Theory & Applications of Online Learning

Shai Shalev-Shwartz

Yoram Singer

ICML, July 5th 2008

Motivation - Spam Filtering

For $t = 1, 2, \ldots, T$

- Receive an email
- Expert advice: Apply d spam filters to get $\mathbf{x} \in \{+1, -1\}^d$
- Predict $\hat{y}_t \in \{+1, -1\}$
- Receive true label $y_t \in \{+1, -1\}$
- Suffer loss $\ell(y_t, \hat{y}_t)$

Motivation - Spam Filtering

Goal – Low Regret

- We don't know in advance the best performing expert
- We'd like to find the best expert in an online manner
- We'd like to make as few filtering errors as possible
- This setting is called "regret analysis". Our goal:

$$\sum_{t=1}^{T} \ell(\hat{y}_t, y_t) - \min_i \sum_{t=1}^{T} \ell(x_{t,i}, y_t) \;\leq\; o(T)$$

Regret Analysis

- Low regret means that we do not loose much from not knowing future events
- We can perform almost as well as someone who observes the entire sequence and picks the best prediction strategy in hindsight
- No statistical assumptions
- We can also compete with changing environment

Why Online ?

- In many cases, data arrives sequentially while predictions are required on-the-fly
- Applicable also in adversarial and competitive environments (e.g. spam filtering, stock market)
- Can adapt to changing environment
- Simple algorithms
- Theoretical guarantees
- Online-to-batch conversions, generalization properties

