algorithms



Four Algorithms



Concluding Remarks

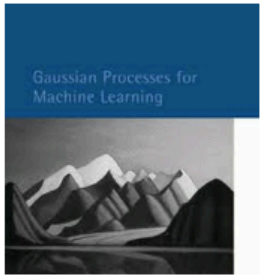
- Batch vs. Online
  - Two phases: Training and then Test
  - Single continues process
- Statistical Assumption
  - Distribution over examples
  - All sequences
- Conversions
  - Online -> Batch
  - Batch -> Online

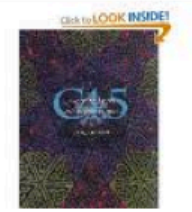
machine learning book

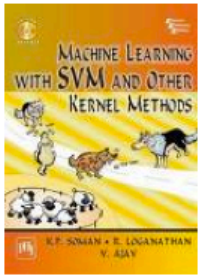
About 14,500,000 results (0.34 seconds)

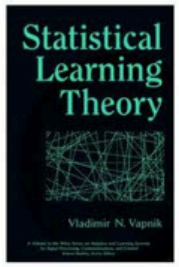
Advanced search

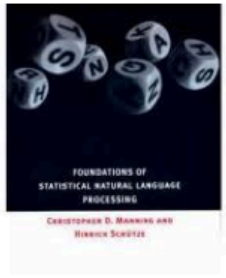
SafeSearch moderate ▼

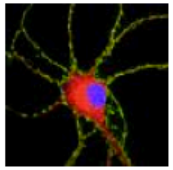
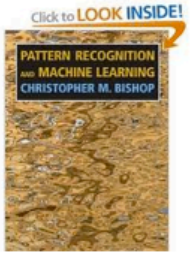


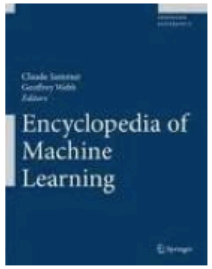


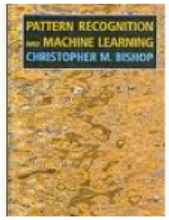


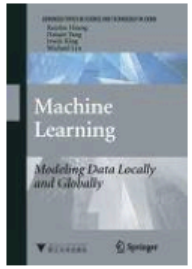


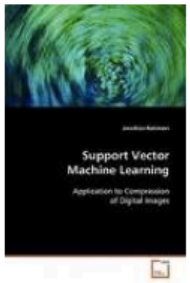


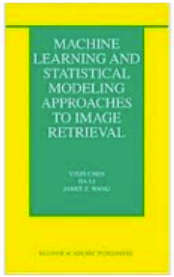


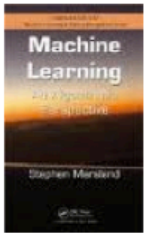


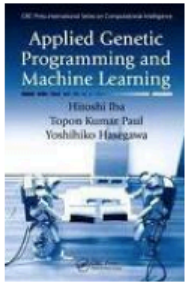


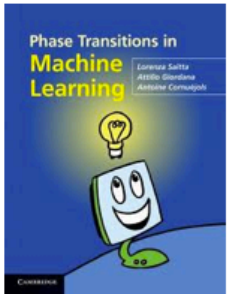


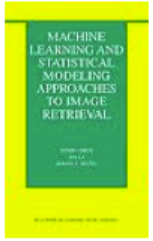


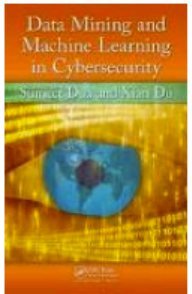












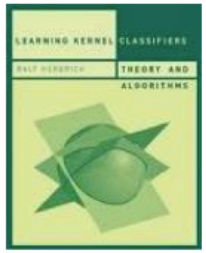

machine learning book

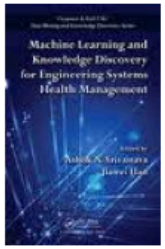
About 14,500,000 results (0.34 seconds)