Outline

Tutorial's goals: provide design and analysis tools for online algorithms

Part I:
What prediction tasks are possible

Outline

Tutorial's goals: provide design and analysis tools for online algorithms

Part I:
What prediction tasks are possible

Regression with squared-loss

Classification with 0-1 loss

Outline

Tutorial's goals: provide design and analysis tools for online algorithms

Part I:
What prediction tasks are possible

Regression with squared-loss

Classification with 0-1 loss

Outline

Tutorial's goals: provide design and analysis tools for online algorithms

Part I:
What prediction tasks are possible

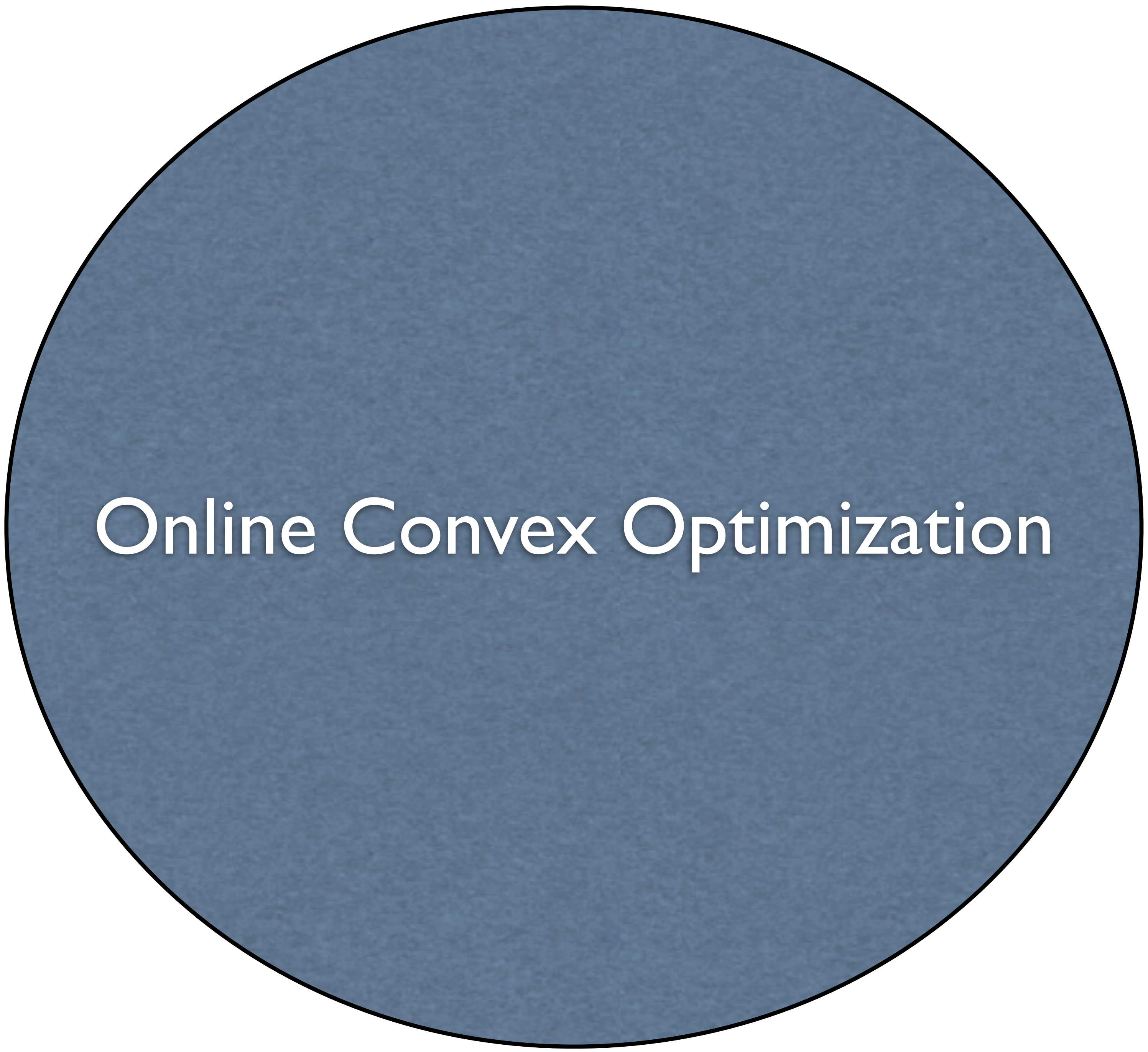
Regression with squared-loss

Classification with 0-1 loss

Convexity is a key property

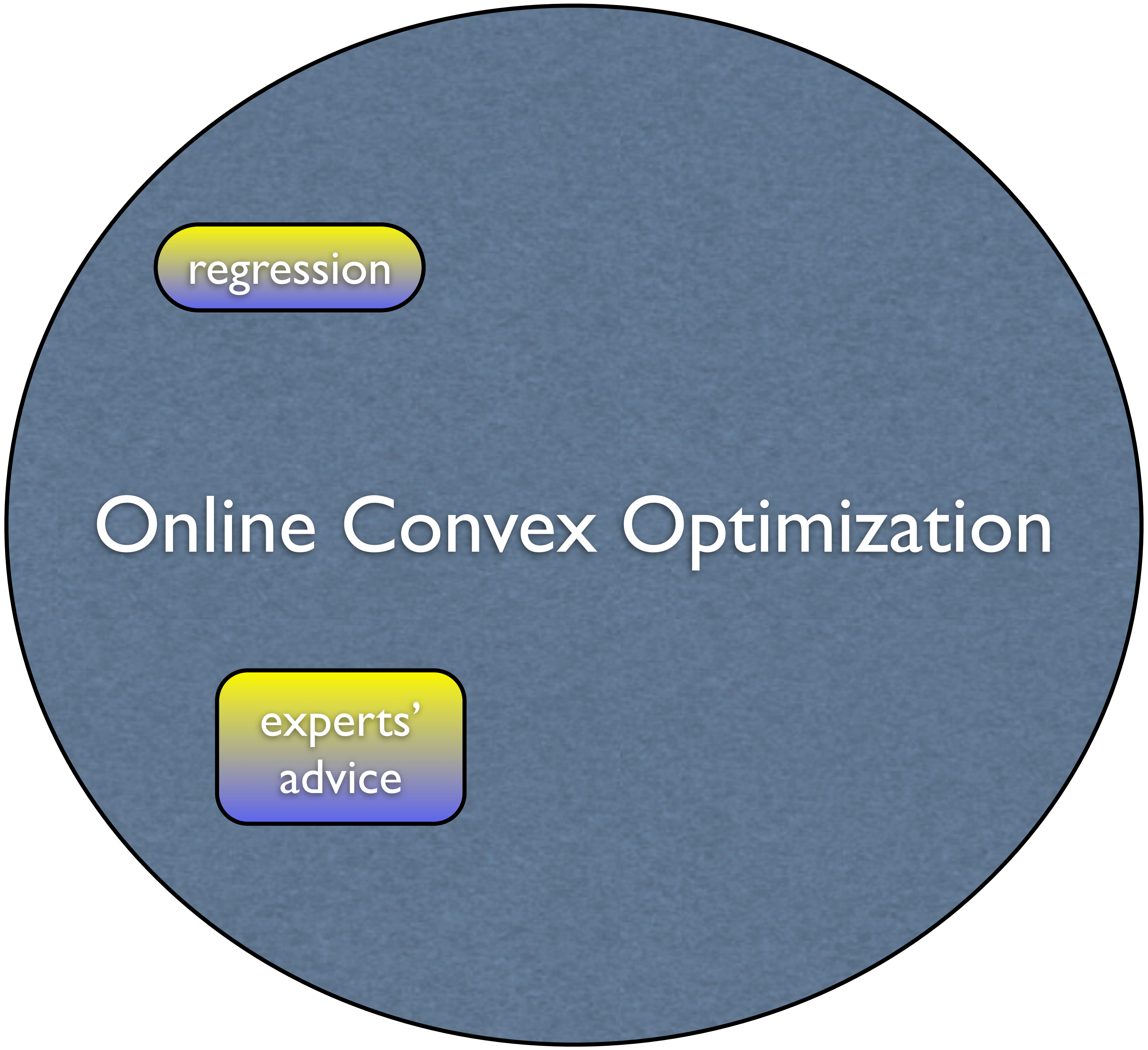
Outline

Tutorial's goals: provide design and analysis tools for online algorithms



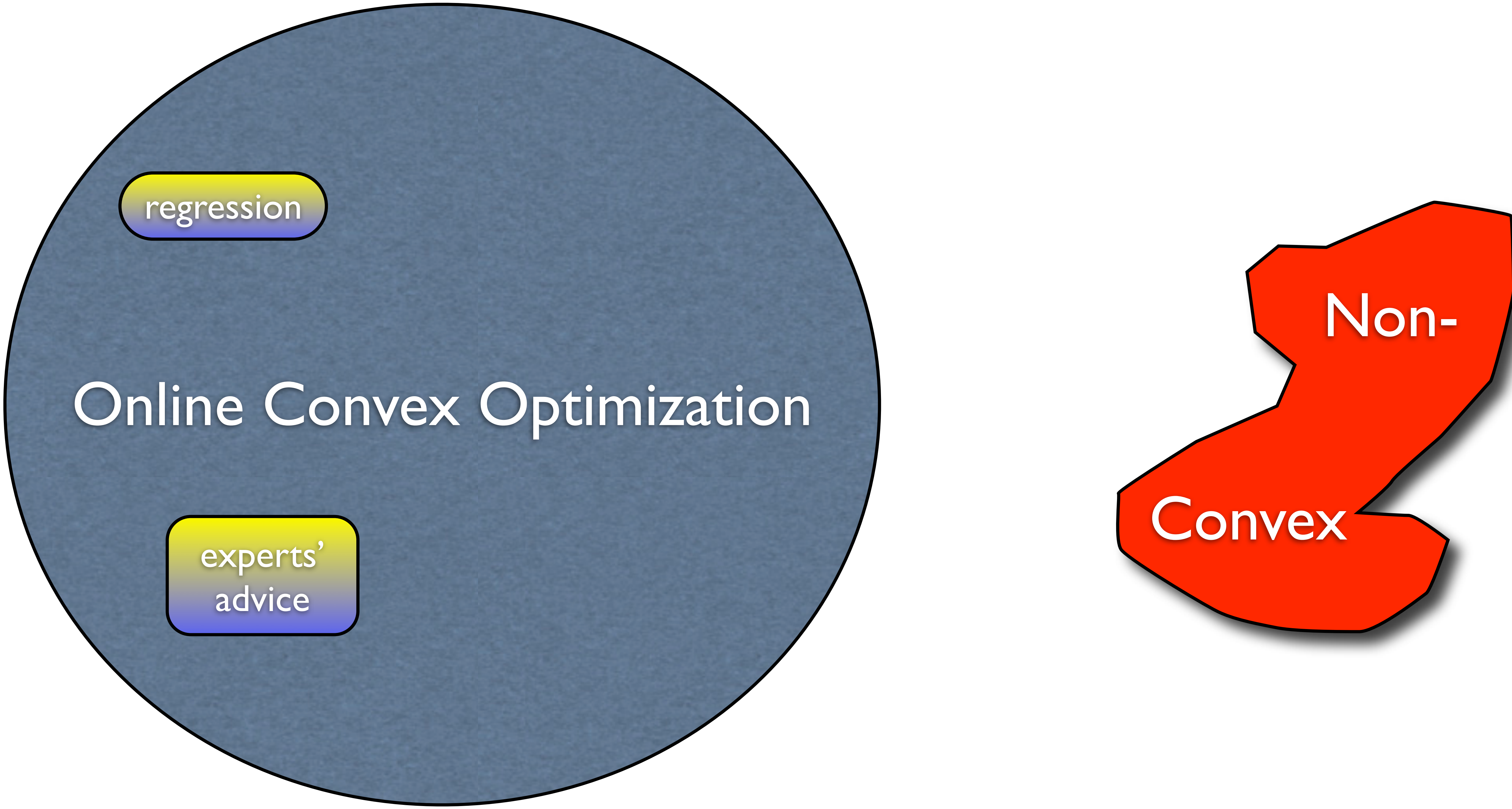
Outline

Tutorial's goals: provide design and analysis tools for online algorithms



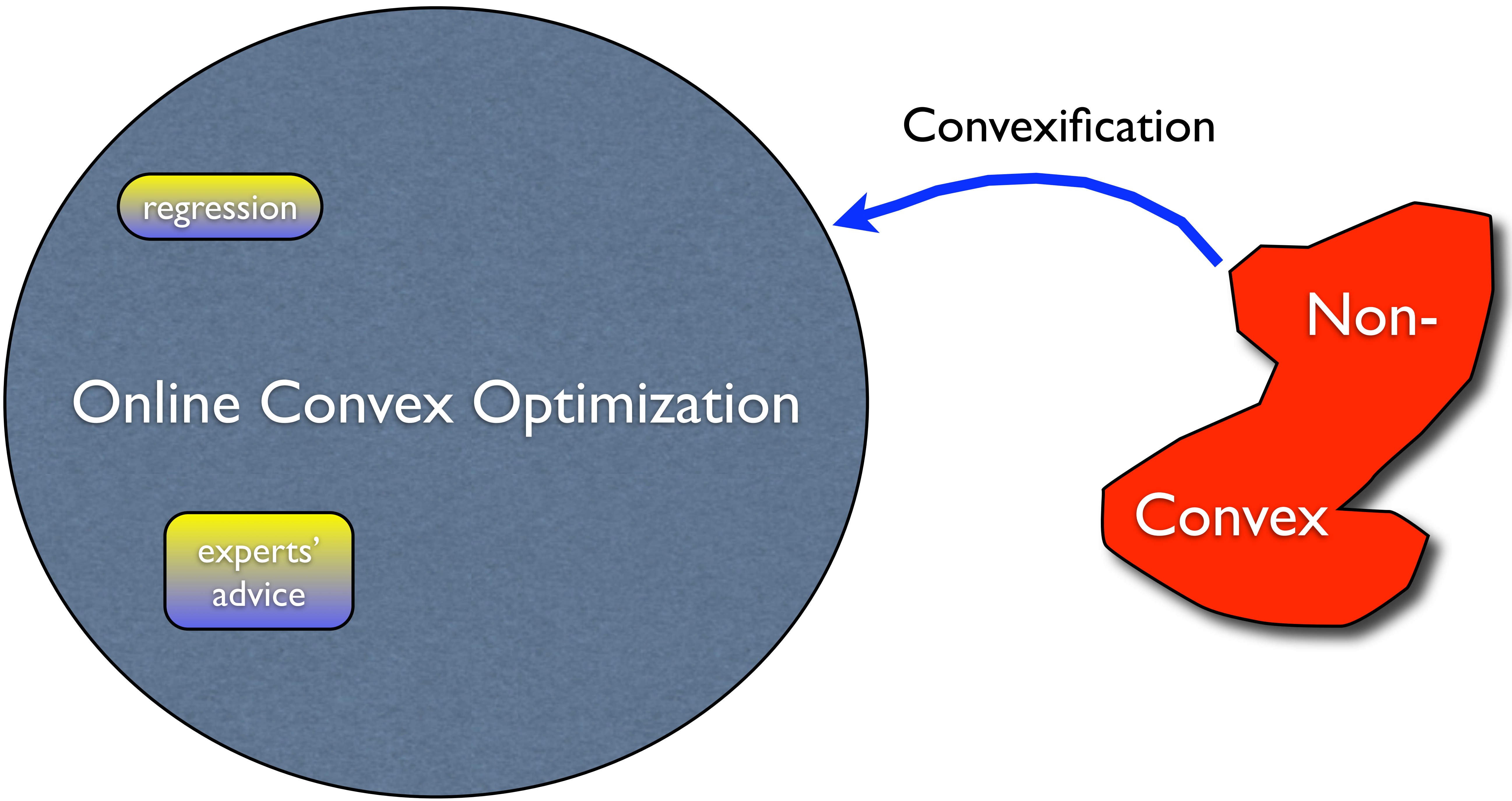
Outline

Tutorial's goals: provide design and analysis tools for online algorithms



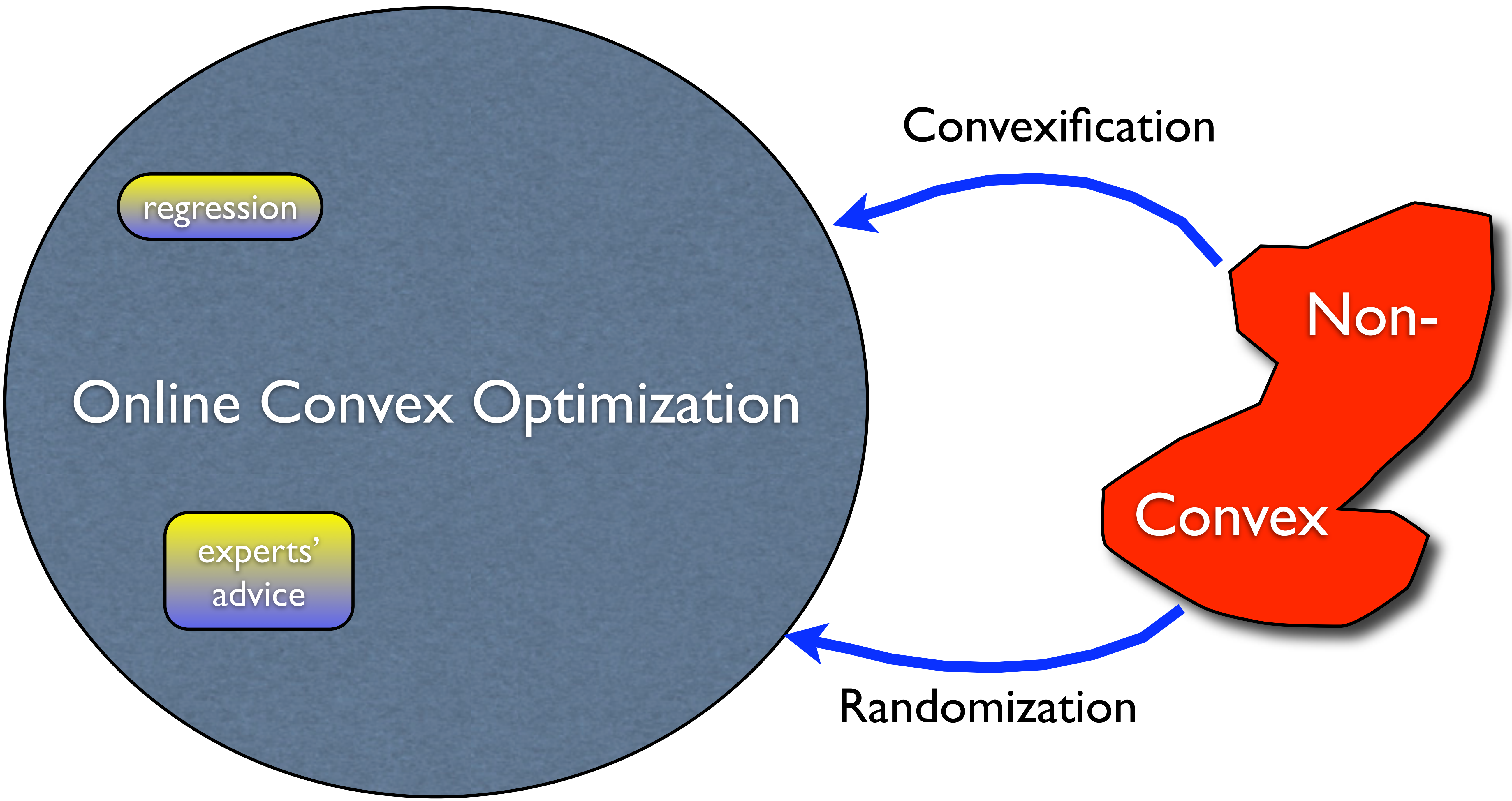
Outline

Tutorial's goals: provide design and analysis tools for online algorithms



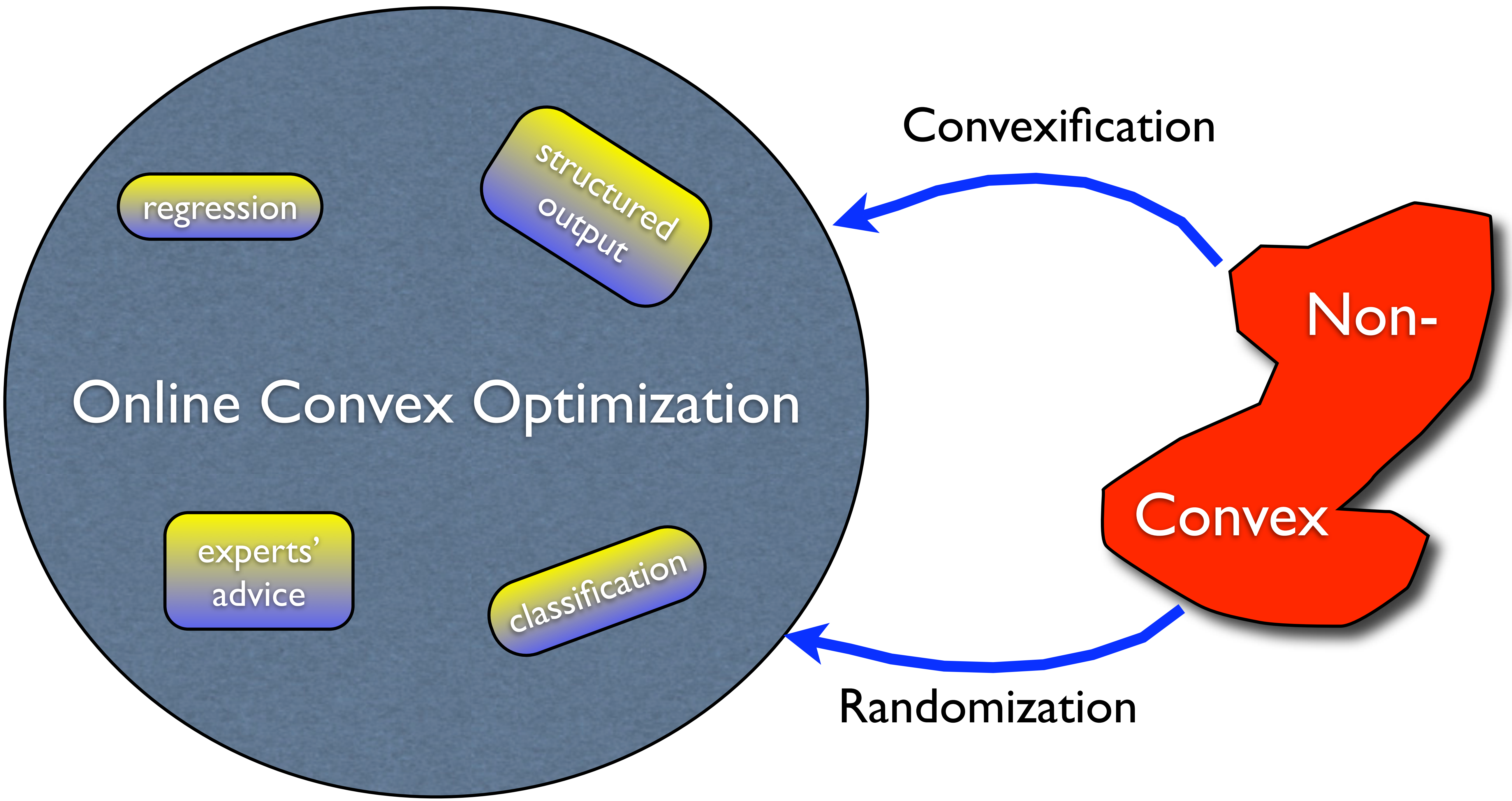
Outline

Tutorial's goals: provide design and analysis tools for online algorithms



Outline

Tutorial's goals: provide design and analysis tools for online algorithms



Outline

Tutorial's goals: provide design and analysis tools for online algorithms

Part II:
An algorithmic framework for online convex optimization

Outline

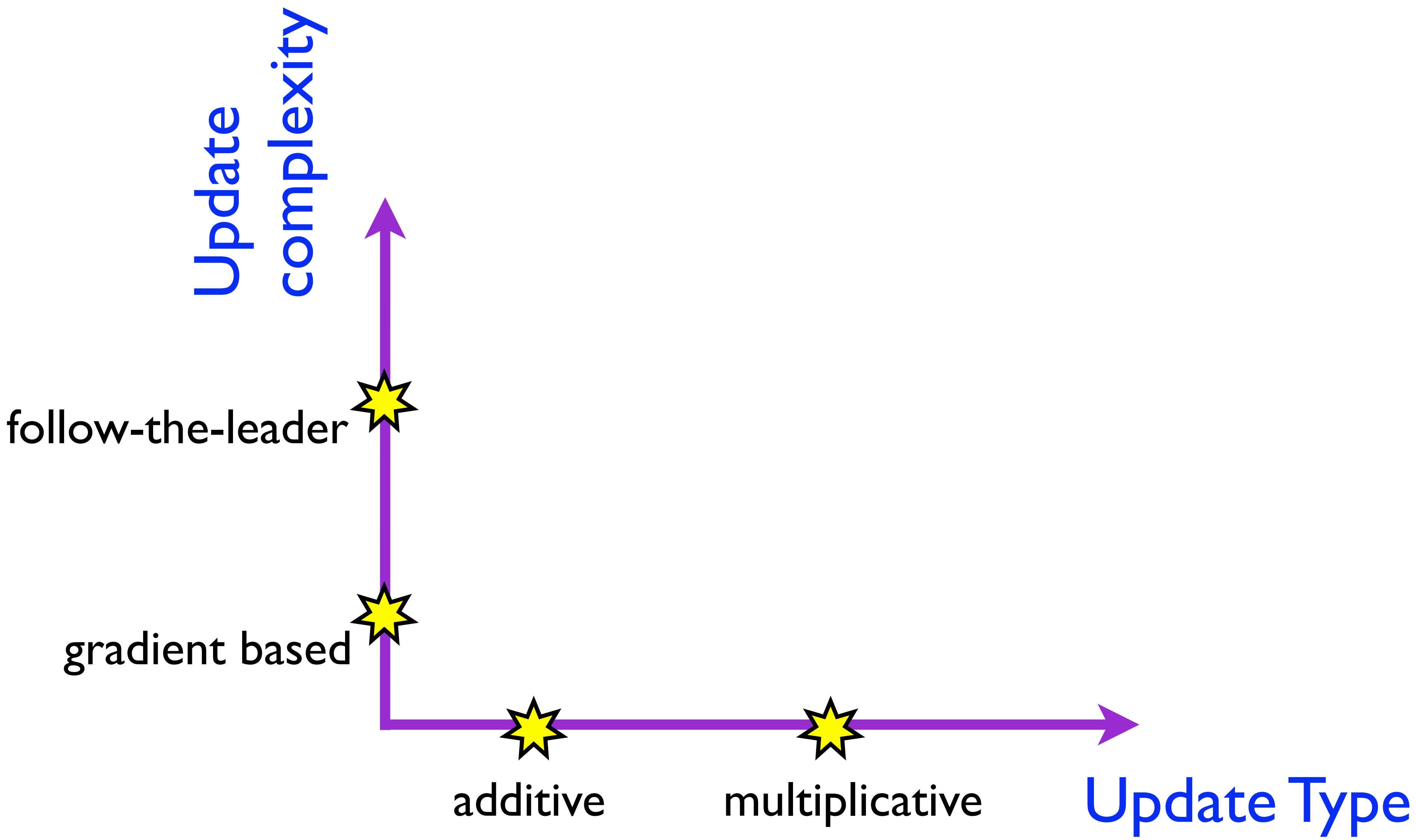
Tutorial's goals: provide design and analysis tools for online algorithms

Part II:
An algorithmic framework for online convex optimization

Outline

Tutorial's goals: provide design and analysis tools for online algorithms

Part II:
An algorithmic framework for online convex optimization



Outline

Tutorial's goals: provide design and analysis tools for online algorithms

Part III:
Derived algorithms

- Perceptrons (aggressive, conservative)
- Passive-Aggressive algorithms for the hinge-loss
- Follow the regularized leader (online SVM)
- Prediction with expert advice using multiplicative updates
- Online logistic regresssion with multiplicative updates

Outline

Tutorial's goals: provide design and analysis tools for online algorithms

Part IV:
Application - Mail filtering

- Algorithms derived from framework for online convex optimization:
  - Additive & multiplicative dual steppers
  - Aggressive update schemes: instantaneous dual maximizers
- Mail filtering by online multiclass categorization

Outline

Part V:
Not covered due to lack of time

- Improved algorithms and regret bounds:
  - Self-tuning
  - Logarithmic regret for strongly convex losses
- Other notions of regret: internal regret, drifting hypotheses
- Partial feedback: Bandit problems, Reinforcement learning
- Online-to-batch conversions

Problem I: Regression

Task: guess the next element of a real-valued sequence

Online Regression

For $t = 1, 2, \ldots$

- Predict a real number $\hat{y}_t \in \mathbb{R}$
- Receive $y_t \in \mathbb{R}$
- Suffer loss $(\hat{y}_t - y_t)^2$

What could constitute a good prediction strategy ?

Regression (cont.)

Follow-The-Leader

- Predict: $\hat{y}_t = \frac{1}{t-1}\sum_{i=1}^{t-1} y_t$
- Similar to Maximum Likelihood

Regret Analysis

- The FTL predictor satisfies:

$$\forall y^\star,\ \sum_{t=1}^{T}(\hat{y}_t - y_t)^2 - \sum_{t=1}^{T}(y^\star - y_t)^2 \leq O(\log(T))$$

- FTL is minimax optimal (outside scope)

Regression (cont.)

Proof Sketch

- Be-The-Leader: $\tilde{y}_t = \frac{1}{t}\sum_{i=1}^{t} y_t$
- The regret of BTL is at most 0 (elementary)
- FTL is close enough to BTL (simple algebra)

$$(\hat{y}_t - y_t)^2 - (\tilde{y}_t - y_t)^2 \leq O(\tfrac{1}{t})$$

- Summing over t (harmonic series) and we are done

Problem II: Classification

Guess the next element of a binary sequence

Online Prediction

For $t = 1, 2, \ldots$

- Predict a binary number $\hat{y}_t \in \{+1, -1\}$
- Receive $y_t \in \{+1, -1\}$
- Suffer $0 - 1$ loss

$$\ell(\hat{y}_t, y_t) = \begin{cases} 1 & \text{if } \; y_t \neq \hat{y}_t \\ 0 & \text{otherwise} \end{cases}$$

Classification (cont.)

No algorithm can guarantee low regret !

Proof Sketch

- Adversary can force the cumulative loss of the learner to be as large as T by using $y_t = -\hat{y}_t$
- The loss of the constant prediction
$$y^\star = \text{sign}\left(\sum_t y_t\right) \text{ is at most } T/2$$
- Regret is at least $T/2$

Intermediate Conclusion

- Two similar problems
  - Predict the next real-valued element with squared loss ✓
  - Predict the next binary-valued element with 0-1 loss ✗
- Size of decision set does not matter !
- In the first problem, loss is convex and decision set is convex
- Is convexity sufficient for predictability ?

Online Convex Optimization

- Abstract game between learner and environment
- Game board is a convex set S
- Learner plays with vectors in S
- Environment plays with convex functions over S

Online Convex Optimization

For $t = 1, 2, \ldots, T$

- Learner picks $\mathbf{w}_t \in S$
- Environment responds with convex loss $\ell_t : S \to \mathbb{R}$
- Learner suffers loss $\ell_t(\mathbf{w}_t)$

Online Convex Optimization – Example I

Regression

- $S = \mathbb{R}$
- Learner predicts element $\hat{y}_t = w_t \in S$
- A true target $y_t \in \mathbb{R}$ defines a loss function $\ell_t(w) = (w - y_t)^2$

Online Convex Optimization – Example II

Regression with Experts Advice

- $S = \{\mathbf{w} \in \mathbb{R}^d : w_i \geq 0,\ \|\mathbf{w}\|_1 = 1\}$
- Learner picks $\mathbf{w}_t \in S$
- Learner predicts $\hat{y}_t = \langle \mathbf{w}_t, \mathbf{x}_t \rangle$
- A pair $(\mathbf{x}_t, y_t)$ defines a loss function over S: $\ell_t(\mathbf{w}) = (\langle \mathbf{w}, \mathbf{x}_t \rangle - y_t)^2$

Coping with Non-convex Loss Functions

- Method I: Convexification
  - Find a surrogate convex loss function
  - Mistake bound model
- Method II: Randomization
  - Allow randomized predictions
  - Analyzed expected regret
  - Loss in expectation is convex

Convexification and Mistake Bound

- Non-convex loss: mistake indicator a.k.a 0-1 loss

$$\ell_{0-1}(\hat{y}_t, y_t) \;=\; \begin{cases} 1 & \text{if} \;\; y_t \neq \hat{y}_t \\ 0 & \text{otherwise} \end{cases}$$

- Recall that regret can be as large as T/2

- Surrogate loss function: hinge-loss

$$\ell_{\text{hi}}(\mathbf{w}, (\mathbf{x}_t y_t)) \;=\; [1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle]_+$$

where

$$[a]_+ = \max\{a, 0\}$$

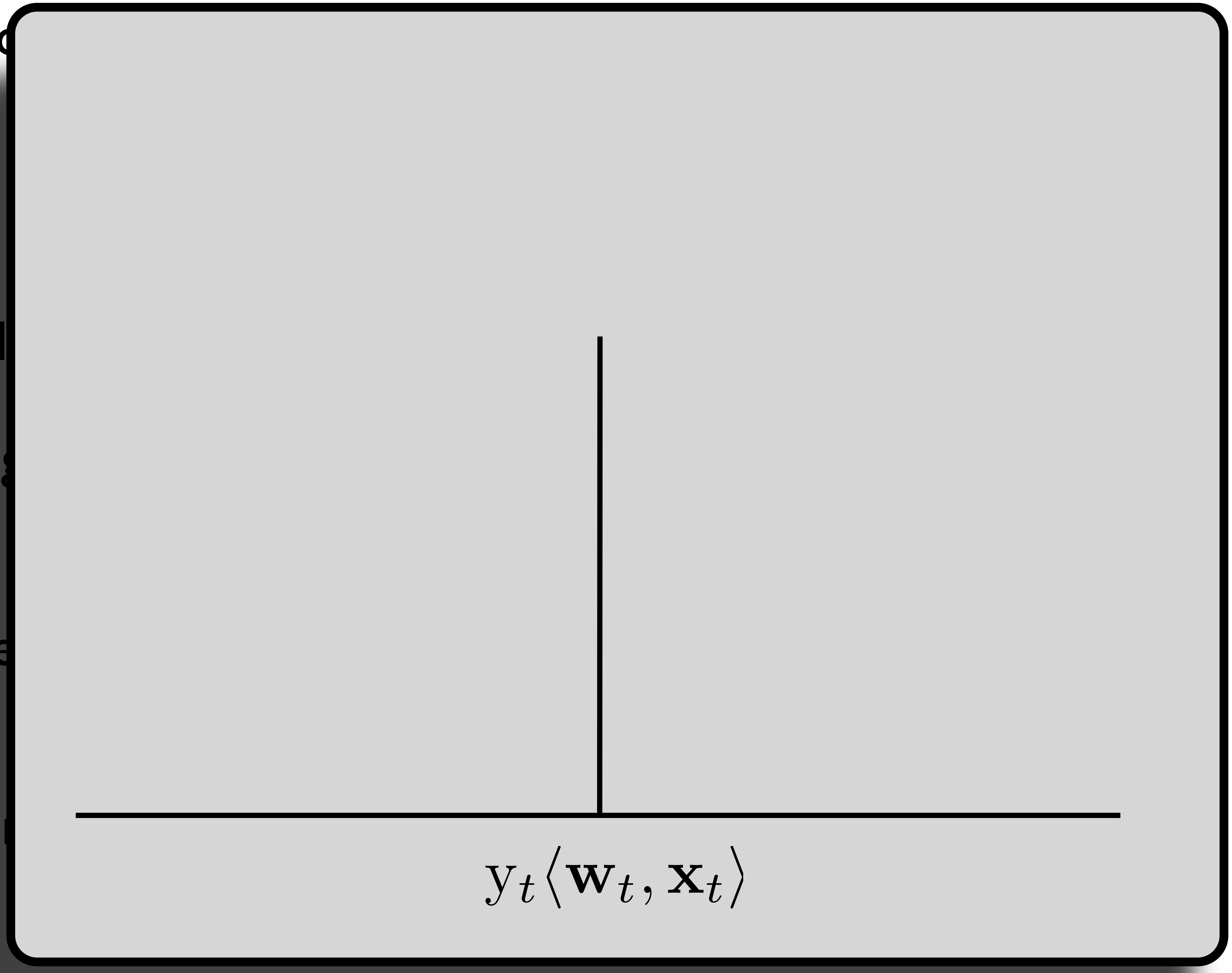
- By construction $\ell_{0-1}(\hat{y}_t, y_t) \;\leq\; \ell_{\text{hi}}(\mathbf{w}_t, (\mathbf{x}_t y_t))$

Convexification and Mistake Bound

- Non-c
- Recall
- Surrog

  where

- By co

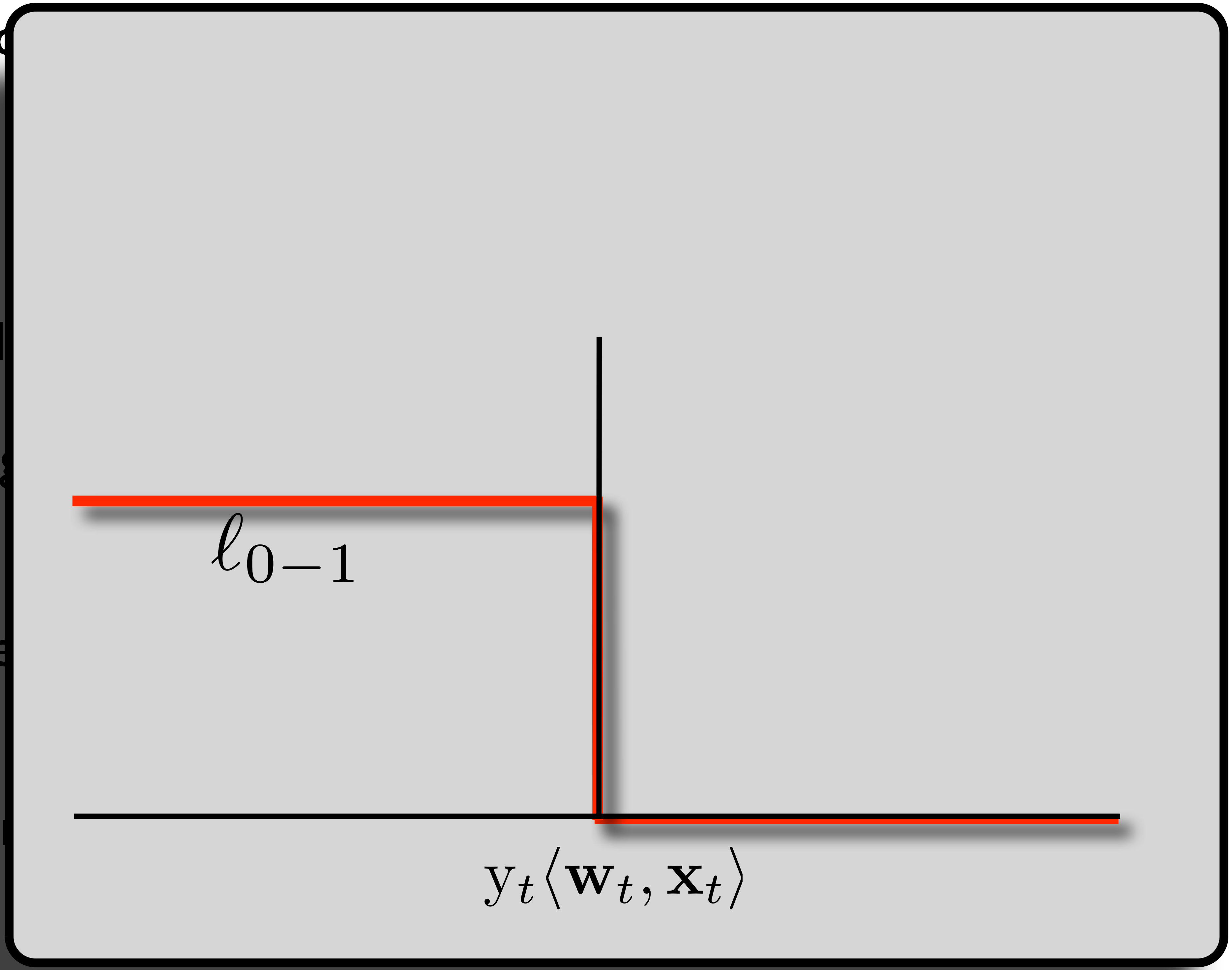


Convexification and Mistake Bound

- Non-c
- Recall
- Surrog

  where

- By co

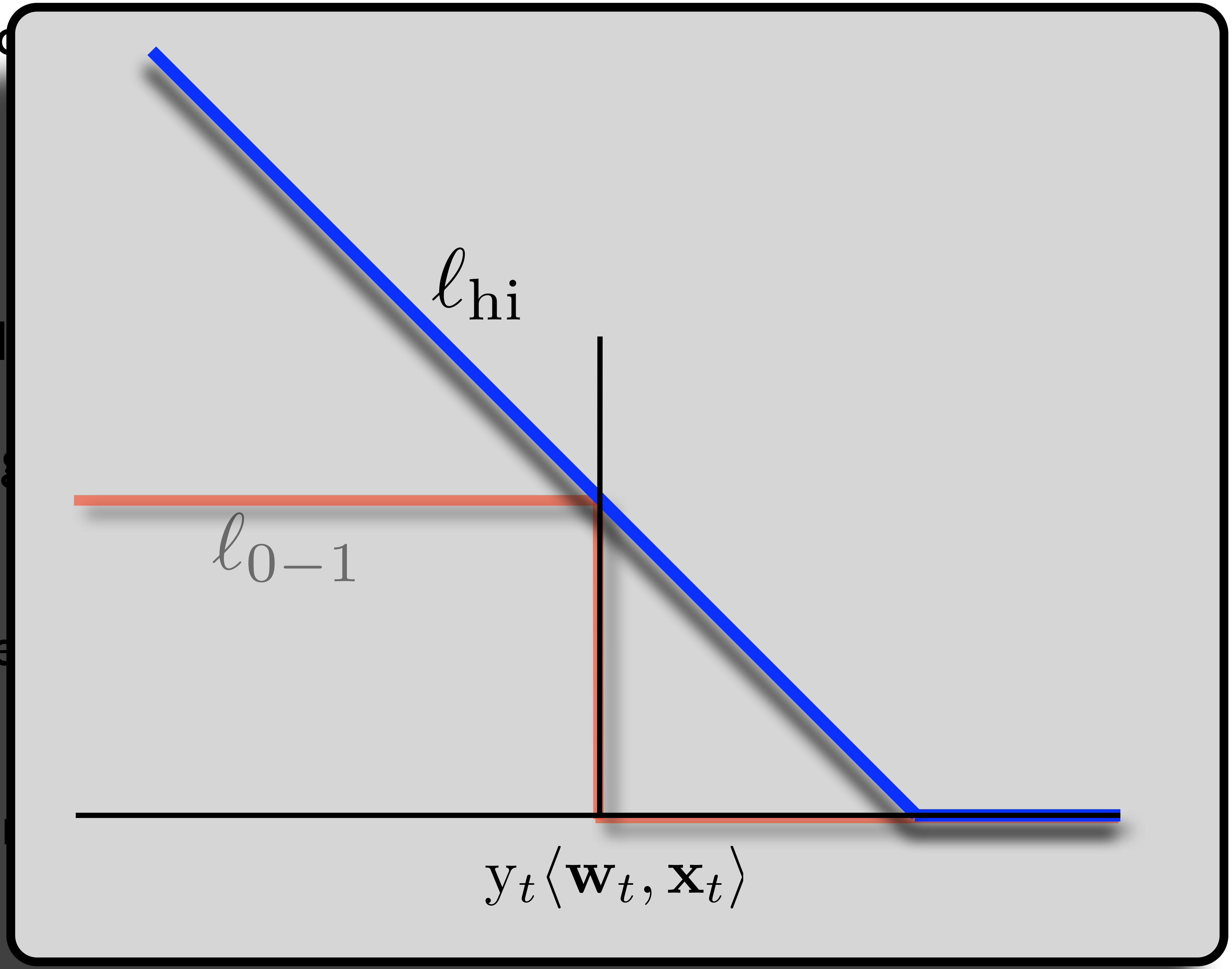


Convexification and Mistake Bound

- Non-c
- Recall
- Surrog

  where

- By co



Randomization and Expected Regret

Example – Classification with Expert Advice

- Learner receives expert advice $\mathbf{x}_t \in [0, 1]^d$
- Should predict $\hat{y}_t \in \{+1, -1\}$
- Receive $y_t \in \{+1, -1\}$
- Suffer $0 - 1$ loss $\ell_{0-1}(\hat{y}_t, y_t) \ = \ 1 - \delta(y_t, \hat{y}_t)$

Convexify by randomization:

- Learner picks $\mathbf{w}_t$ in d-dim probability simplex
- Predict $\hat{y}_t = 1$ with probability $\langle \mathbf{w}_t, \mathbf{x}_t \rangle$
- Expected $0 - 1$ loss is convex w.r.t. $\mathbf{w}_t$

$$\mathbb{E}[\hat{y}_t \neq y_t] = \tfrac{y_t+1}{2} - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle$$

Part II:

An Algorithmic Framework for Online Convex Optimization

Online Learning with the Perceptron

Get a PhD in 3 month! A better job, more income and a better life can all be yours. No books to buy, no classes to go …

Spam ?

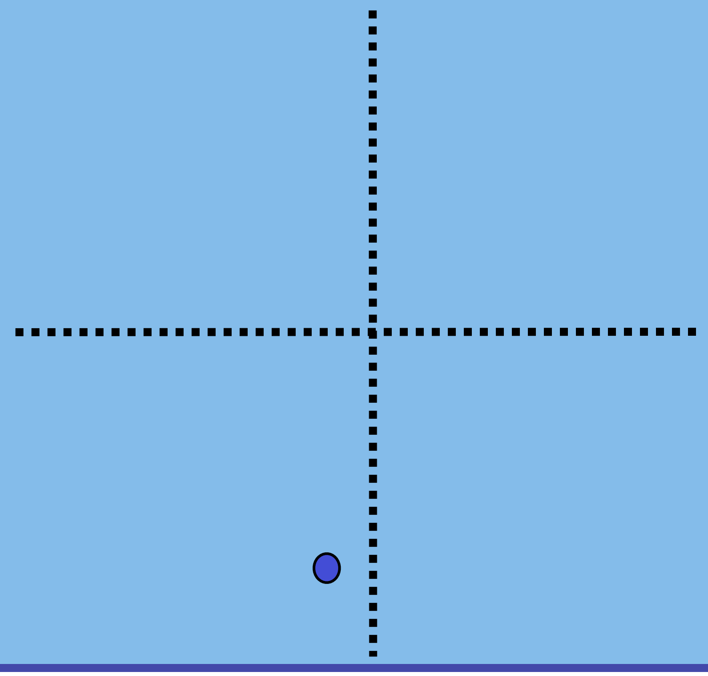
Online Learning with the Perceptron

Get a PhD in 3 month! A better job, more income and a better life can all be yours. No books to buy, no classes to go …

Spam ?

Perceptron (Rosenblatt58)

- emails encoded as vectors



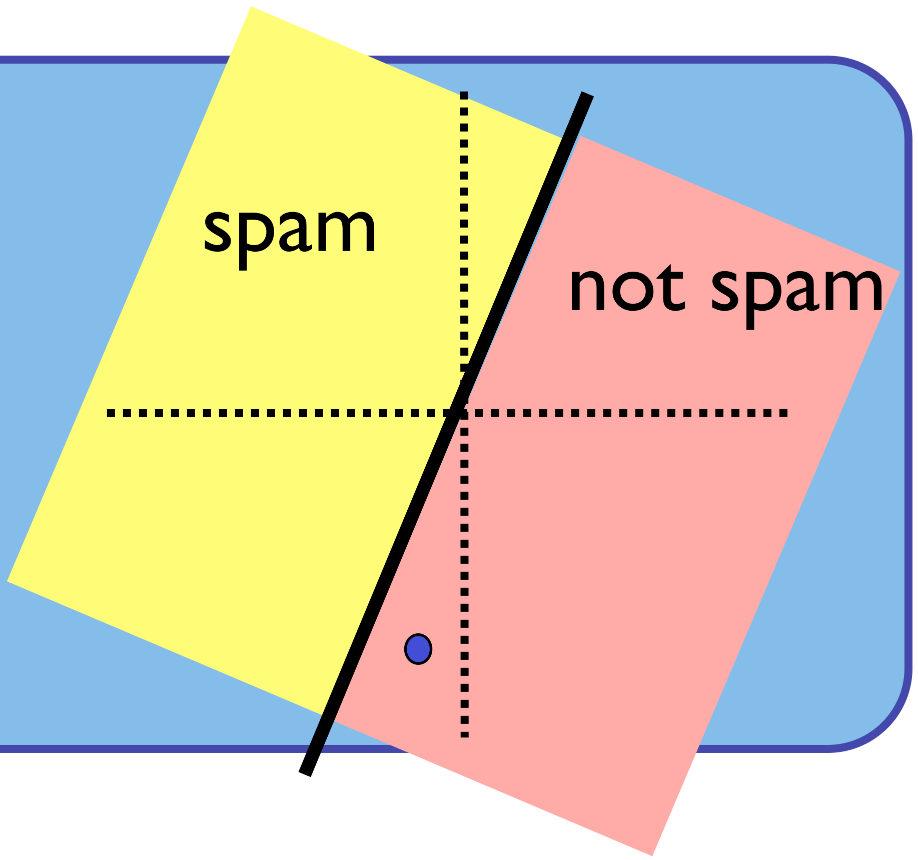
Online Learning with the Perceptron

Get a PhD in 3 month! A better job, more income and a better life can all be yours. No books to buy, no classes to go …

Spam ?

Perceptron (Rosenblatt58)

- emails encoded as vectors
- hypothesis - linear separator



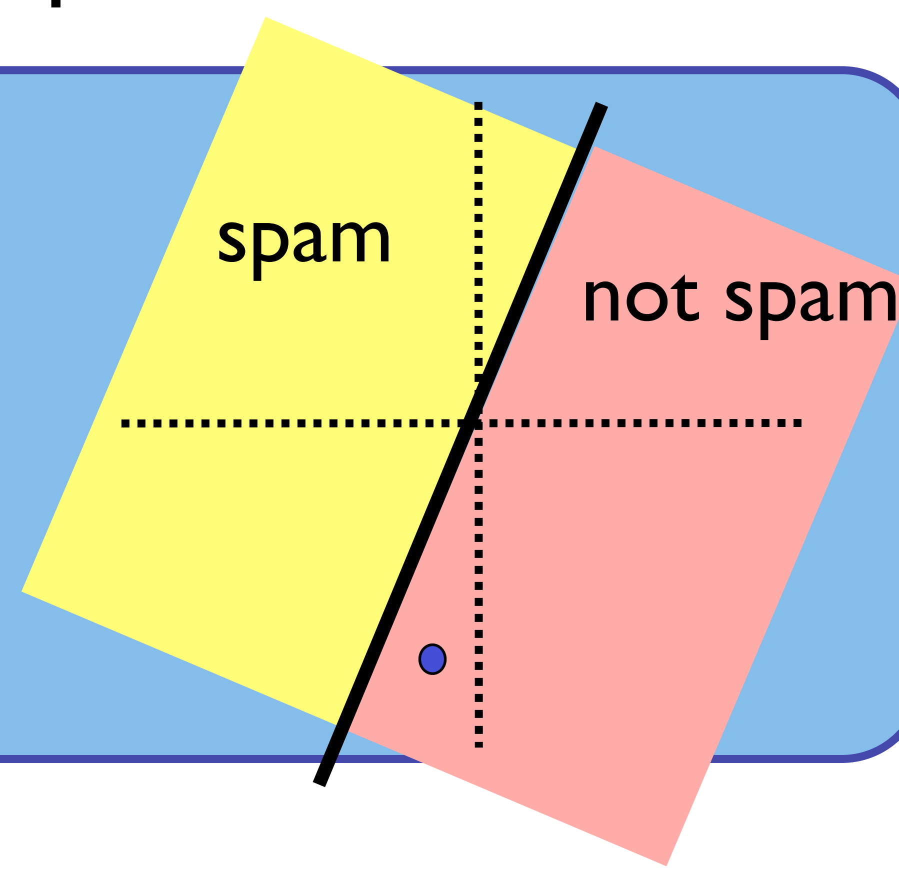
Online Learning with the Perceptron

Get a PhD in 3 month! A better job, more income and a better life can all be yours. No books to buy, no classes to go …

Spam ?

Perceptron (Rosenblatt58)

- emails encoded as vectors
- hypothesis - linear separator



No spam

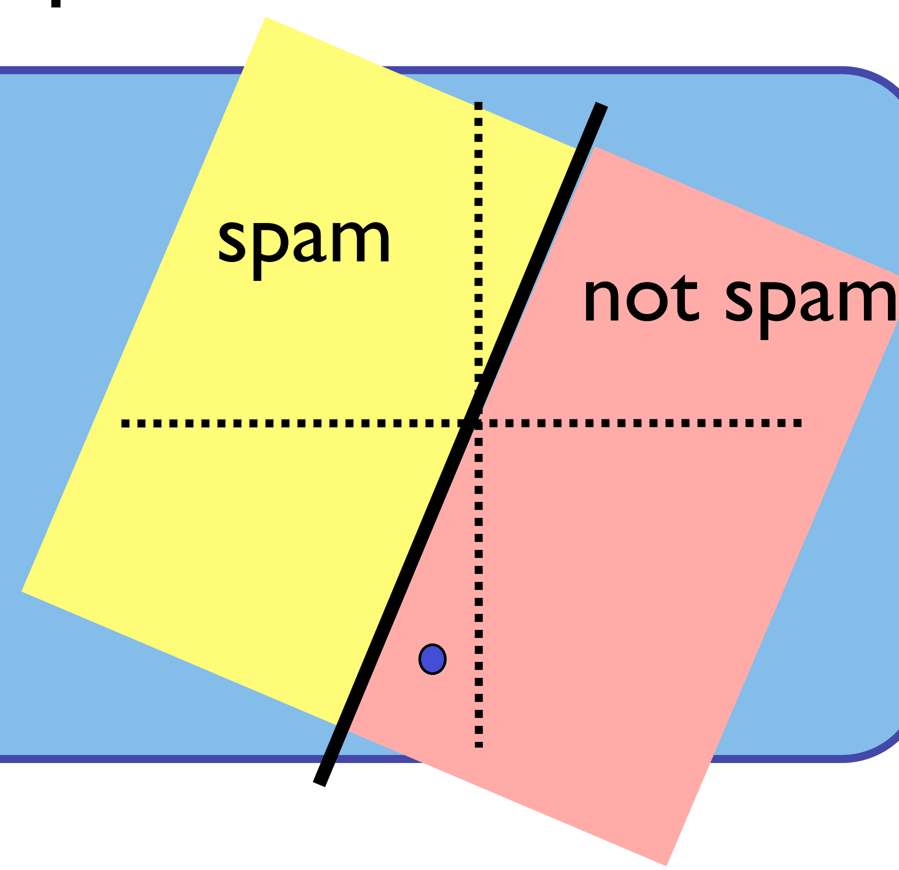
Online Learning with the Perceptron

Get a PhD in 3 month! A better job, more income and a better life can all be yours. No books to buy, no classes to go …

Spam ?

Perceptron (Rosenblatt58)

- emails encoded as vectors
- hypothesis - linear separator



Spam !

Feedback

No spam

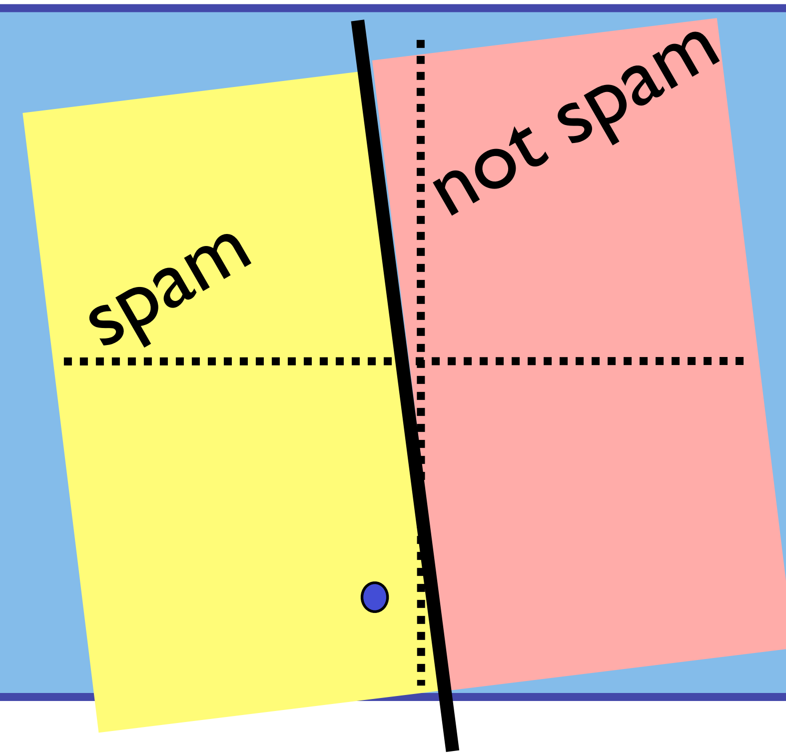
Online Learning with the Perceptron

Get a PhD in 3 month! A better job, more income and a better life can all be yours. No books to buy, no classes to go …

Spam ?

Perceptron (Rosenblatt58)

- emails encoded as vectors
- hypothesis - linear separator
- update: $\mathbf{w} \leftarrow \mathbf{w} + y\mathbf{x}$



Spam !

Feedback

No spam

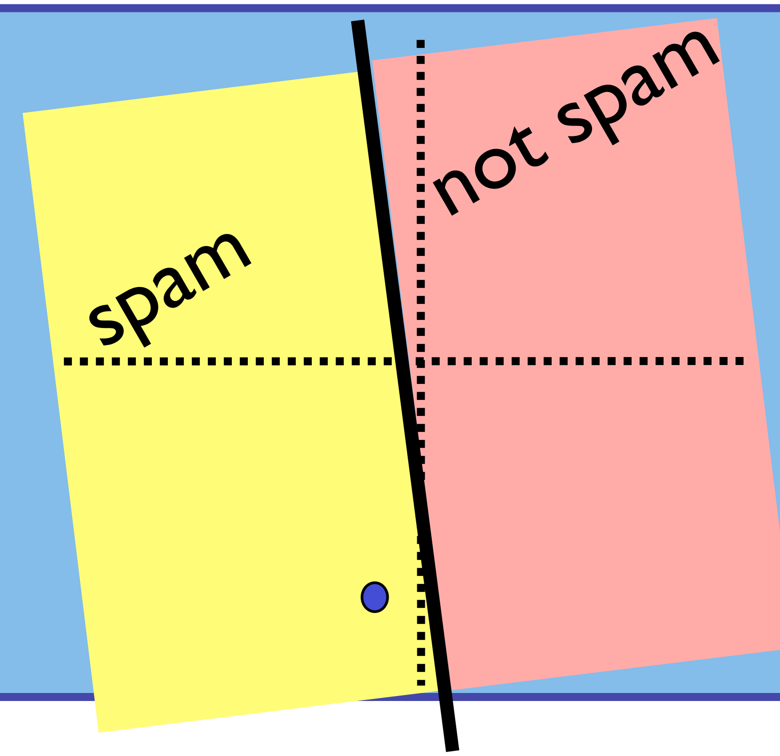
Online Learning with the Perceptron

Get a PhD in 3 month! A better job, more income and a better life can all be yours. No books to buy, no classes to go …

Spam ?

Perceptron (Rosenblatt58)

- emails encoded as vectors
- hypothesis - linear separator
- update: $\mathbf{w} \leftarrow \mathbf{w} + y\mathbf{x}$



Spam !

Feedback

No spam

update if $y\langle \mathbf{w}, \mathbf{x} \rangle < 1$

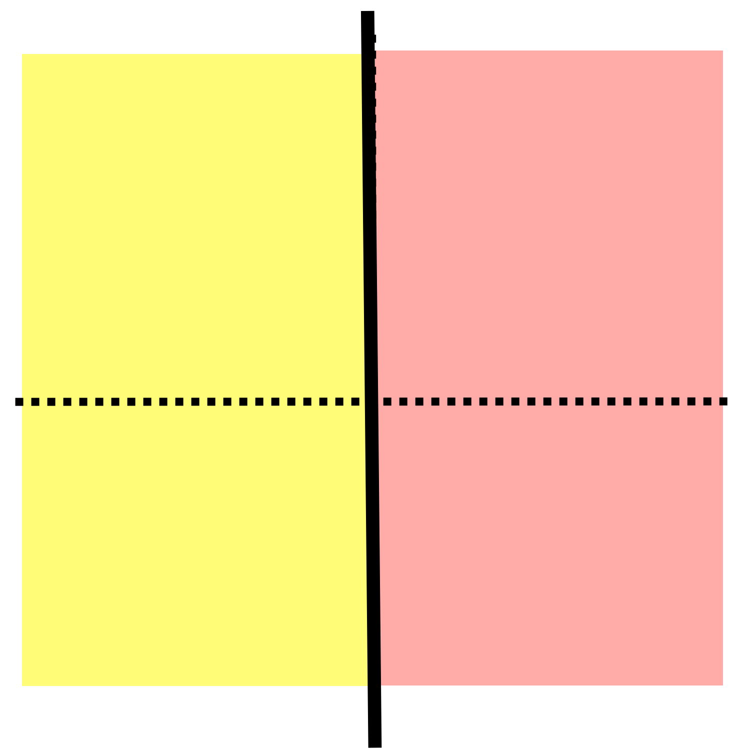
(aggressive perceptron)