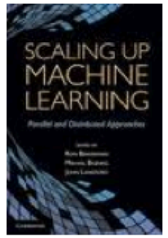
Advanced search

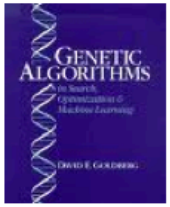
SafeSearch moderate ▼

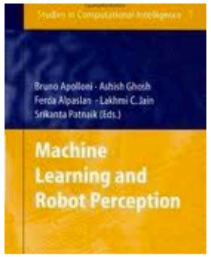
Page 2

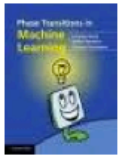


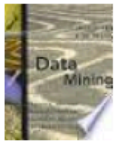


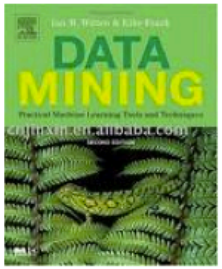


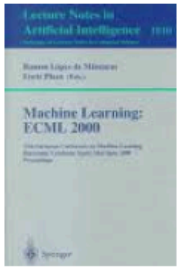


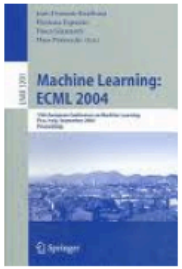


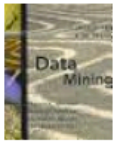


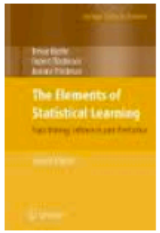


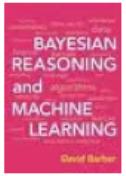


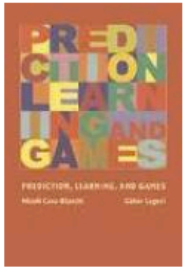


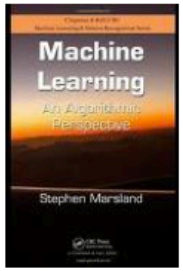










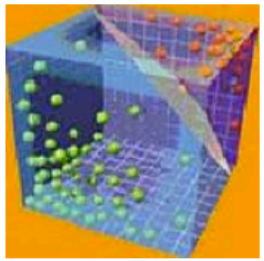


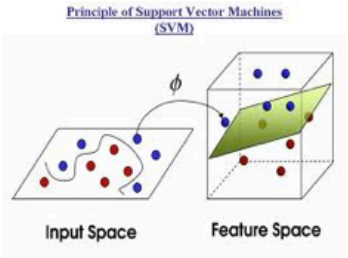
support vector machine book

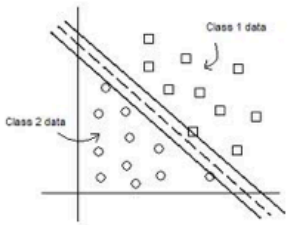
About 1,390,000 results (0.34 seconds)