Regret

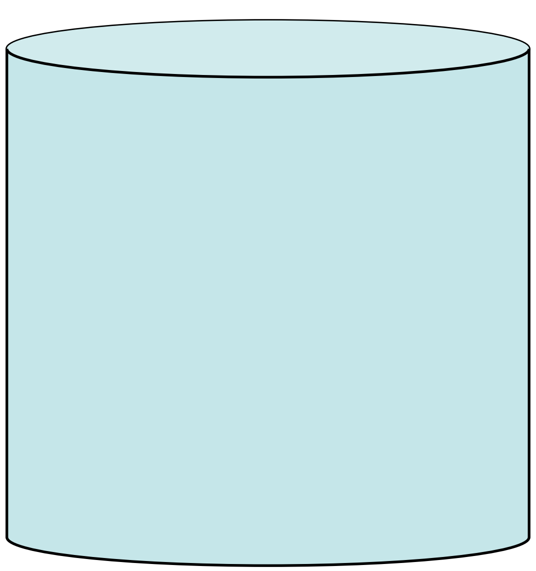
spam

not spam

Learner   Environment   Loss



Loss
of learner

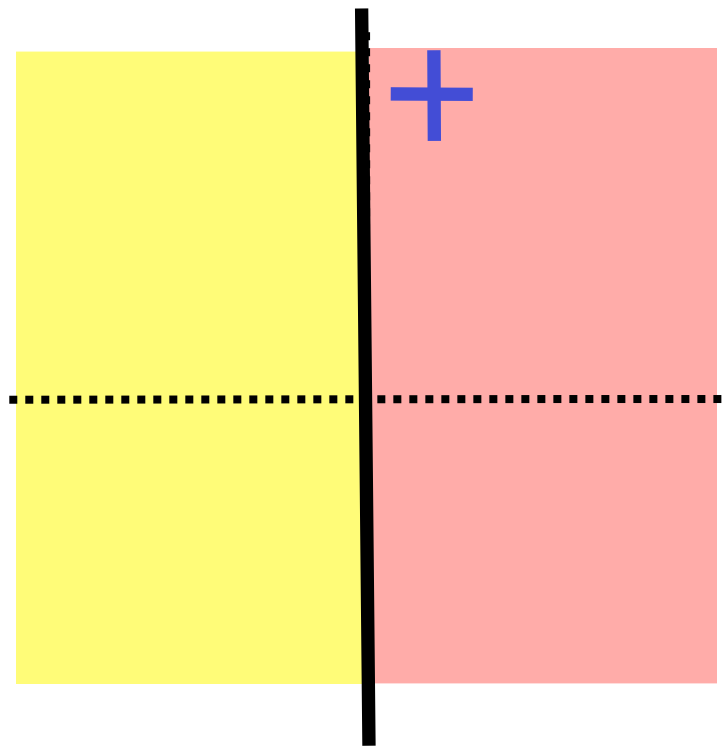


Regret

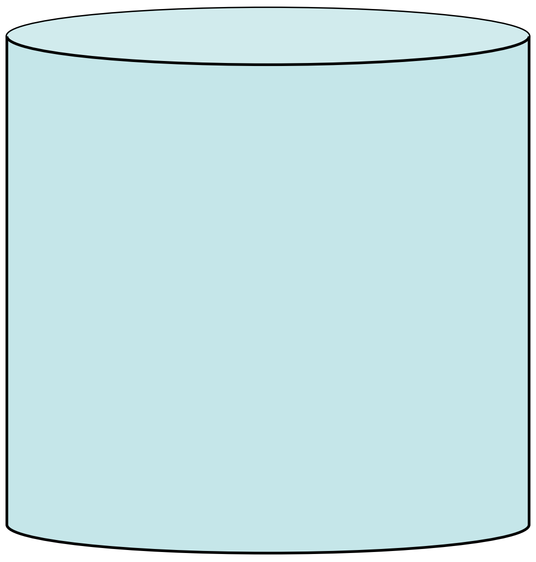
spam

not spam

Learner Environment Loss



Loss
of learner

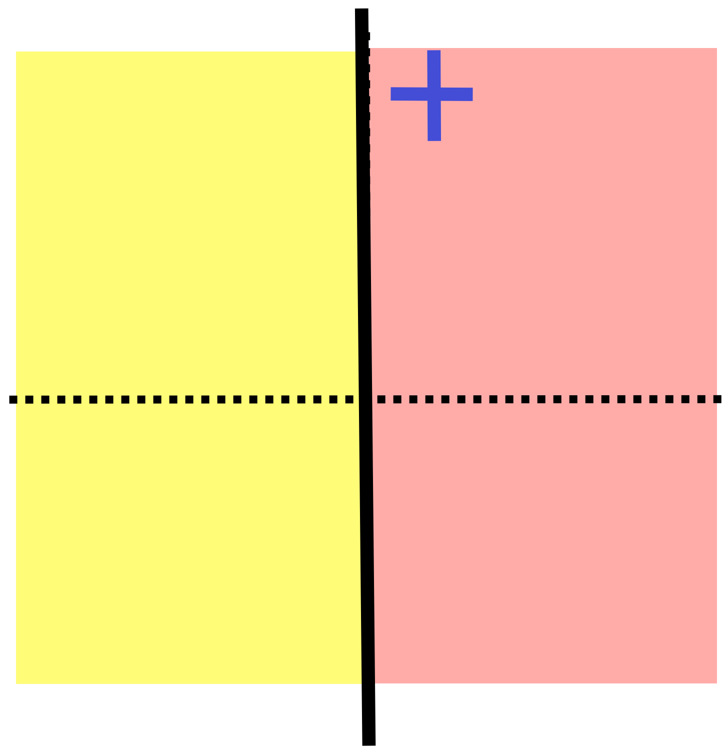


Regret

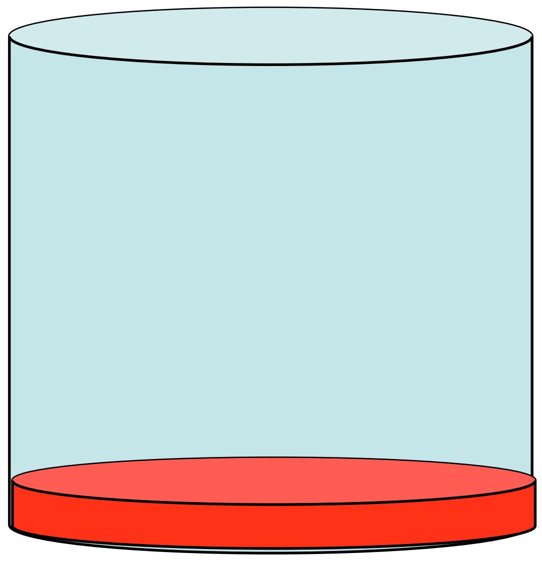
spam

not spam

Learner Environment Loss



Loss
of learner

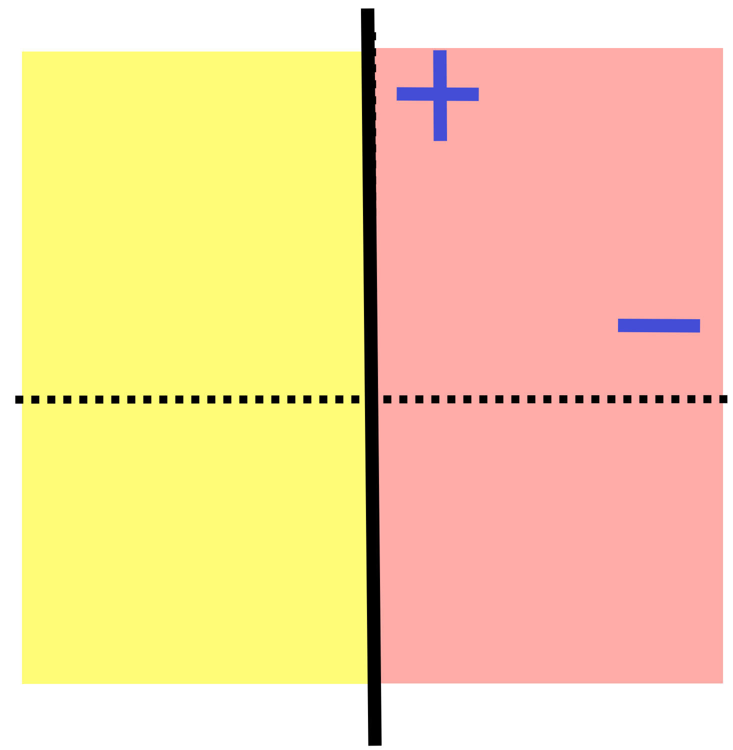


Regret

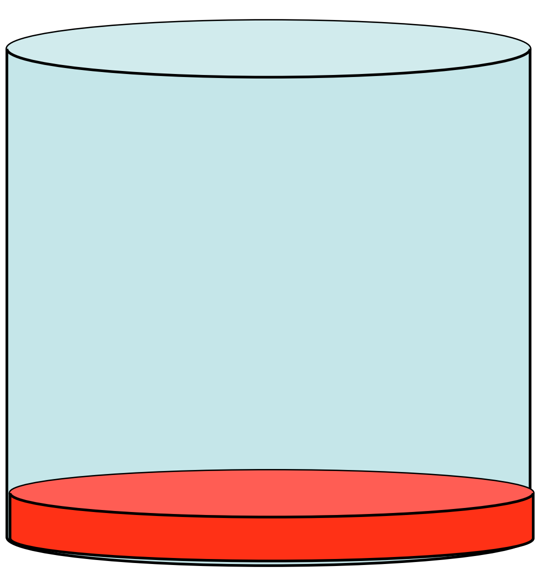
spam

not spam

Learner Environment Loss



Loss
of learner

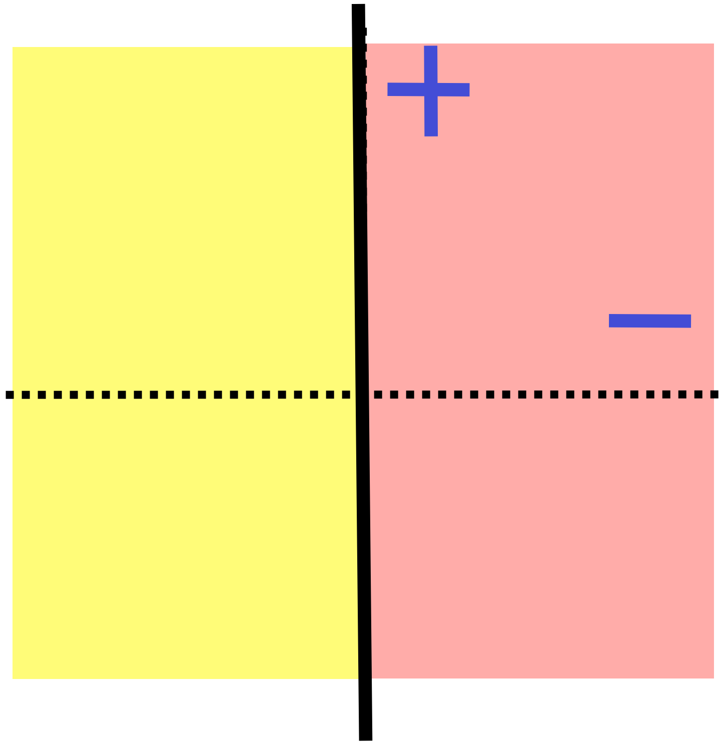


Regret

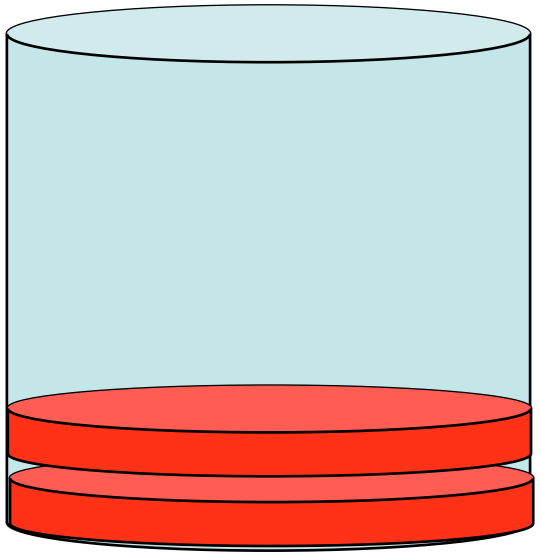
spam

not spam

Learner Environment Loss



Loss
of learner

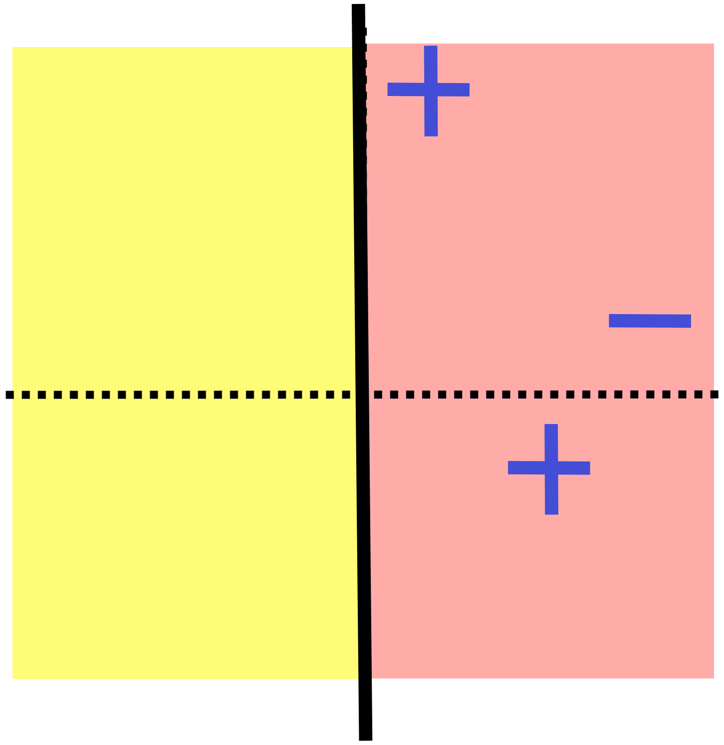


Regret

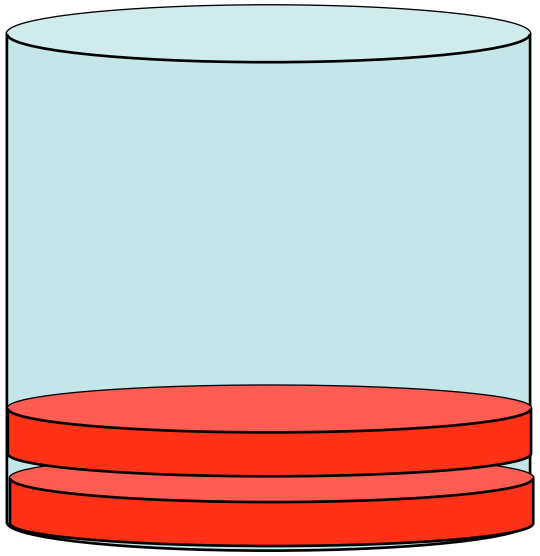
spam

not spam

Learner Environment Loss



Loss
of learner

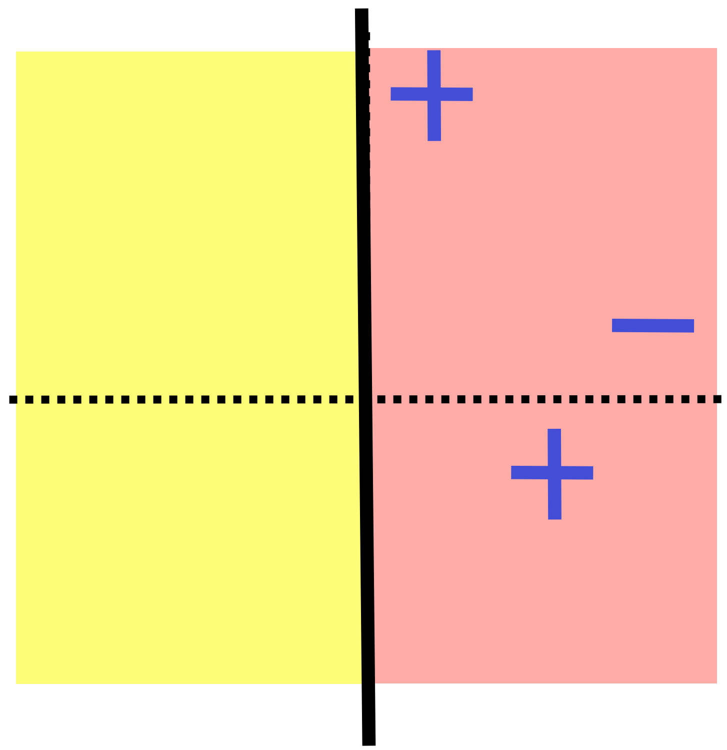


Regret

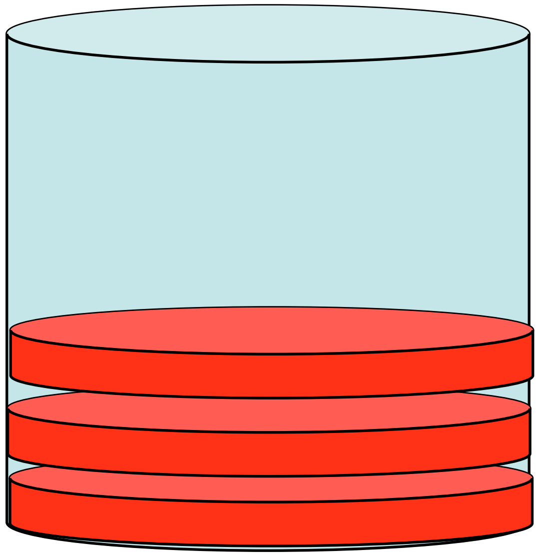
spam

not spam

Learner Environment Loss



Loss
of learner

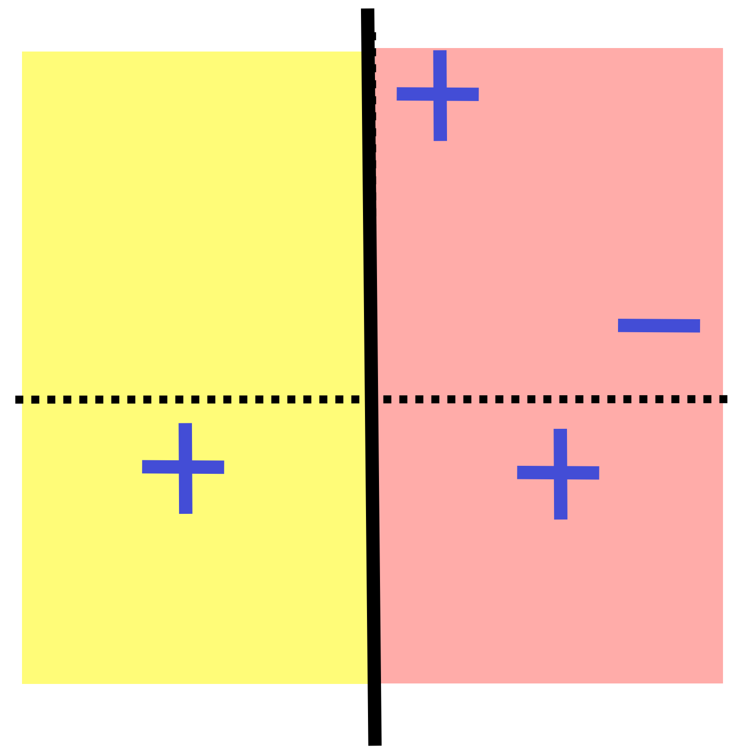


Regret

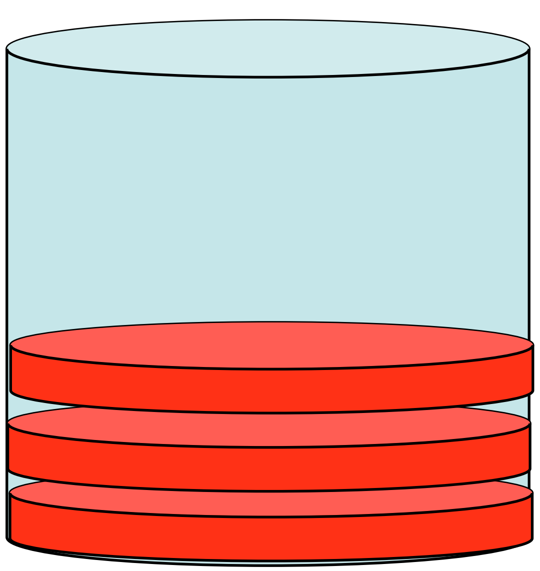
spam

not spam

Learner Environment Loss



Loss
of learner

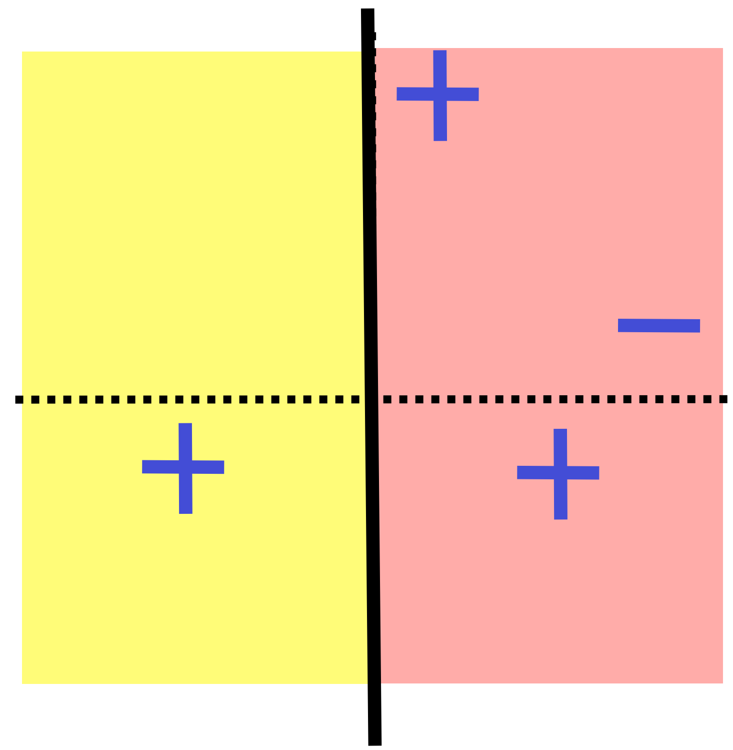


Regret

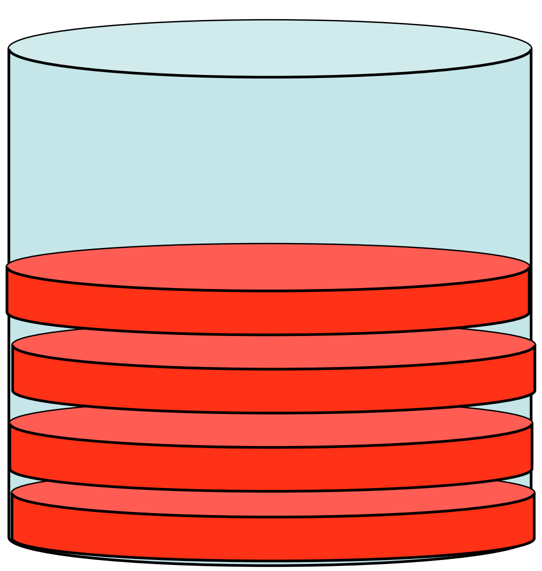
spam

not spam

Learner Environment Loss



Loss
of learner

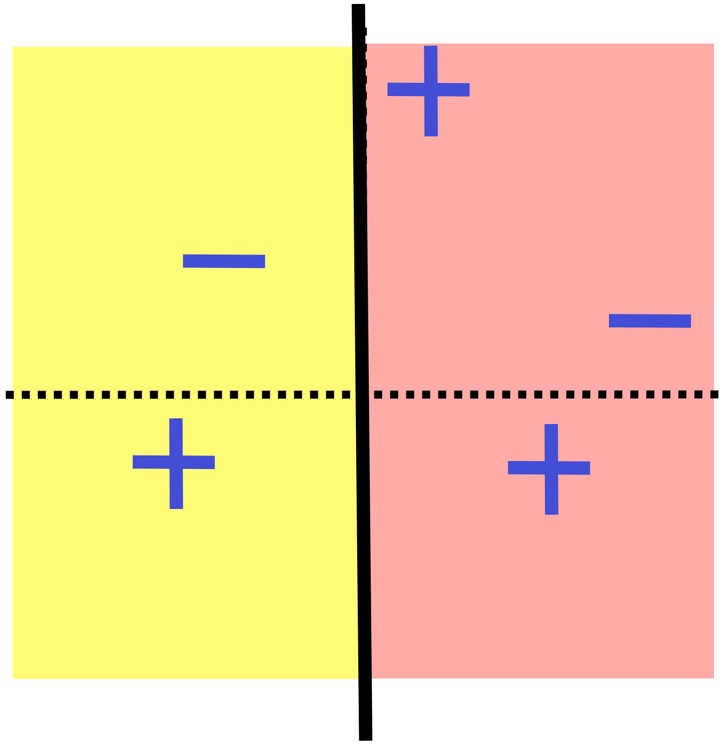


Regret

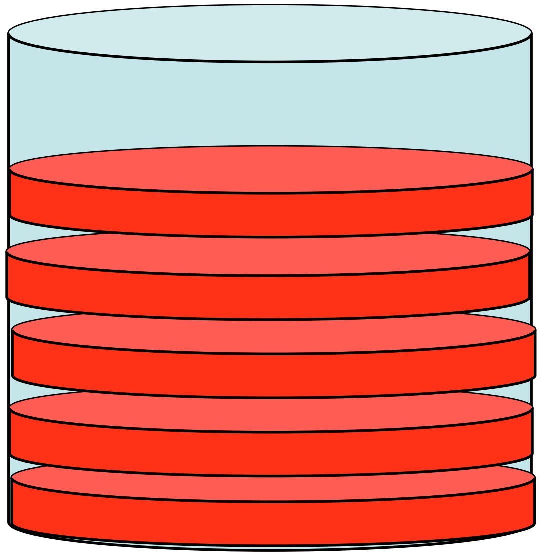
spam

not spam

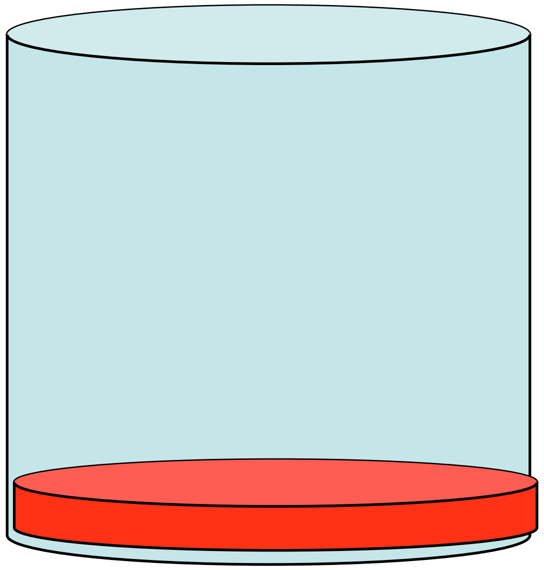
Learner   Environment   Loss



Loss
of learner



Best Loss
in hindsight



More Stringent Form of Regret

- Original regret goal:

$$\sum_{t=1}^{T} \ell_{hi}(\mathbf{w}_t, (\mathbf{x}_t, y_t)) \leq \min_{\mathbf{w}:\|\mathbf{w}\|\leq D} \sum_{t=1}^{T} \ell_{hi}(\mathbf{w}, (\mathbf{x}_t, y_t)) + o(T)$$

- A stronger requirement:

$$\sum_{t=1}^{T} \ell_t(\mathbf{w}_t) \leq \min_{\mathbf{w}} \frac{\sigma}{2}\|\mathbf{w}\|^2 + \sum_{t=1}^{T} \ell_{hi}(\mathbf{w}, (\mathbf{x}_t, y_t))$$

From Regret to SVM

- Rewriting $\ell_{hi}(\cdot)$

$$\xi_t = \ell_{hi}(\mathbf{w}_t, (\mathbf{x}_t, y_t)) \;\Rightarrow\; \xi_t \geq 0 \;\wedge\; \xi_t \geq 1 - y_t\langle \mathbf{w}_t, \mathbf{x}_t \rangle$$

- The target regret

$$\min_{\mathbf{w}} \;\frac{\sigma}{2}\|\mathbf{w}\|^2 + \sum_{t=1}^{T} \ell_{hi}(\mathbf{w}, (\mathbf{x}_t, y_t))$$

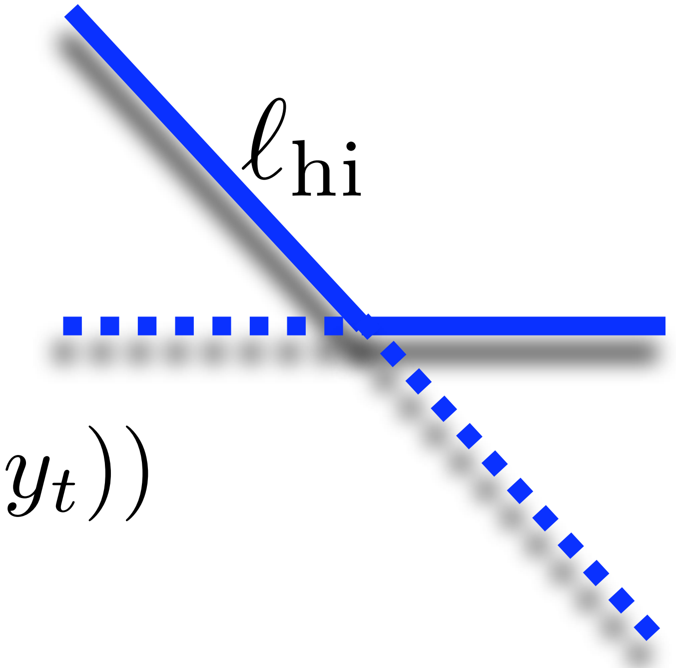
can be rewritten as

$$\min_{\mathbf{w}, \boldsymbol{\xi} \succeq \mathbf{0}} \;\frac{\sigma}{2}\|\mathbf{w}\|^2 + \sum_{t=1}^{T} \xi_t \quad \text{s.t.} \quad \xi_t \geq 1 - y_t\langle \mathbf{w}_t, \mathbf{x}_t \rangle$$

From Regret to SVM

- Rewriting $\ell_{hi}(\cdot)$

$$\xi_t = \ell_{hi}(\mathbf{w}_t, (\mathbf{x}_t, y_t)) \;\Rightarrow\; \xi_t \geq 0 \;\wedge\; \xi_t \geq 1 - y_t\langle \mathbf{w}_t, \mathbf{x}_t \rangle$$



- The target regret

$$\min_{\mathbf{w}} \; \frac{\sigma}{2}\|\mathbf{w}\|^2 + \sum_{t=1}^{T} \ell_{hi}(\mathbf{w}, (\mathbf{x}_t, y_t))$$

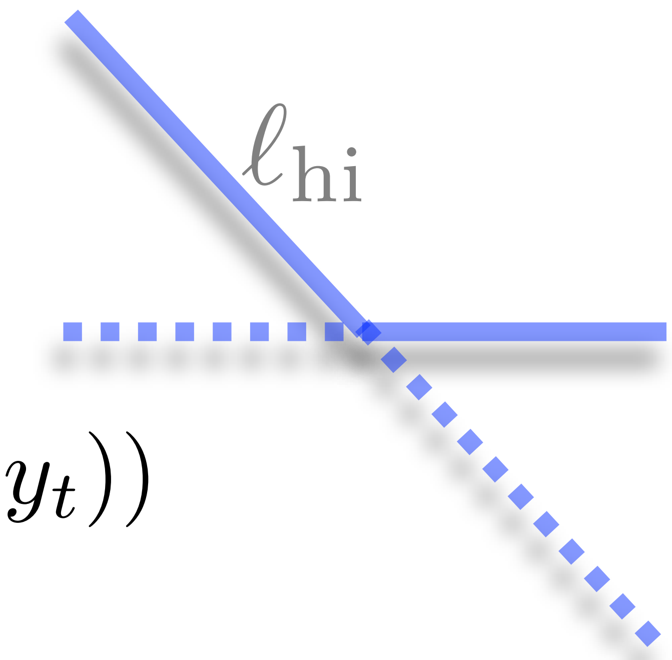
can be rewritten as

$$\min_{\mathbf{w}, \boldsymbol{\xi} \succeq \mathbf{0}} \; \frac{\sigma}{2}\|\mathbf{w}\|^2 + \sum_{t=1}^{T} \xi_t \;\; \text{s.t.} \;\; \xi_t \geq 1 - y_t\langle \mathbf{w}_t, \mathbf{x}_t \rangle$$

From Regret to SVM

- Rewriting $\ell_{hi}(\cdot)$

$$\xi_t = \ell_{hi}(\mathbf{w}_t, (\mathbf{x}_t, y_t)) \;\Rightarrow\; \xi_t \geq 0 \,\wedge\, \xi_t \geq 1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle$$



- The target regret

$$\min_{\mathbf{w}} \; \frac{\sigma}{2}\|\mathbf{w}\|^2 + \sum_{t=1}^{T} \ell_{hi}(\mathbf{w}, (\mathbf{x}_t, y_t))$$

can be rewritten as

$$\underbrace{\min_{\mathbf{w}, \boldsymbol{\xi} \succeq \mathbf{0}} \; \frac{\sigma}{2}\|\mathbf{w}\|^2 + \sum_{t=1}^{T} \xi_t \;\; \text{s.t.} \;\; \xi_t \geq 1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle}_{\text{SVM Objective}}$$

Regret and Duality

The loss of Perceptron should be smaller than SVM objective

SVM duality

- Primal SVM: $\mathcal{P}(\mathbf{w}) = \frac{\sigma}{2}\|\mathbf{w}\|^2 + \sum_{t=1}^{T} \ell_{hi}(\mathbf{w}, (\mathbf{x}_t, y_t))$
- Constrained form

$$\min_{\mathbf{w}, \xi \geq 0} \frac{\sigma}{2}\|\mathbf{w}\|^2 + \sum_{t=1}^{T} \xi_t \quad \text{s.t.} \quad 1 - y_t\langle \mathbf{w}, \mathbf{x}_t \rangle \leq \xi_t$$

- Dual objective $\mathcal{D}(\boldsymbol{\alpha}) = \sum_t \alpha_t - \frac{1}{2\,\sigma}\left\|\sum_t \alpha_t y_t \mathbf{x}_t\right\|^2$

Properties of Dual Problem

$$\mathcal{D}(\boldsymbol{\alpha}) \;=\; \sum_{t=1}^{T} \alpha_t - \frac{1}{2\,\sigma} \left\| \sum_{t=1}^{T} \alpha_t y_t \mathbf{x}_t \right\|^2$$

- Dedicated variable for each online round
- If $\alpha_t = \ldots = \alpha_T = 0$ then $\mathcal{D}(\boldsymbol{\alpha})$ can be optimized without the knowledge of $(\mathbf{x}_t, y_t), \ldots, (\mathbf{x}_T, y_T)$
- $\mathcal{D}(\boldsymbol{\alpha})$ can be optimized along the online process
- Weak Duality $\max_{\boldsymbol{\alpha} \in [0,1]^m} \mathcal{D}(\boldsymbol{\alpha}) \;\leq\; \min_{\mathbf{w}} \mathcal{P}(\mathbf{w})$
- Core idea:
  Online learning by incremental dual ascent

Properties of Dual Problem

$$\mathcal{D}(\boldsymbol{\alpha}) = \sum_{t=1}^{T} \alpha_t - \frac{1}{2\,\sigma} \left\| \sum_{t=1}^{T} \alpha_t y_t \mathbf{x}_t \right\|^2$$

- Dedicated variable for each online round
- If $\alpha_t = \ldots = \alpha_T = 0$ then $\mathcal{D}(\boldsymbol{\alpha})$ can be optimized without the knowledge of $(\mathbf{x}_t, y_t), \ldots, (\mathbf{x}_T, y_T)$
- $\mathcal{D}(\boldsymbol{\alpha})$ can be optimized along the online process
- Weak Duality $\boxed{\max_{\boldsymbol{\alpha} \in [0,1]^m} \mathcal{D}(\boldsymbol{\alpha}) \le \min_{\mathbf{w}} \mathcal{P}(\mathbf{w})}$ $\Rightarrow$ Key Analysis Tool
- Core idea:
  Online learning by incremental dual ascent

Online Learning by Dual Ascent

Abstract Dual Ascent Learner

- Initialize $\alpha_1 = \ldots = \alpha_T = 0$
- For $t = 1, 2, \ldots, T$
  - Construct $\mathbf{w}_t$ from dual variables (how ?)
  - Receive $(\mathbf{x}_t, y_t)$ from environment
  - Inform dual optimizer of new example
  - Obtain α_t from dual optimizer

Online Learning by Dual Ascent

Lemma

- Let $\mathcal{D}_t$ be the dual value at round t
- Let $\Delta_t = \mathcal{D}_{t+1} - \mathcal{D}_t$ be the dual increase
- Assume that $\Delta_t \geq \ell(\mathbf{w}_t, (\mathbf{x}_t, y_t)) - \frac{1}{2\,\sigma}$
- Then,

$$\sum_{t=1}^{T} \ell(\mathbf{w}_t, (\mathbf{x}_t, y_t)) - \sum_{t=1}^{T} \ell(\mathbf{w}^\star, (\mathbf{x}_t, y_t)) \ \leq \ O(\sqrt{T})$$

Online Learning by Dual Ascent

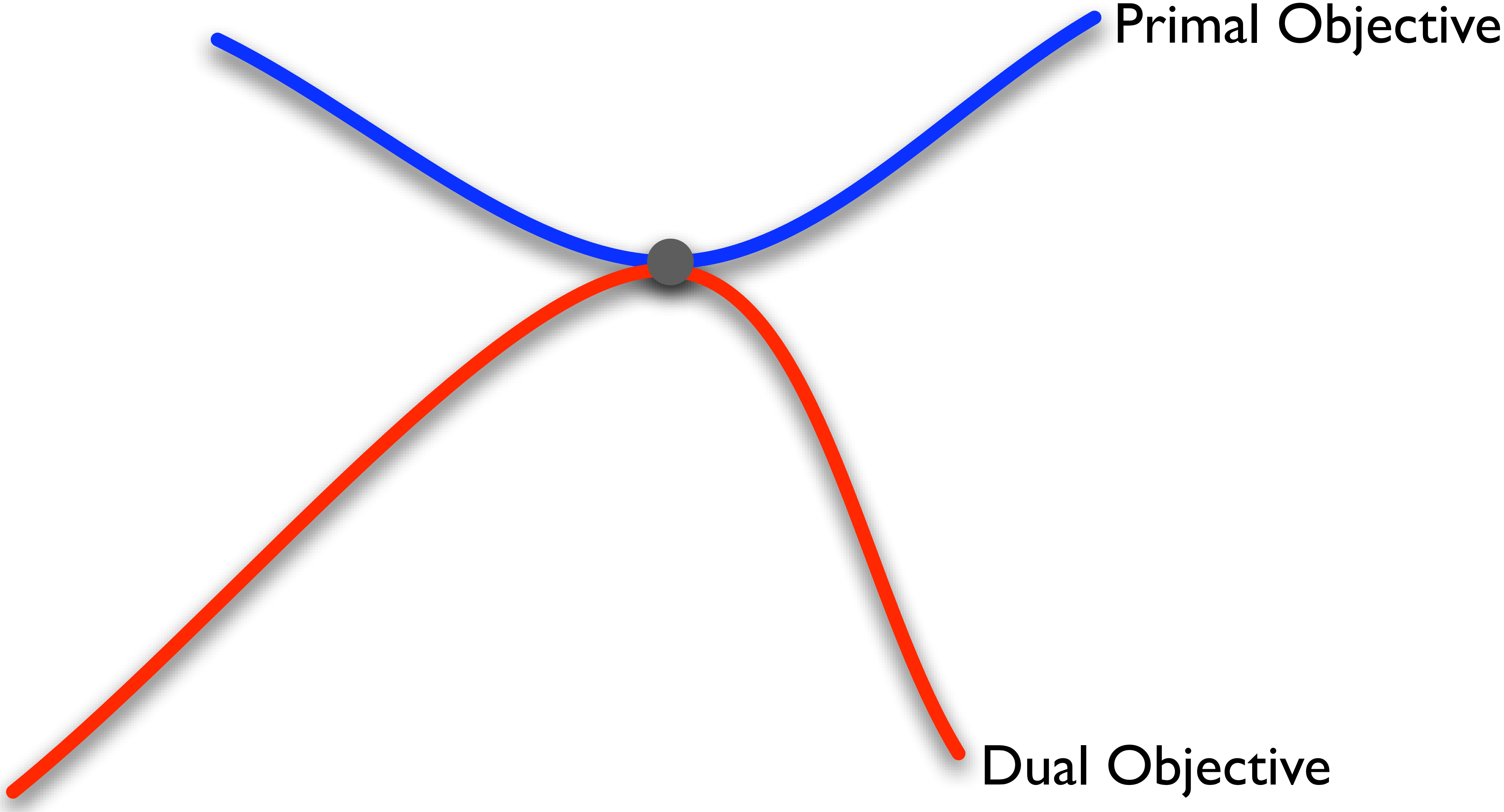
Lemma

- Let $\mathcal{D}_t$ be the dual value at round t
- Let $\Delta_t = \mathcal{D}_{t+1} - \mathcal{D}_t$ be the dual increase
- Assume that $\Delta_t \geq \ell(\mathbf{w}_t, (\mathbf{x}_t, y_t)) - \frac{1}{2\,\sigma}$
- Then,

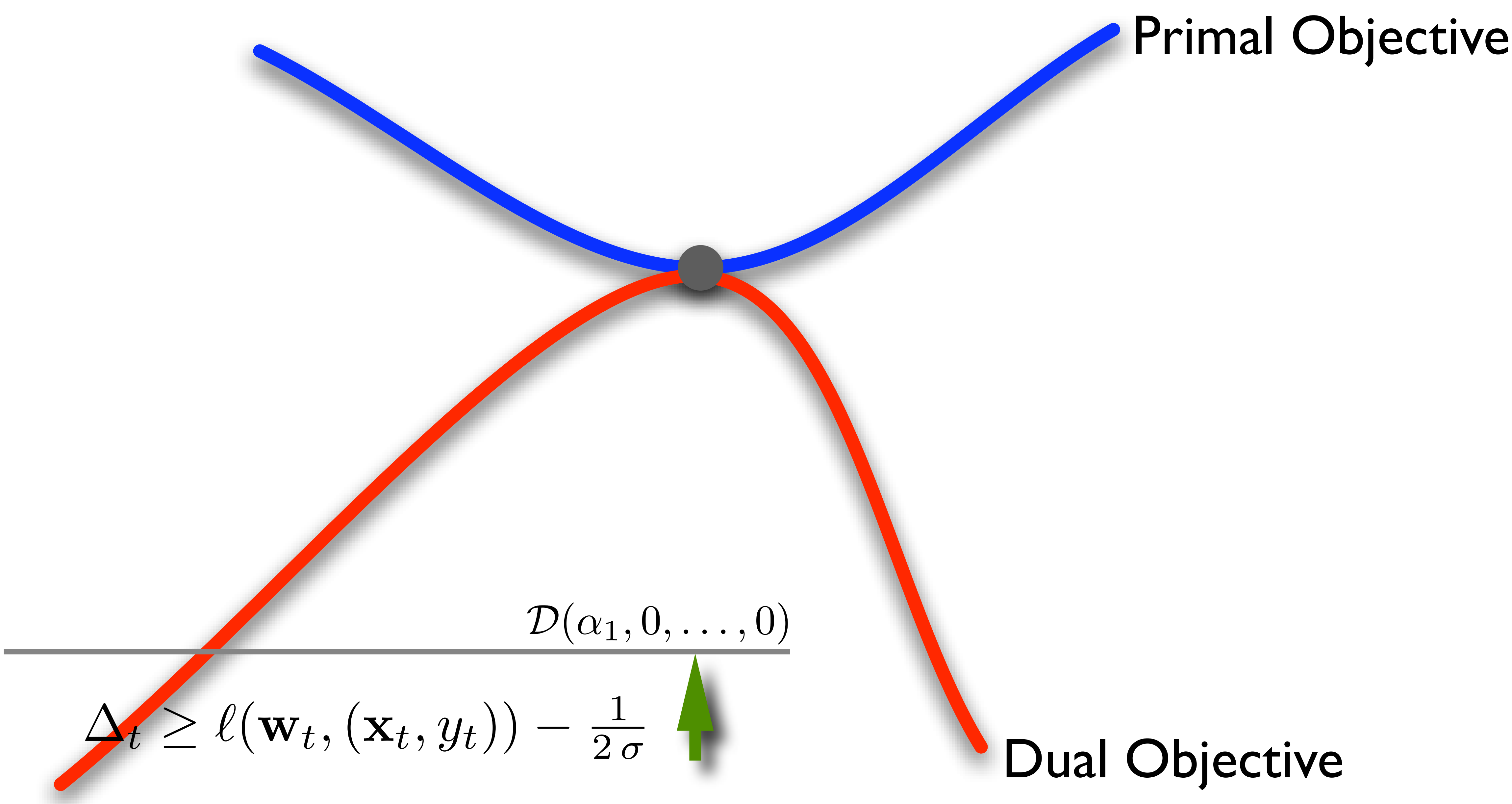
$$\sum_{t=1}^{T} \ell(\mathbf{w}_t, (\mathbf{x}_t, y_t)) - \sum_{t=1}^{T} \ell(\mathbf{w}^\star, (\mathbf{x}_t, y_t)) \;\leq\; O(\sqrt{T})$$