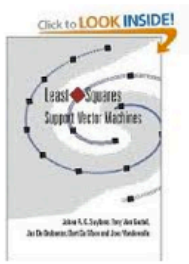
Advanced search

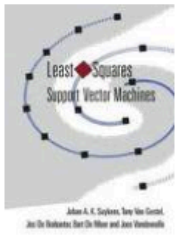
SafeSearch moderate ▼

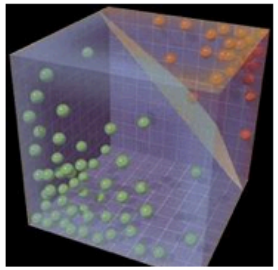


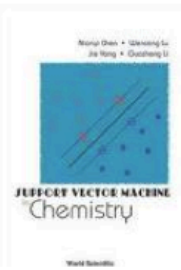


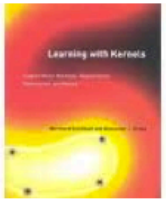


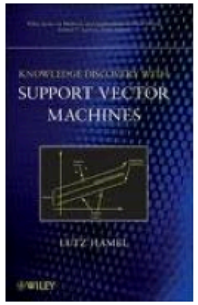


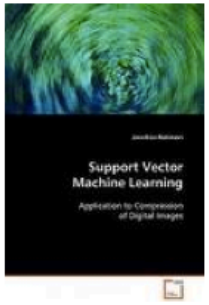


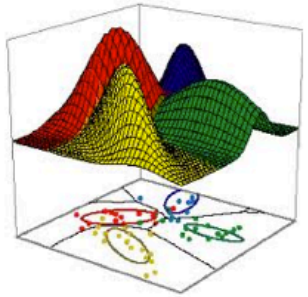


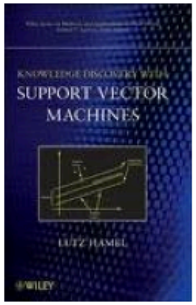


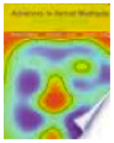


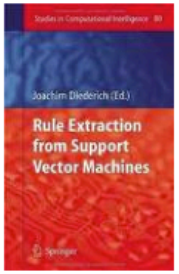


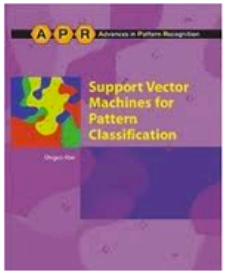


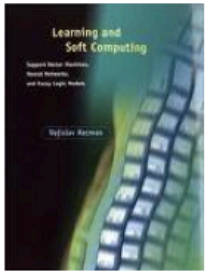


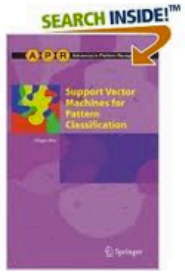


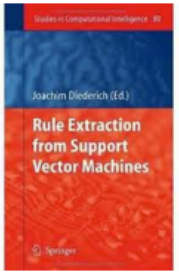










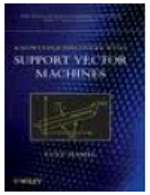


Page 2

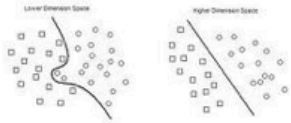

Click to LOOK INSIDE!
Support Vector Machines


Ingo Steinwart • Andreas Christmann
Support Vector Machines

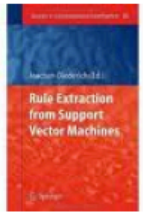

Click to LOOK INSIDE!
Pattern Recognition with Support Vector Machines

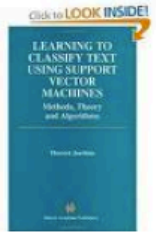

SUPPORT VECTOR MACHINES


Ingo Steinwart • Andreas Christmann
Support Vector Machines


Lower Dimension Space
Higher Dimension Space

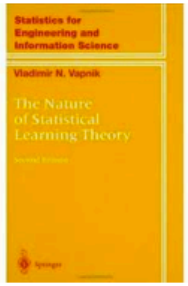

Automatic Text Categorization


Rule Extraction from Support Vector Machines


Click to LOOK INSIDE!
LEARNING TO CLASSIFY TEXT USING SUPPORT VECTOR MACHINES
Methods, Theory and Algorithms

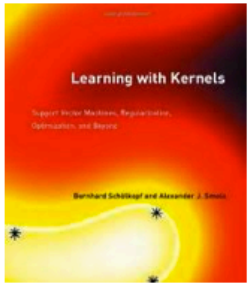

CLASSROOM IN A BOOK


Support Vector Machines for Pattern Classification


Statistics for Engineering and Information Science
Vladimir N. Vapnik
The Nature of Statistical Learning Theory


Pattern Recognition with Support Vector Machines

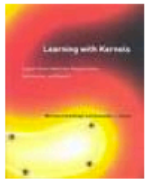

Support Vector Machines for Pattern Classification

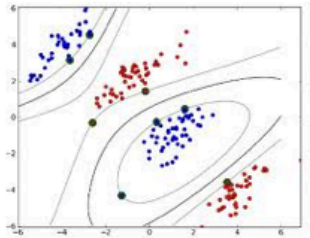

Learning with Kernels

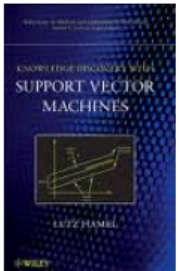

Support Vector Machines and Their Applications in
CHEMISTRY and BIOTECHNOLOGY


Bio-Inspired Credit Risk Analysis


Learning with Kernels


Learning with Kernels




SUPPORT VECTOR MACHINES

machine learning natural language processing book

About 700,000 results (0.34 seconds)

Advanced search

SafeSearch moderate ▼