Proof follows from weak duality

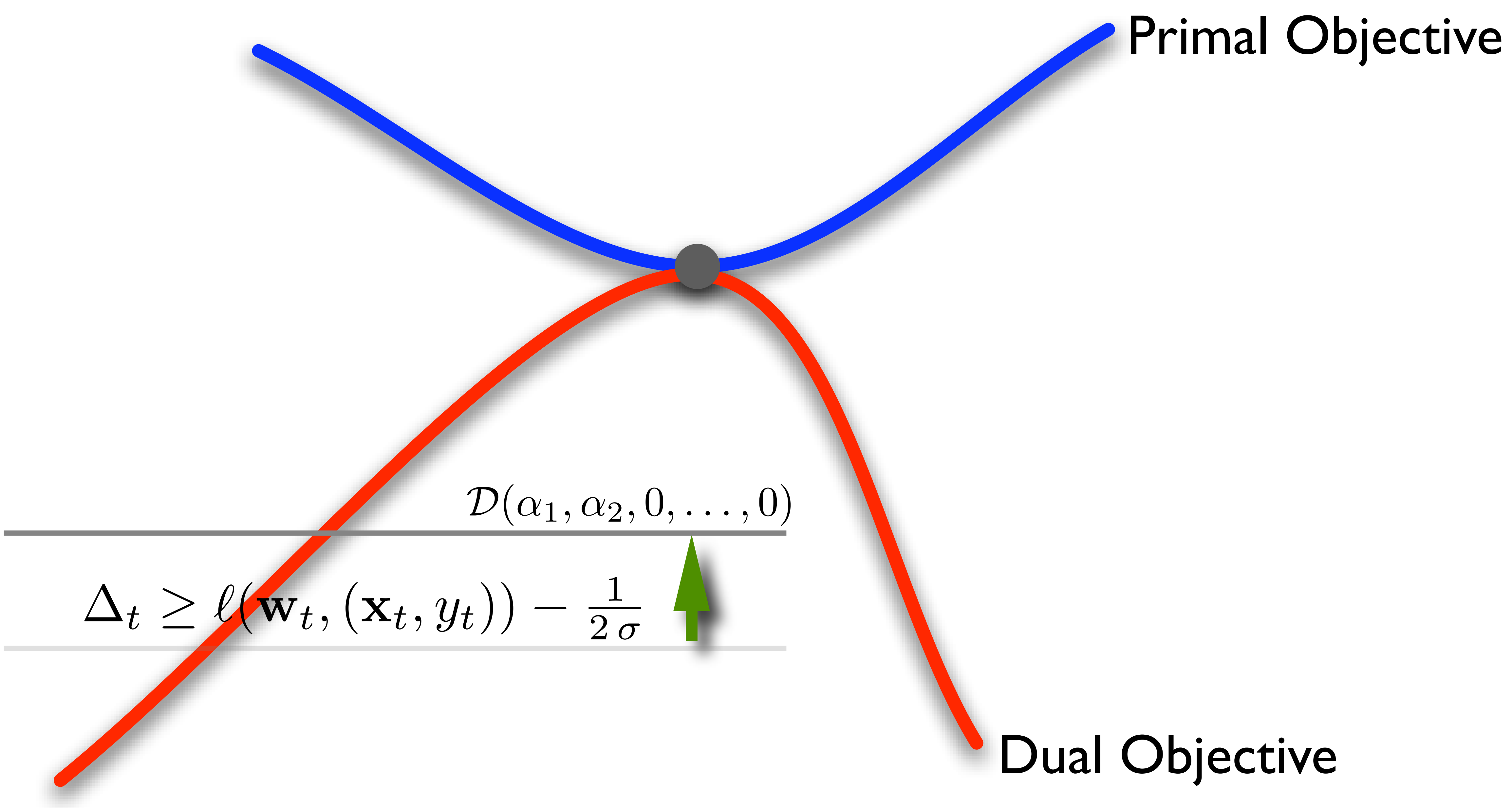
Proof by animation



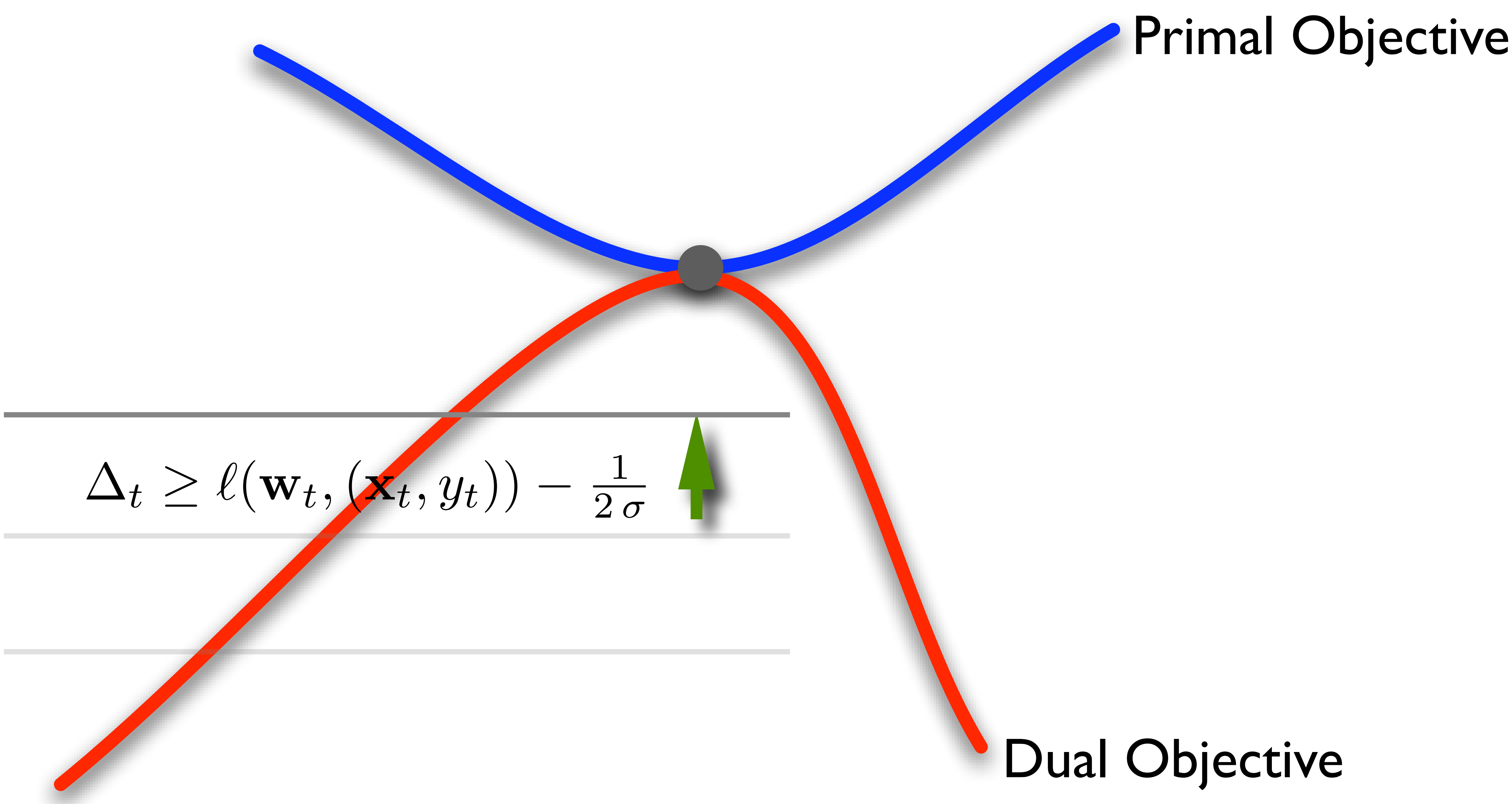
Proof by animation



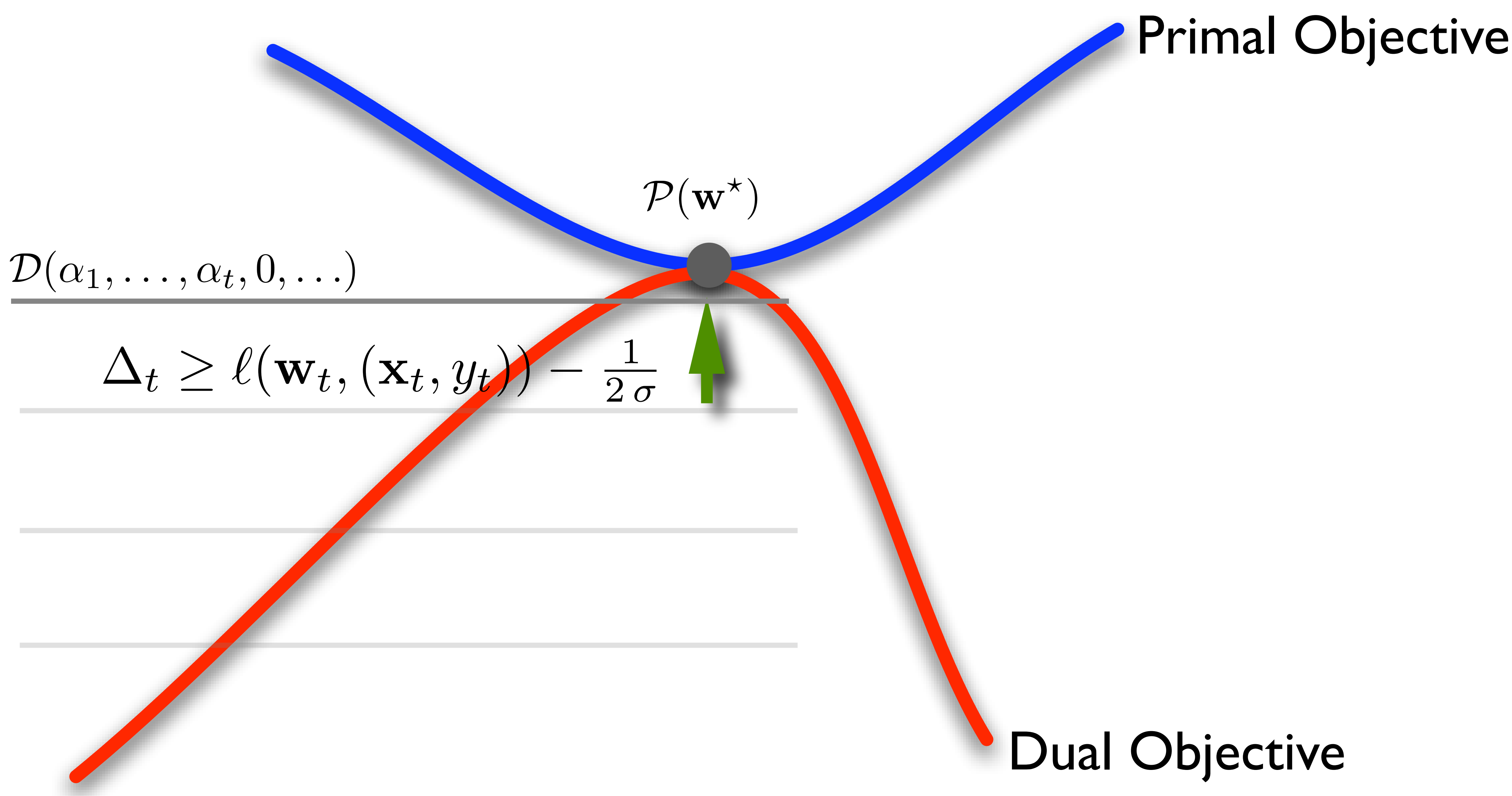
Proof by animation



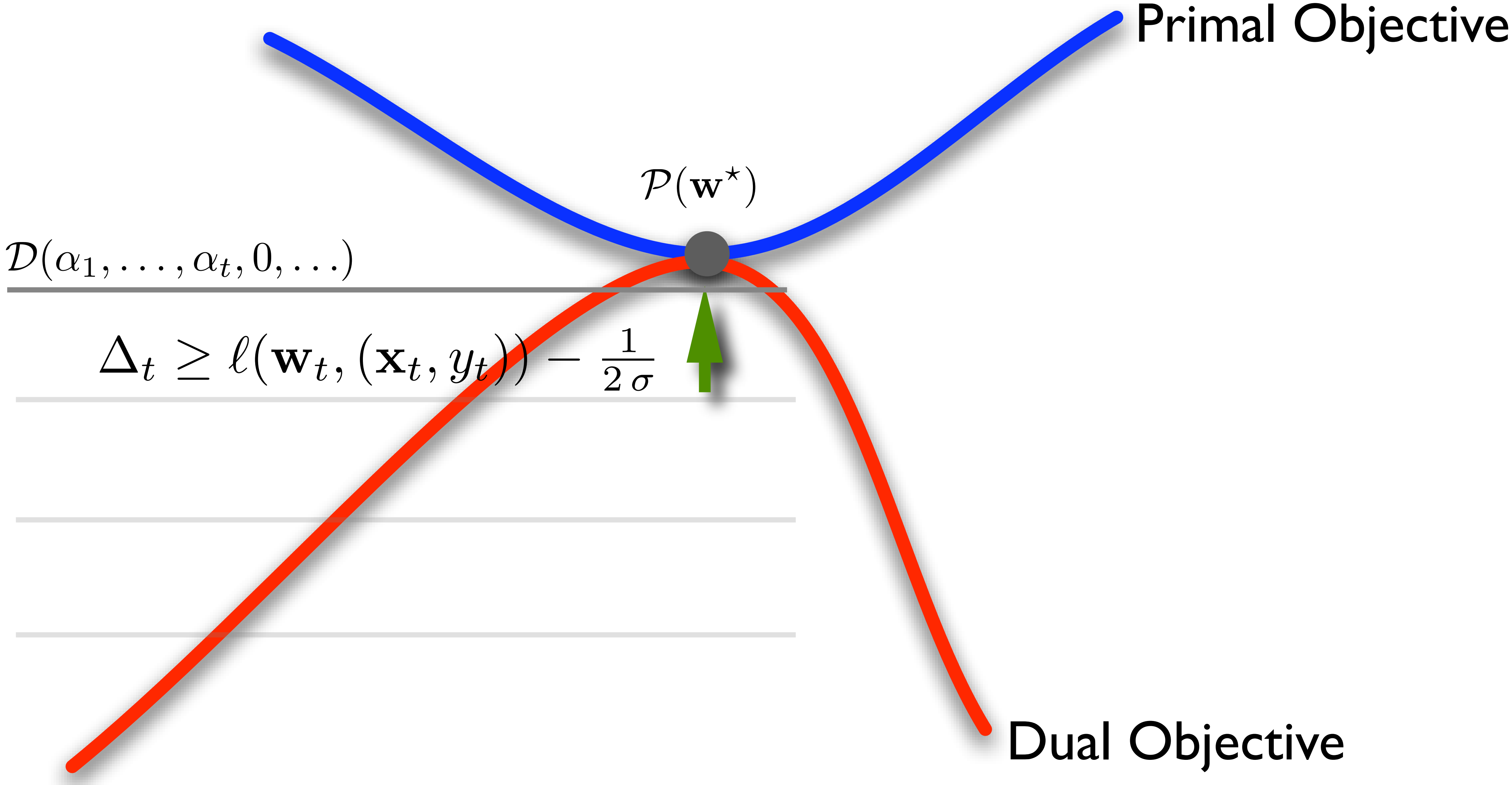
Proof by animation



Proof by animation



Proof by animation



$$\sum_t \ell_t(\mathbf{w}_t) - \frac{T}{2\,\sigma} \leq \sum_t \Delta_t = \mathcal{D}(\alpha_1, \ldots, \alpha_T) \leq \mathcal{P}(\mathbf{w}^\star)$$

Interim Recap

- To design an online algorithm:
  - Write an "SVM-like" problem
  - Switch to dual problem
  - Incrementally increase the dual

- Remains to describe:
  - How to construct $\alpha \Rightarrow \mathbf{w}$
  - Scheme works only if can guarantee a sufficient increase in dual form
    - Sufficient dual increase procedures

$\boldsymbol{\alpha} \Rightarrow \mathbf{w}$

- At the optimum $\quad \mathbf{w}^\star = \dfrac{1}{\sigma} \displaystyle\sum_t \alpha_t^\star y_t \mathbf{x}_t$

- Along the online learning process $\quad \mathbf{w}_t = \dfrac{1}{\sigma} \displaystyle\sum_{i<t} \alpha_i y_i \mathbf{x}_i$

- Recursive form (weight update) $\quad \mathbf{w}_{t+1} = \mathbf{w}_t + \dfrac{1}{\sigma} \alpha_t y_t \mathbf{x}_t$

- Note that dual can be rewritten as

$$\mathcal{D}_t \;=\; \sum_{i<t} \alpha_i - \frac{1}{2\,\sigma} \|\sigma \mathbf{w}_t\|^2$$

Sufficient Dual Increase

- For aggressive Perceptron
$$\alpha_t = \begin{cases} 1 & \text{if} \;\; 1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle > 0 \\ 0 & \text{else} \end{cases}$$

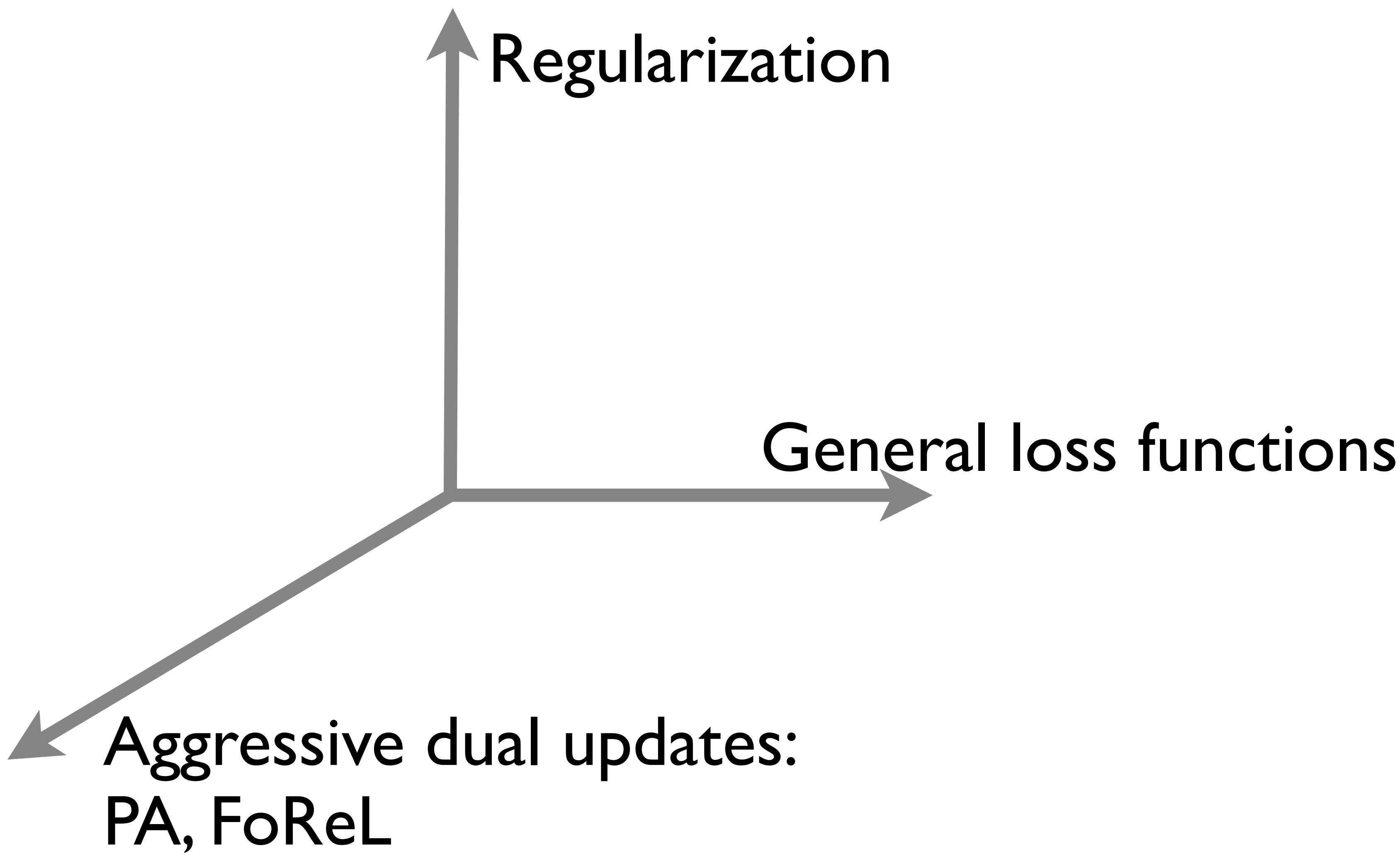
- If $\alpha_t = 0$ then $0 = \Delta_t = \ell_t(\mathbf{w}_t)$ and we're good

- If $\alpha_t = 1$ then

$$\begin{aligned} \Delta_t &= \left(\sum_{i \le t} \alpha_t - \tfrac{1}{2\,\sigma} \|\sigma \mathbf{w}_t + \alpha_t y_t \mathbf{x}_t\|^2 \right) - \left(\sum_{i < t} \alpha_i - \tfrac{1}{2\,\sigma} \|\sigma \mathbf{w}_t\|^2 \right) \\ &= 1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle - \frac{\|\mathbf{x}_t\|^2}{2\,\sigma} \\ &\ge \ell_t(\mathbf{w}_t) - \frac{1}{2\,\sigma} \end{aligned}$$

- Thus, in both cases we're good

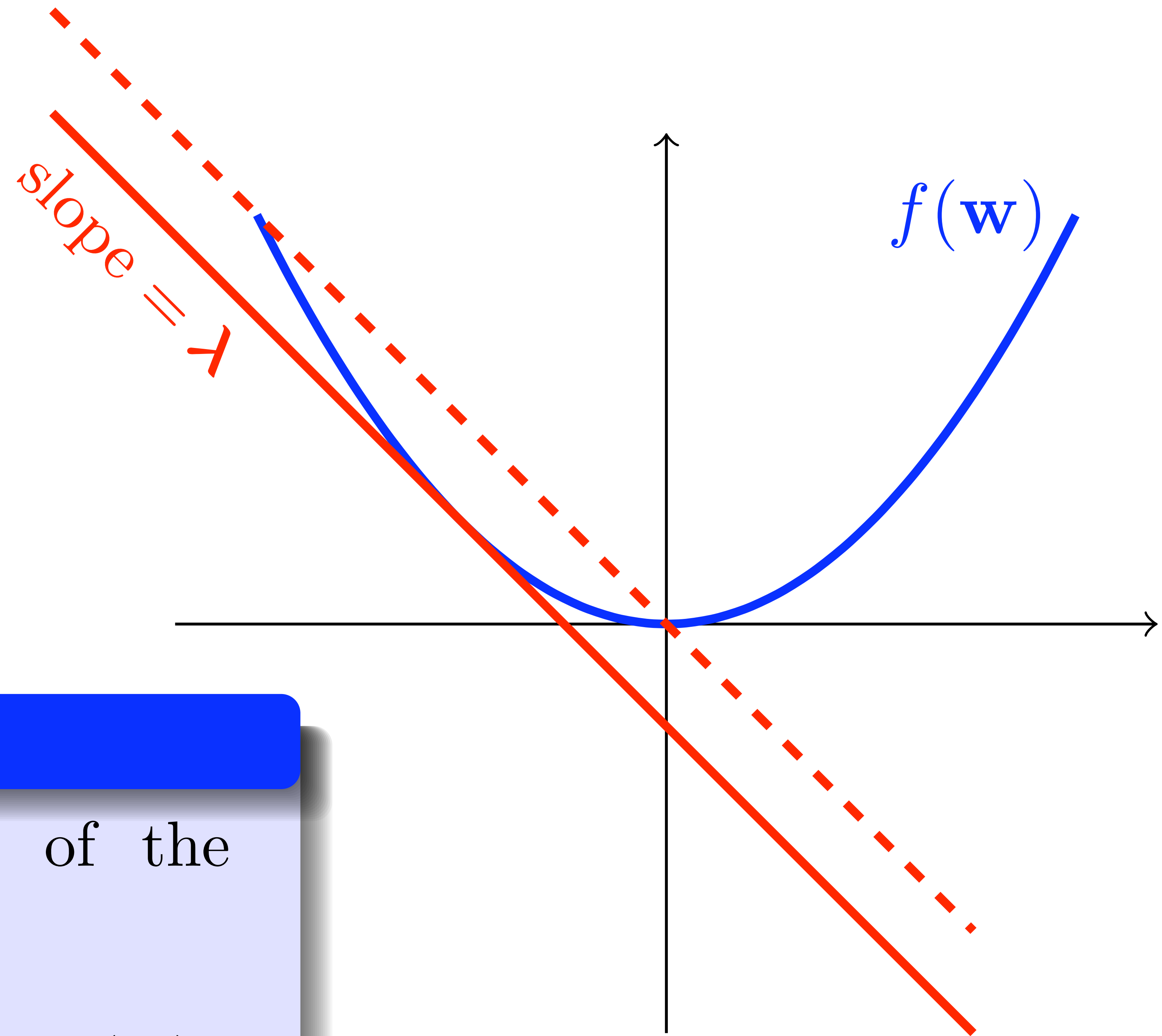
Three Directions for Generalization



Thus far:
specific settings

Next:
Primal-Dual apparatus
for online learning

Background – Fenchel Duality

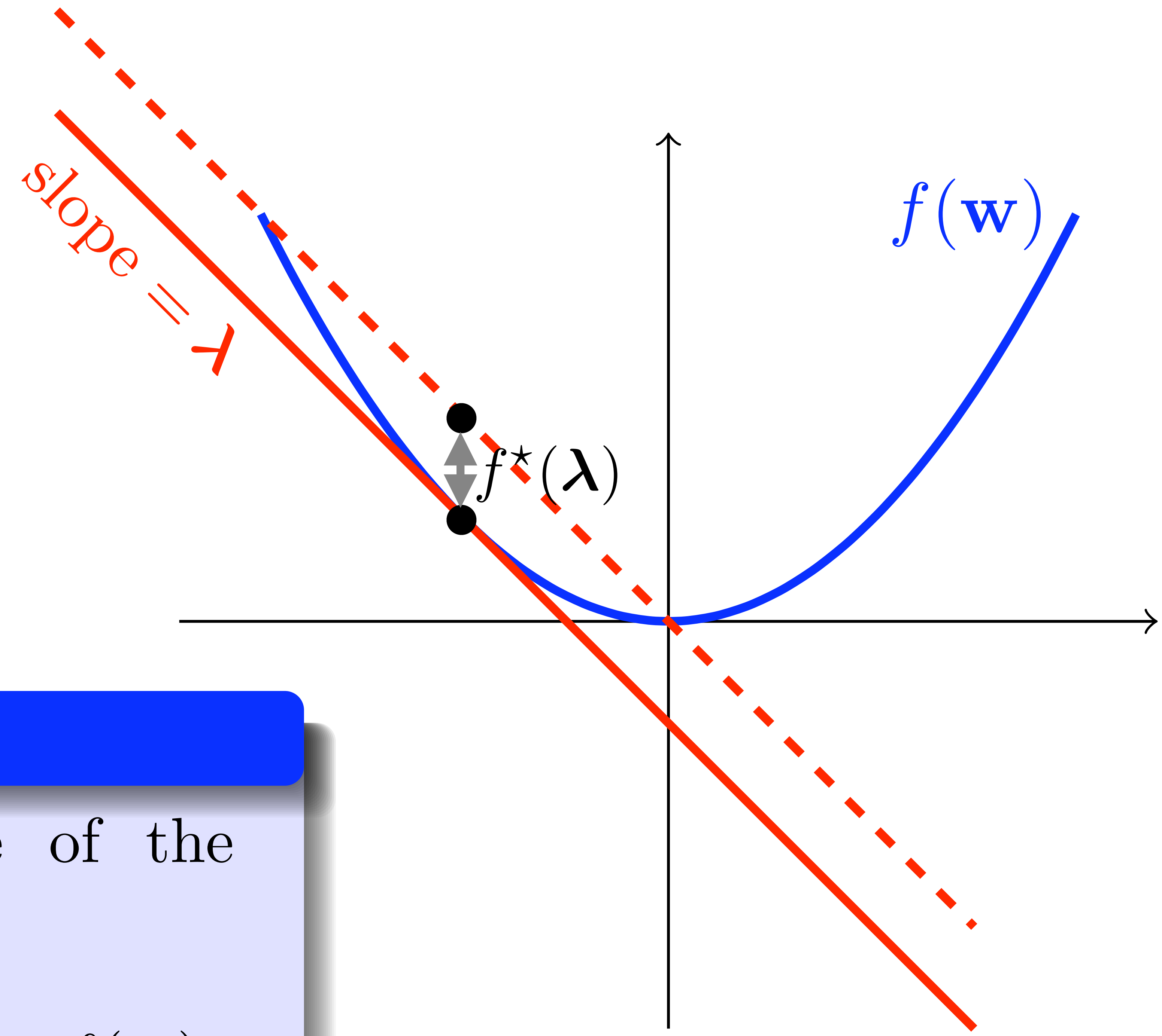


Fenchel Conjugate

The Fenchel conjugate of the function $f : S \to \mathbb{R}$ is

$$f^\star(\boldsymbol{\lambda}) \;=\; \max_{\mathbf{w} \in S} \; \langle \mathbf{w}, \boldsymbol{\lambda} \rangle - f(\mathbf{w})$$

Background – Fenchel Duality



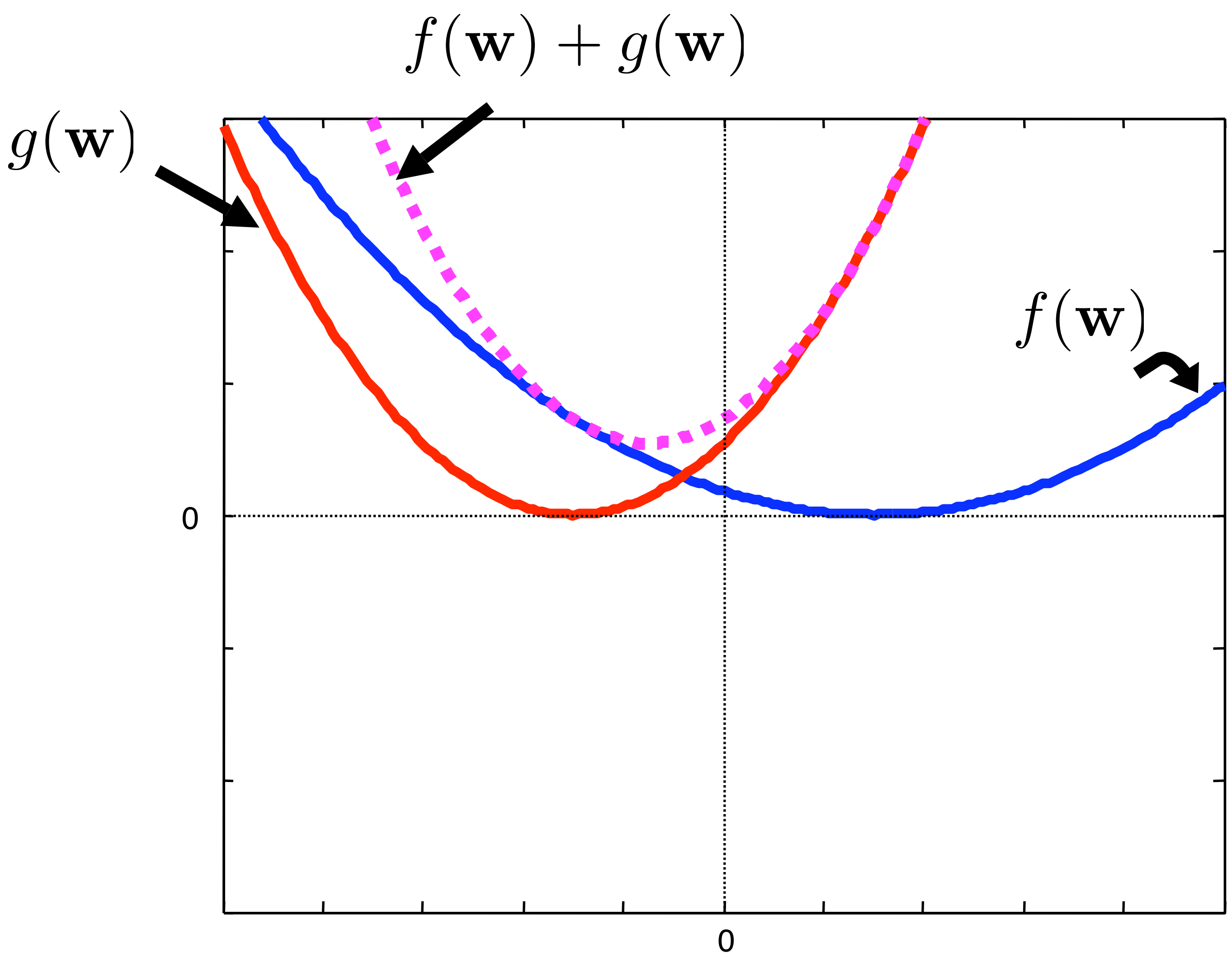
Fenchel Conjugate

The Fenchel conjugate of the function $f : S \to \mathbb{R}$ is

$$f^\star(\boldsymbol{\lambda}) = \max_{\mathbf{w} \in S} \langle \mathbf{w}, \boldsymbol{\lambda} \rangle - f(\mathbf{w})$$

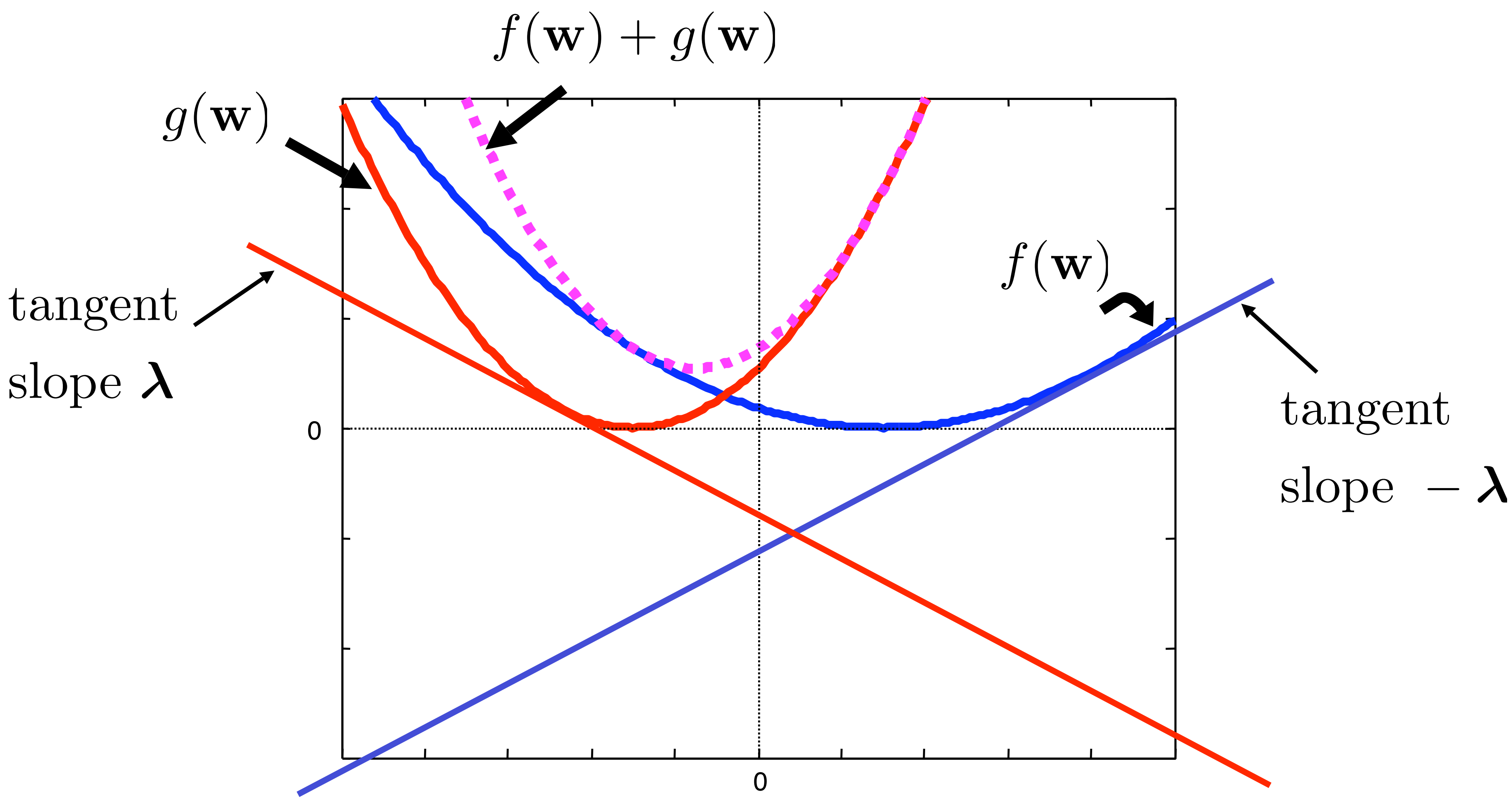
Background – Fenchel Duality

$$\max_{\boldsymbol{\lambda}} \; -f^{\star}(-\boldsymbol{\lambda}) - g^{\star}(\boldsymbol{\lambda}) \quad \leq \quad \min_{\mathbf{w}} \; f(\mathbf{w}) + g(\mathbf{w})$$



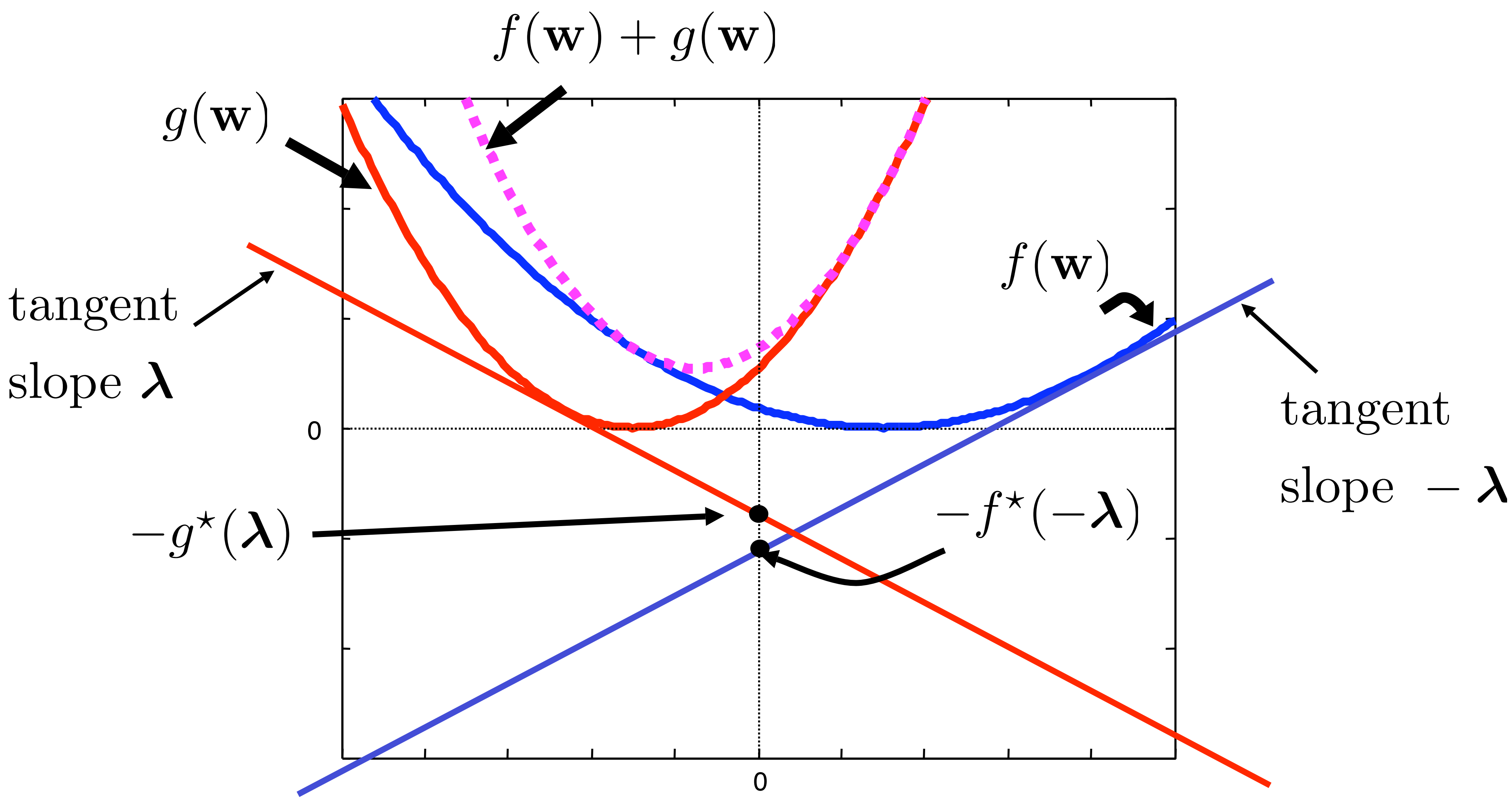
Background – Fenchel Duality

$$\max_{\boldsymbol{\lambda}} \; -f^\star(-\boldsymbol{\lambda}) - g^\star(\boldsymbol{\lambda}) \quad \leq \quad \min_{\mathbf{w}} \; f(\mathbf{w}) + g(\mathbf{w})$$



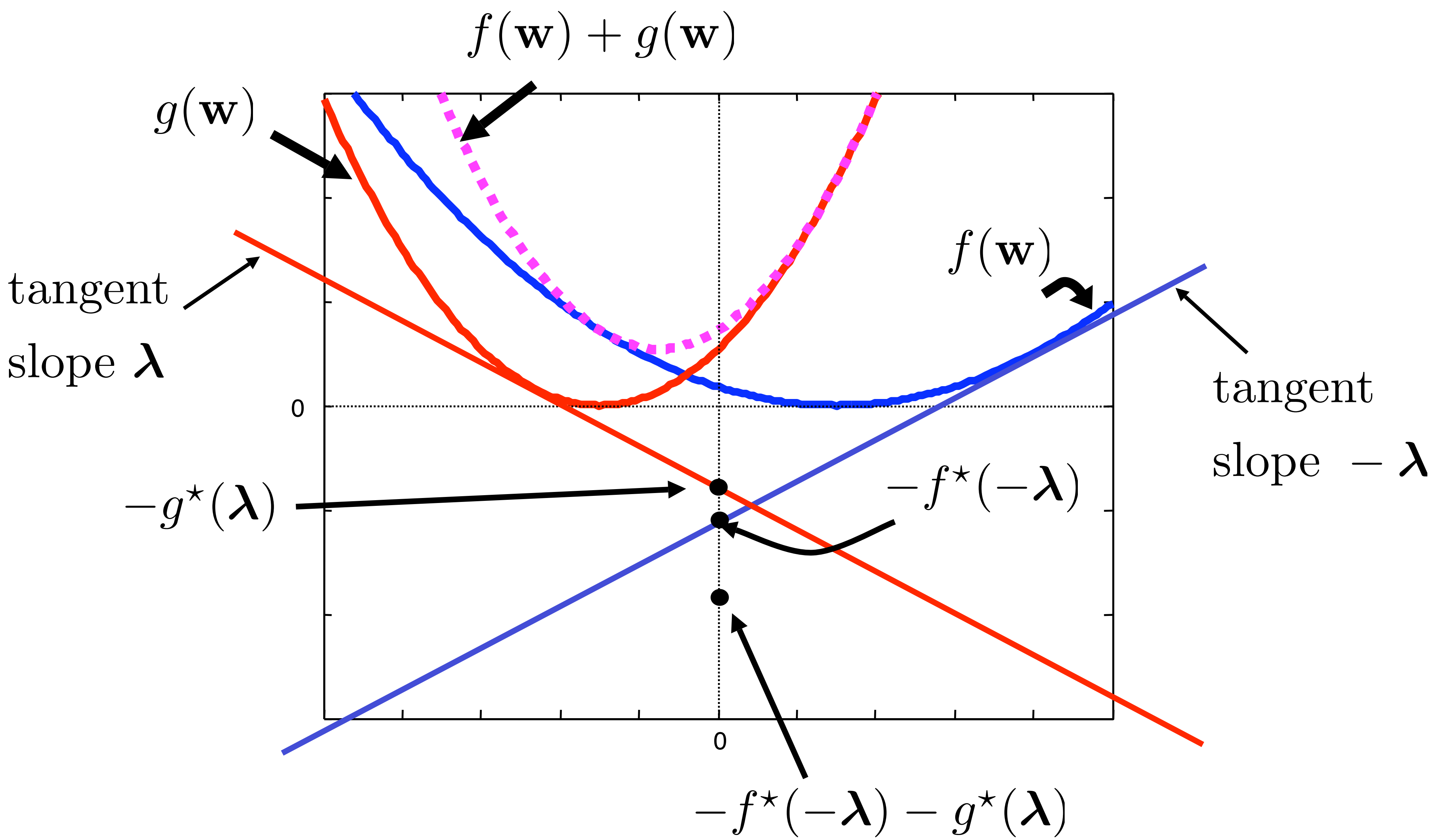
Background – Fenchel Duality

$$\max_{\boldsymbol{\lambda}} \; -f^\star(-\boldsymbol{\lambda}) - g^\star(\boldsymbol{\lambda}) \quad \leq \quad \min_{\mathbf{w}} \; f(\mathbf{w}) + g(\mathbf{w})$$



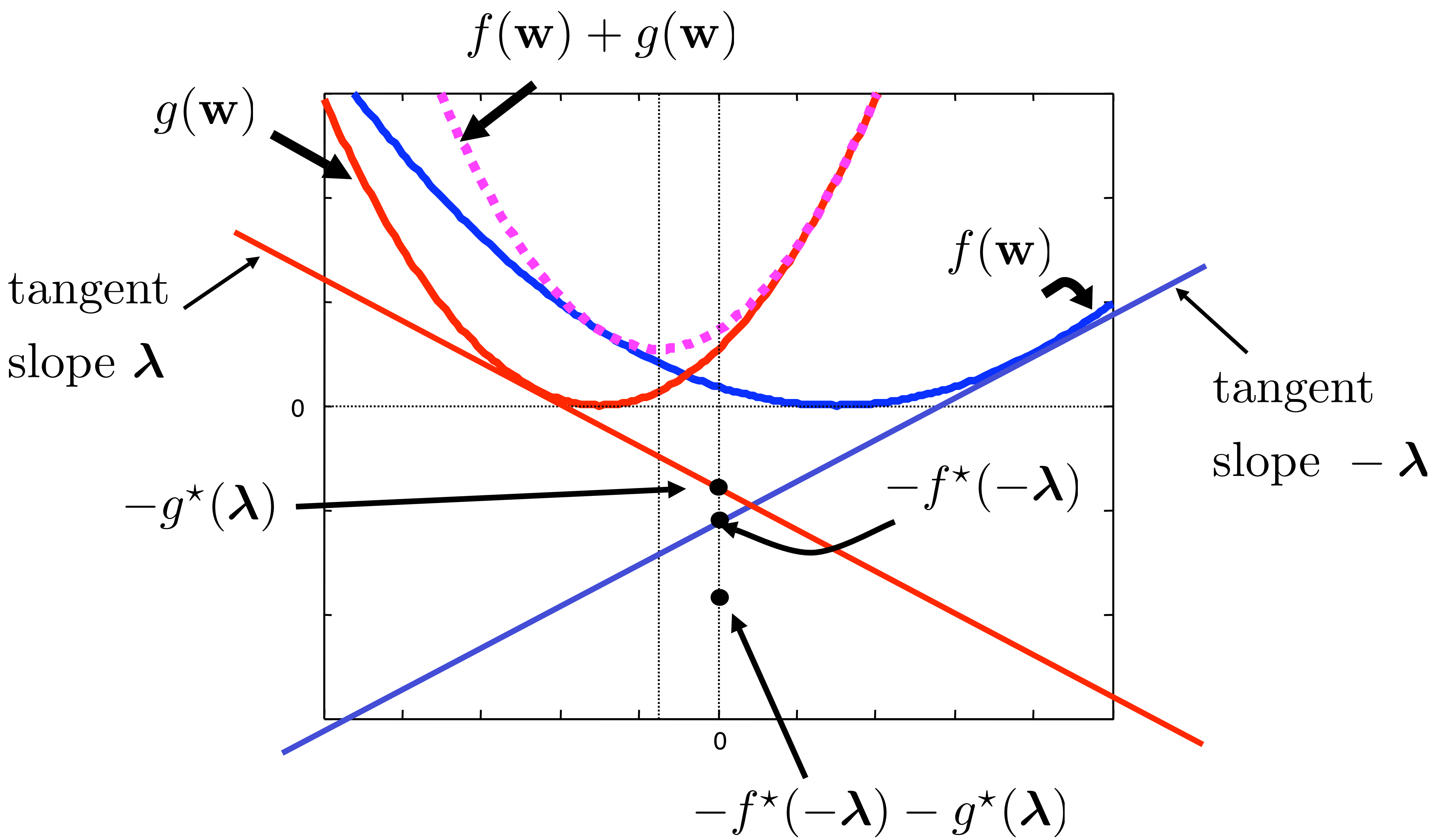
Background – Fenchel Duality

$$\max_{\boldsymbol{\lambda}} \; -f^\star(-\boldsymbol{\lambda}) - g^\star(\boldsymbol{\lambda}) \quad \leq \quad \min_{\mathbf{w}} \; f(\mathbf{w}) + g(\mathbf{w})$$



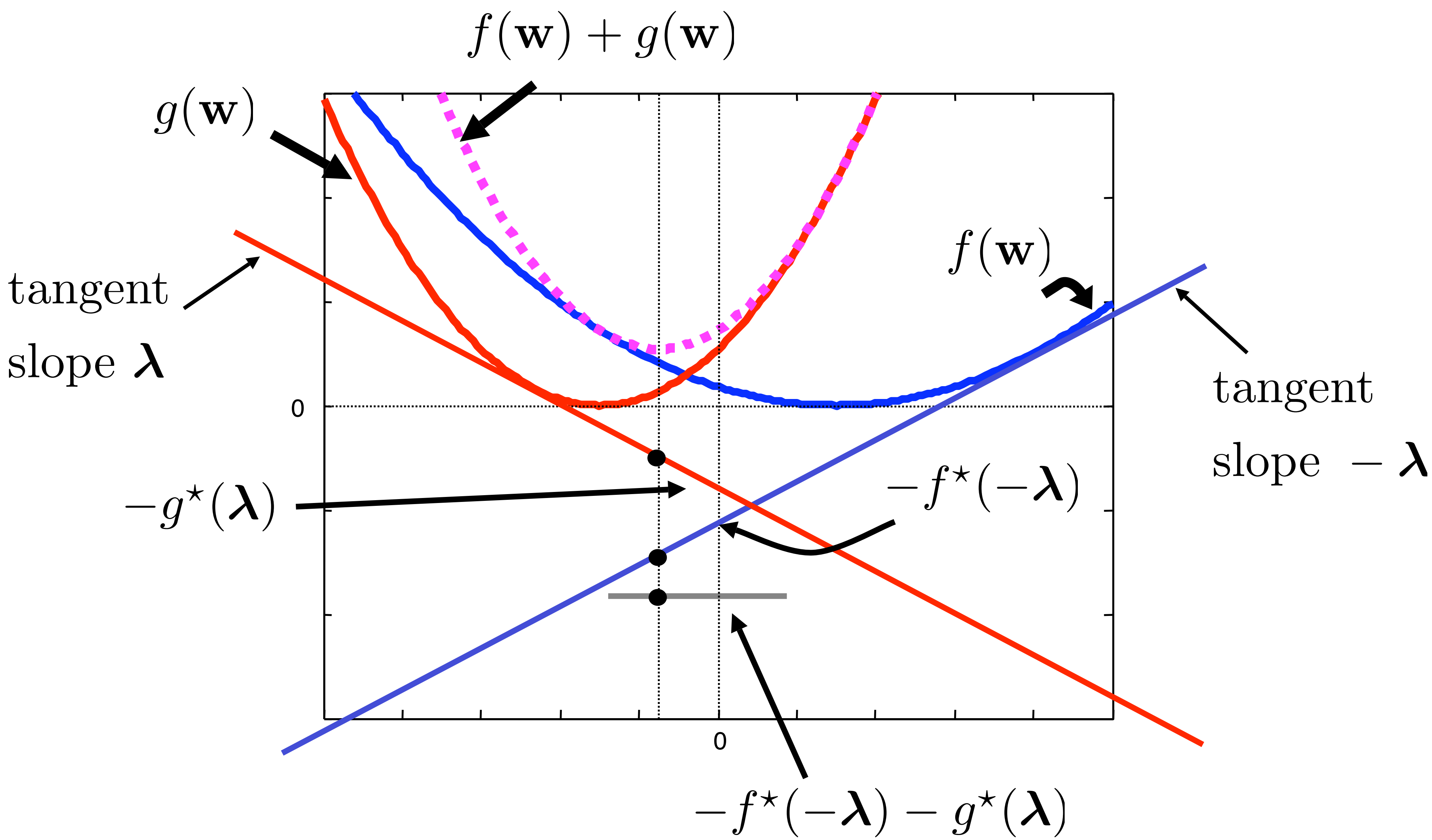
Background – Fenchel Duality

$$\max_{\boldsymbol{\lambda}} \; -f^\star(-\boldsymbol{\lambda}) - g^\star(\boldsymbol{\lambda}) \quad \leq \quad \min_{\mathbf{w}} \; f(\mathbf{w}) + g(\mathbf{w})$$



Background – Fenchel Duality

$$\max_{\boldsymbol{\lambda}} \; -f^\star(-\boldsymbol{\lambda}) - g^\star(\boldsymbol{\lambda}) \quad \le \quad \min_{\mathbf{w}} \; f(\mathbf{w}) + g(\mathbf{w})$$



Regret and Duality

$$\max_{\boldsymbol{\lambda}_1,\ldots,\boldsymbol{\lambda}_T} -f^\star(-\sum_t \boldsymbol{\lambda}_t) - \sum_t \ell_t^\star(\boldsymbol{\lambda}_t) \quad \leq \quad \min_{\mathbf{w}\in S} \; f(\mathbf{w}) + \sum_{t=1}^{T} \ell_t(\mathbf{w})$$

Decomposability of the dual

- Different dual variable associated with each online round
- Future loss functions do not affect dual variables of current and past rounds
- Therefore, the dual can be improved incrementally
- To optimize $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_t$, it is enough to know $\ell_1, \ldots, \ell_t$

Primal-Dual Online Prediction Strategy

Online Learning by Dual Ascent

- Initialize $\boldsymbol{\lambda}_1 = \ldots = \boldsymbol{\lambda}_T = \mathbf{0}$
- For $t = 1, 2, \ldots, T$
  - Construct $\mathbf{w}_t$ from the dual variables
  - Receive ℓ_t
  - Update dual variables $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_t$

Sufficient Dual Ascent → Low Regret

Lemma

Let $\mathcal{D}_t$ be the dual value at round t.

- Assume that $\mathcal{D}_{t+1} - \mathcal{D}_t \geq \ell_t(\mathbf{w}_t) - \frac{a}{\sqrt{T}}$
- Assume that $\max_{\mathbf{w} \in S} f(\mathbf{w}) \leq a\sqrt{T}$

Then, the regret is bounded by $2a\sqrt{T}$

Proof follows directly from weak duality !

Proof Sketch of Low Regret

- On one hand

$$\mathcal{D}_{T+1} = \sum_{t=1}^{T} (\mathcal{D}_{t+1} - \mathcal{D}_t) \geq \sum_t \ell_t(\mathbf{w}_t) - \frac{T\,a}{\sqrt{T}}$$

- On the other hand, from weak duality

$$\mathcal{D}_{T+1} \leq f(\mathbf{u}) + \sum_t \ell_t(\mathbf{u}) \leq a\sqrt{T} + \sum_t \ell_t(\mathbf{u})$$

- Comparing the lower and upper bound on $\mathcal{D}_{T+1}$

$$\sum_t \ell_t(\mathbf{w}_t) - \frac{T\,a}{\sqrt{T}} \leq a\sqrt{T} + \sum_t \ell_t(\mathbf{u}) \;\Rightarrow\; \sum_t \ell_t(\mathbf{w}_t) \leq \sum_t \ell_t(\mathbf{u}) + 2a\sqrt{T}$$

Proof Sketch of Low Regret

- On one hand

$$\mathcal{D}_{T+1} = \sum_{t=1}^{T}(\mathcal{D}_{t+1} - \mathcal{D}_t) \geq \sum_t \ell_t(\mathbf{w}_t) - \frac{T\,a}{\sqrt{T}}$$

- On the other hand, from weak duality

$$\mathcal{D}_{T+1} \leq f(\mathbf{u}) + \sum_t \ell_t(\mathbf{u}) \leq a\sqrt{T} + \sum_t \ell_t(\mathbf{u})$$

- Comparing the lower and upper bound on $\mathcal{D}_{T+1}$

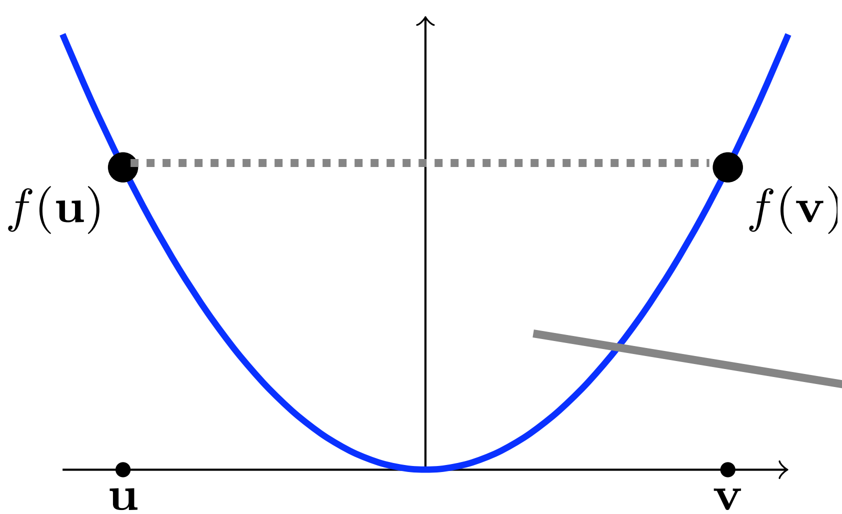
$$\sum_t \ell_t(\mathbf{w}_t) - \frac{T\,a}{\sqrt{T}} \leq a\sqrt{T} + \sum_t \ell_t(\mathbf{u}) \;\Rightarrow\; \sum_t \ell_t(\mathbf{w}_t) \leq \sum_t \ell_t(\mathbf{u}) + 2a\sqrt{T}$$

Strong Convexity $\rightarrow$ Sufficient Dual Increase

Definition – Strong Convexity

A function f is σ-strongly convex over S w.r.t $\|\cdot\|$ if

$$\forall \mathbf{u}, \mathbf{v} \in S, \quad \frac{f(\mathbf{u})+f(\mathbf{v})}{2} \;\geq\; f\left(\frac{\mathbf{u}+\mathbf{v}}{2}\right) + \frac{\sigma}{8}\|\mathbf{u}-\mathbf{v}\|^2$$



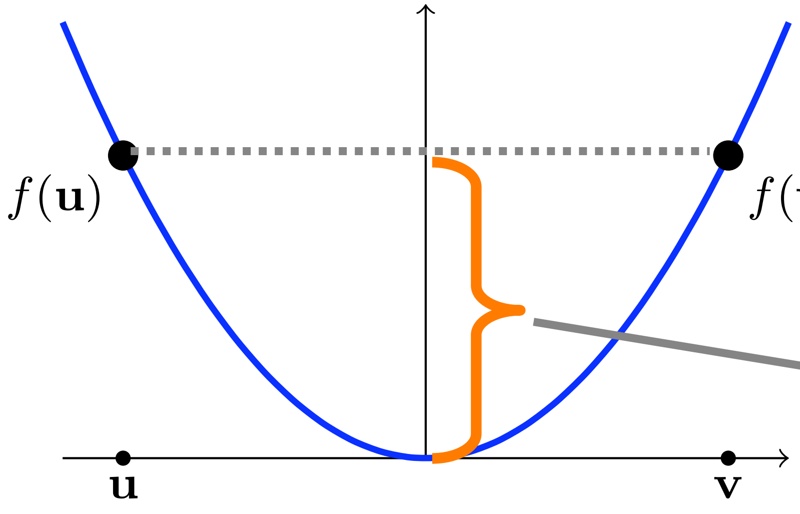
$$\frac{f(\mathbf{u})+f(\mathbf{v})}{2} - f\left(\frac{\mathbf{u}+\mathbf{v}}{2}\right)$$

Strong Convexity ➡ Sufficient Dual Increase

Definition – Strong Convexity

A function f is σ-strongly convex over S w.r.t $\|\cdot\|$ if

$$\forall \mathbf{u}, \mathbf{v} \in S, \quad \frac{f(\mathbf{u})+f(\mathbf{v})}{2} \;\geq\; f\left(\frac{\mathbf{u}+\mathbf{v}}{2}\right) + \frac{\sigma}{8}\|\mathbf{u}-\mathbf{v}\|^2$$



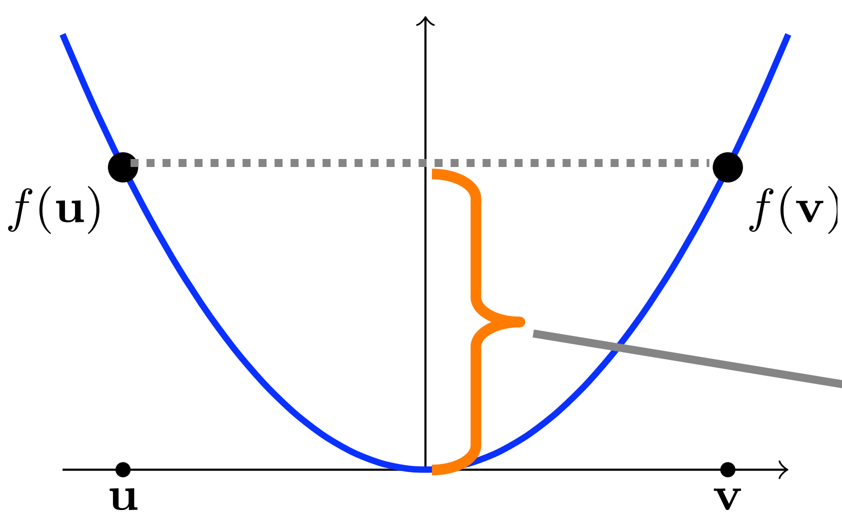
$$\frac{f(\mathbf{u})+f(\mathbf{v})}{2} - f\left(\frac{\mathbf{u}+\mathbf{v}}{2}\right)$$

Strong Convexity ➡ Sufficient Dual Increase

Definition – Strong Convexity

A function f is σ-strongly convex over S w.r.t $\|\cdot\|$ if

$$\forall \mathbf{u},\mathbf{v} \in S, \quad \frac{f(\mathbf{u})+f(\mathbf{v})}{2} \geq f\left(\frac{\mathbf{u}+\mathbf{v}}{2}\right) + \frac{\sigma}{8}\|\mathbf{u}-\mathbf{v}\|^2$$



$$\frac{f(\mathbf{u})+f(\mathbf{v})}{2} - f\left(\frac{\mathbf{u}+\mathbf{v}}{2}\right)$$

Example:
$f(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|_2^2$ is 1 strongly convex w.r.t. $\|\cdot\|_2$

L-Lipschitz ➡ Sufficient Dual Increase

Definition – Lipschitz

A function ℓ is L-Lipschitz w.r.t. $\|\cdot\|$ if

$$\forall \mathbf{u}, \mathbf{v} \in S, \quad |\ell(\mathbf{u}) - \ell(\mathbf{v})| \leq L \, \|\mathbf{u} - \mathbf{v}\|$$

Example:
$\ell(\mathbf{w}) = |y - \langle \mathbf{w}, \mathbf{x} \rangle|$ is L-Lipschitz w.r.t. $\|\cdot\|$ with $L = \|\mathbf{x}\|$

Strong Convexity ➡ Sufficient Dual Increase

Sufficient Dual Increase for Gradient Descent

Assume:

- f is σ-strongly convex w.r.t. $\|\cdot\|$
- ℓ_t is convex, and L-Lipschitz w.r.t. $\|\cdot\|_\star$
- $\mathbf{w}_t = \nabla f^\star(-\sum_{i<t} \boldsymbol{\lambda}_i)$
- Set $\boldsymbol{\lambda}_t$ to be a subgradient of ℓ_t at $\mathbf{w}_t$
- Keep $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{t-1}$ in tact

Then,

$$\mathcal{D}_{t+1} - \mathcal{D}_t \;\geq\; \ell_t(\mathbf{w}_t) - \frac{L^2}{2\sigma}$$

General Algorithmic Framework

Online Learning by Dual Ascent

- Choose σ-strongly convex complexity function f
- For $t = 1, 2, \ldots, T$
  - Predict $\mathbf{w}_t = \nabla f^\star(-\sum_{i<t} \boldsymbol{\lambda}_i)$
  - Receive ℓ_t
  - Update dual variables $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_t$ s.t. $\mathcal{D}_{t+1} - \mathcal{D}_t \geq \ell_t(\mathbf{w}_t) - \frac{L^2}{2\sigma}$ (e.g. by gradient descent)

General Algorithmic Framework

Online Learning by Dual Ascent

- Choose σ-strongly convex complexity function f
- For $t = 1, 2, \ldots, T$
  - Predict $\mathbf{w}_t = \nabla f^\star(-\sum_{i<t} \boldsymbol{\lambda}_i)$
  - Receive ℓ_t
  - Update dual variables $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_t$ s.t. $\mathcal{D}_{t+1} - \mathcal{D}_t \geq \ell_t(\mathbf{w}_t) - \frac{L^2}{2\sigma}$ (e.g. by gradient descent)

Gradient descent on the (primal) loss ℓ_t results in sufficient dual increase if f is strongly convex and the losses are L-Lipshitz (do not grow excessively fast)

General Regret Bound

Theorem – General Regret Bound

Assume:

- f is σ-strongly convex w.r.t. $\|\cdot\|$
- ℓ_t is convex, and L-Lipschitz w.r.t. $\|\cdot\|_\star$

Then, the regret of all algorithms derived from the general framework is upper bounded by $f(\mathbf{w}^\star) + \frac{T\,L^2}{2\sigma}$

General Regret Bound

Theorem – General Regret Bound

Assume:

- f is σ-strongly convex w.r.t. $\|\cdot\|$
- ℓ_t is convex, and L-Lipschitz w.r.t. $\|\cdot\|_\star$

Then, the regret of all algorithms derived from the general framework is upper bounded by $f(\mathbf{w}^\star) + \frac{T\,L^2}{2\sigma}$

Corollary – Euclidean norm Regularization

- If S is the Euclidean ball of radius W and ℓ_t is convex, and L-Lipschitz w.r.t. $\|\cdot\|_2$
- Set $f = \frac{\sigma}{2}\|\mathbf{w}\|^2$ with $\sigma = \frac{\sqrt{T}\,L}{W}$
- Then, the regret is upper bounded by $L\,W\,\sqrt{T}$

General Regret Bound

Theorem – General Regret Bound

Assume:

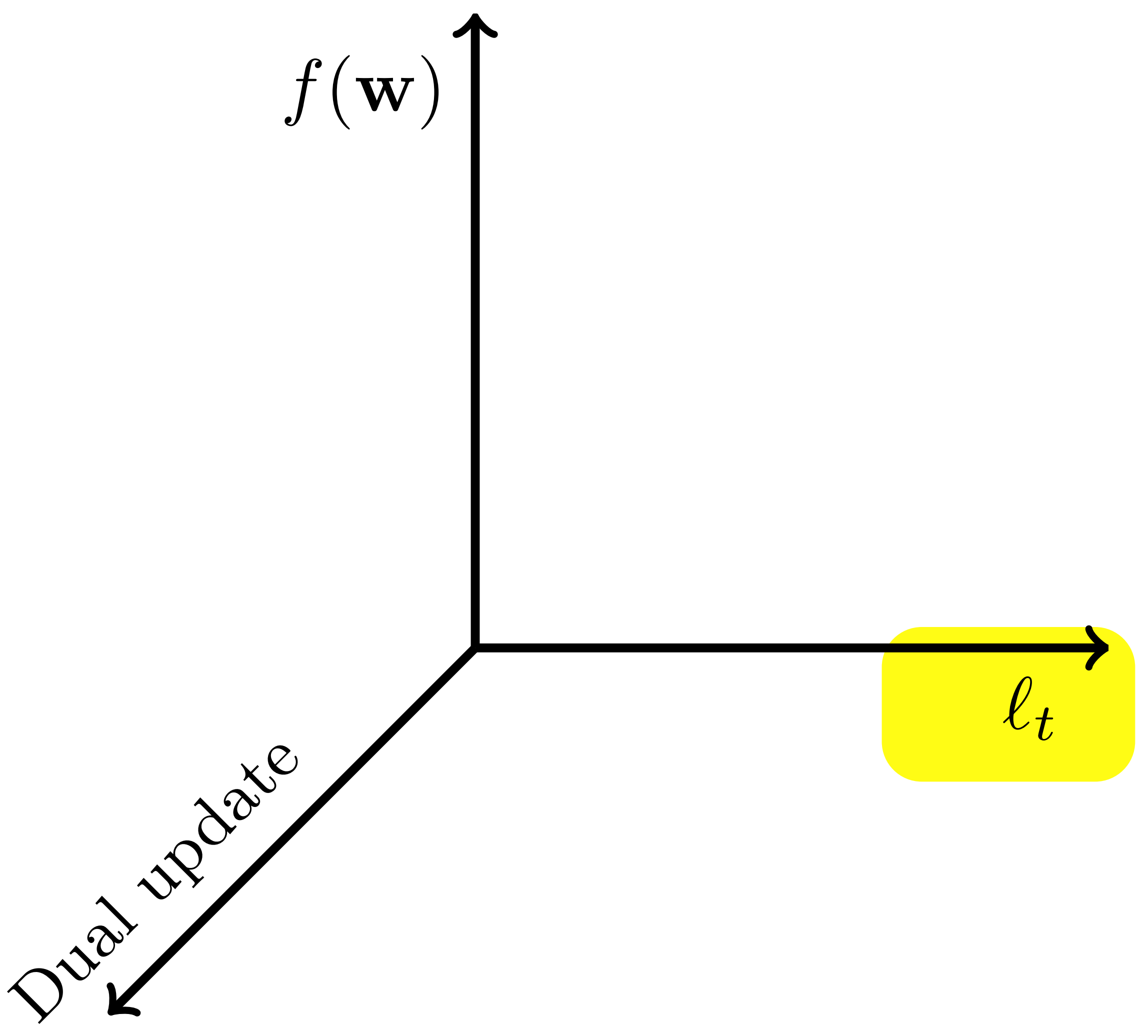
- f is σ-strongly convex w.r.t. $\|\cdot\|$
- ℓ_t is convex, and L-Lipschitz w.r.t. $\|\cdot\|_\star$

Then, the regret of all algorithms derived from the general framework is upper bounded by $f(\mathbf{w}^\star) + \frac{T\,L^2}{2\sigma}$

Corollary – Entropic regularization

- If S is the d-dim probability simplex and ℓ_t is convex, and L-Lipschitz w.r.t. $\|\cdot\|_\infty$
- Set $f = \sigma \sum_i w_i \log(d\,w_i)$ with $\sigma = \frac{\sqrt{T}\,L}{\sqrt{\log(d)}}$
- Then, the regret is upper bounded by $L\sqrt{\log(d)\,T}$

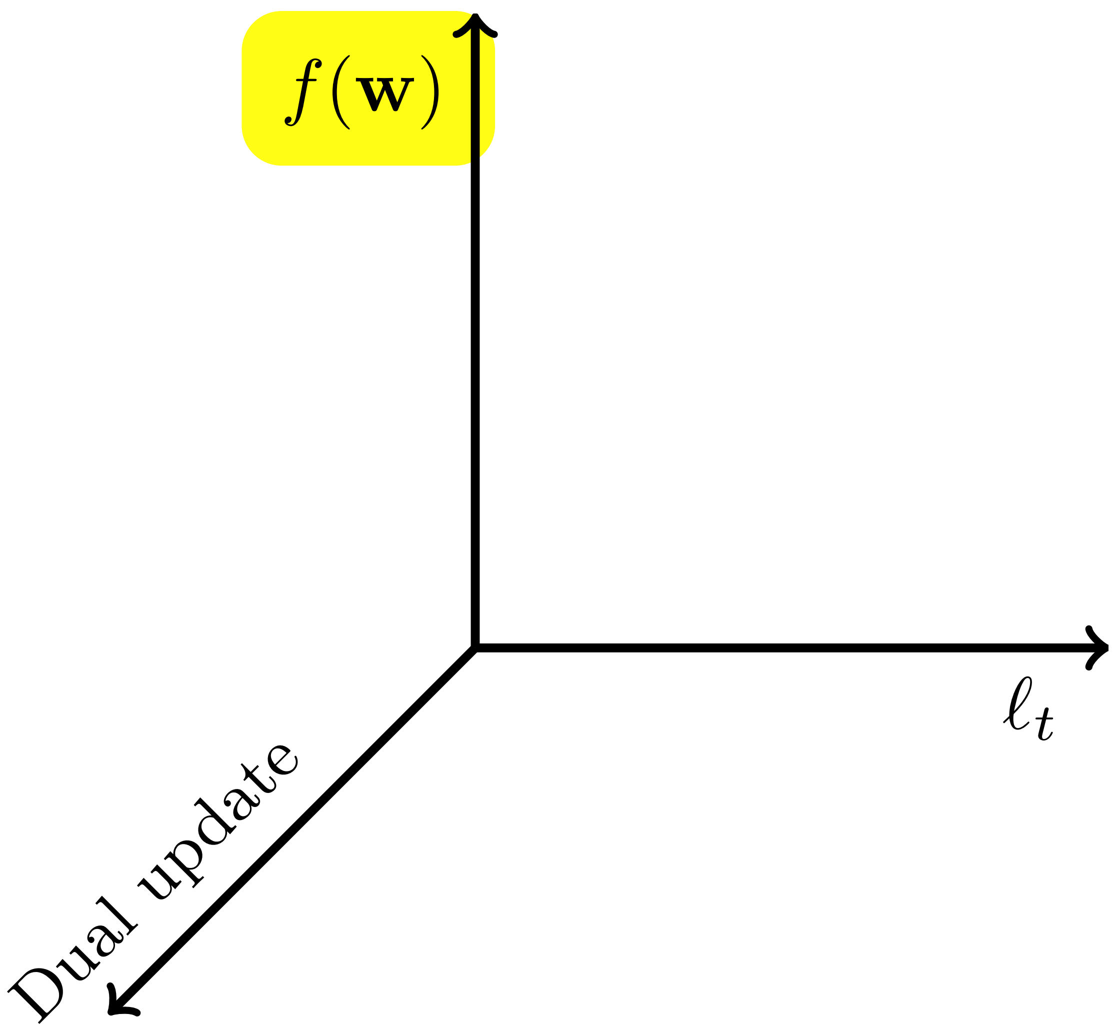
Generalizations and Related Work



Family of loss functions (ℓ_t)

- Online Learning (Perceptron, linear regression, multiclass prediction, structured output, ...)
- Game theory (Playing repeated games, correlated equilibrium)
- Information theory (Prediction of individual sequences)
- Convex optimization (SGD, dual decomposition)

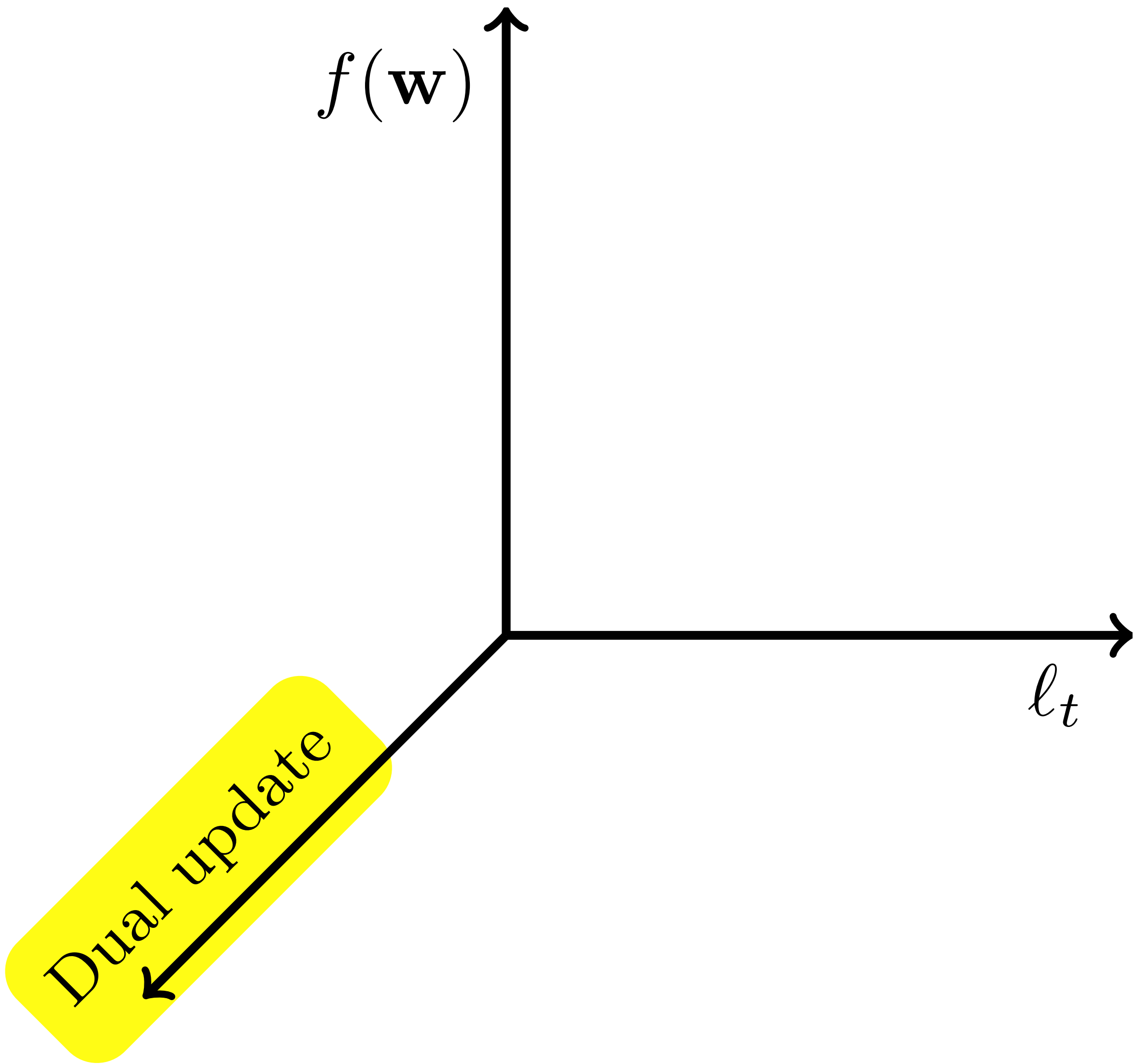
Generality and Related Work



Regularization function (f)

- Online learning
  (Grove, Littlestone, Schuurmans; Kivinen, Warmuth; Gentile; Vovk)
- Game theory
  (Hart and Mas-collel)
- Optimization
  (Nemirovsky, Yudin; Beck, Teboulle, Nesterov)
- Unified frameworks
  (Cesa-Bianchi and Lugosi)

Generality and Related Work



Dual update schemes

- Only two extremes were studied:
  - Gradient update (naive update of a single dual variable)
  - Follow the leader (Equivalent to full optimization)
- Our analysis enables the usage the entire spectrum of possible updates

Part III:
Derived Algorithms

Fenchel Dual of SVM

- SVM primal:

$$\underbrace{\frac{\sigma}{2}\|\mathbf{w}\|^2}_{f(\mathbf{w})} + \sum_{i=1}^{T} \underbrace{[1 - y_i\langle \mathbf{w}, \mathbf{x}_i\rangle]_+}_{\ell_i(\mathbf{w})}$$

- Fenchel dual of $f(\mathbf{w}) \;\Rightarrow\; f^\star(\boldsymbol{\lambda}) = \max_{\mathbf{w}} \langle \mathbf{w}, \boldsymbol{\lambda}\rangle - \frac{\sigma}{2}\|\mathbf{w}\|^2$

$$\boldsymbol{\lambda} - \sigma\mathbf{w} = 0 \;\Rightarrow\; \boldsymbol{\lambda}/\sigma = \mathbf{w} \;\Rightarrow\; f^\star(\boldsymbol{\lambda}) = \langle \boldsymbol{\lambda}/\sigma, \boldsymbol{\lambda}\rangle - \frac{\sigma}{2}\|\boldsymbol{\lambda}/\sigma\|^2 = \frac{1}{2\sigma}\|\boldsymbol{\lambda}\|^2$$

- Fenchel dual of hinge-loss $\quad f^\star(\lambda) = \begin{cases} -\alpha & \lambda = -\alpha\mathbf{x} \text{ and } \alpha \in [0,1] \\ \infty & \text{otherwise} \end{cases}$

- The Fenchel dual of SVM

$$-f^*(-\sum_t \boldsymbol{\lambda}_t) - \sum_t \ell^\star(\boldsymbol{\lambda}_t) \;=\; -\frac{1}{2\sigma}\|-\sum_t \alpha_t y_t \mathbf{x}_t\|^2 - \sum_t -\alpha_t \text{ s.t. } \alpha_i \in [0,1]$$

Online SVM Revisited

- Since $f^\star(\mathbf{v}) = f^\star(-\mathbf{v})$ and $\nabla f^\star(\mathbf{v}) = \mathbf{v}$,

$$\mathbf{w}_{t+1} = \nabla f^\star(-\sum_{i<t+1} \boldsymbol{\lambda}_i) = \sum_{i<t+1} \alpha_i \mathbf{x}_i = \sum_{i<t} \alpha_i \mathbf{x}_i + \alpha_t \mathbf{x}_t = \mathbf{w}_t + \alpha_t \mathbf{x}_t$$

- We saw that obtain a regret bound if $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{L^2}{2\sigma}$ where L is the Lipschitz constant of ℓ_t w.r.t $\|\cdot\|_\star$
- We can use gradient descent (on the primal) to achieve sufficient increase of the dual objective:
  - Gradient descent:
    1. $\boldsymbol{\lambda}_t = -\mathbf{x}_t$ ($\boldsymbol{\lambda}_t = -\alpha_t \mathbf{x}_t$ with $\alpha_t = 1$) when $[1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle]_+ > 0$
    2. $\boldsymbol{\lambda}_t = 0$ otherwise
  - Dual increase: $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{1}{2\sigma}$
- Can we potentially make faster progress in the dual while maintaining the regret bound?

Online SVM Revisited

- Since $f^\star(\mathbf{v}) = f^\star(-\mathbf{v})$ and $\nabla f^\star(\mathbf{v}) = \mathbf{v}$,

$$\mathbf{w}_{t+1} = \nabla f^\star(-\sum_{i<t+1} \boldsymbol{\lambda}_i) = \sum_{i<t+1} \alpha_i \mathbf{x}_i = \sum_{i<t} \alpha_i \mathbf{x}_i + \alpha_t \mathbf{x}_t = \mathbf{w}_t + \alpha_t \mathbf{x}_t$$

- We saw that obtain a regret bound if $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{L^2}{2\sigma}$ where L is the Lipschitz constant of ℓ_t w.r.t $\|\cdot\|_\star$

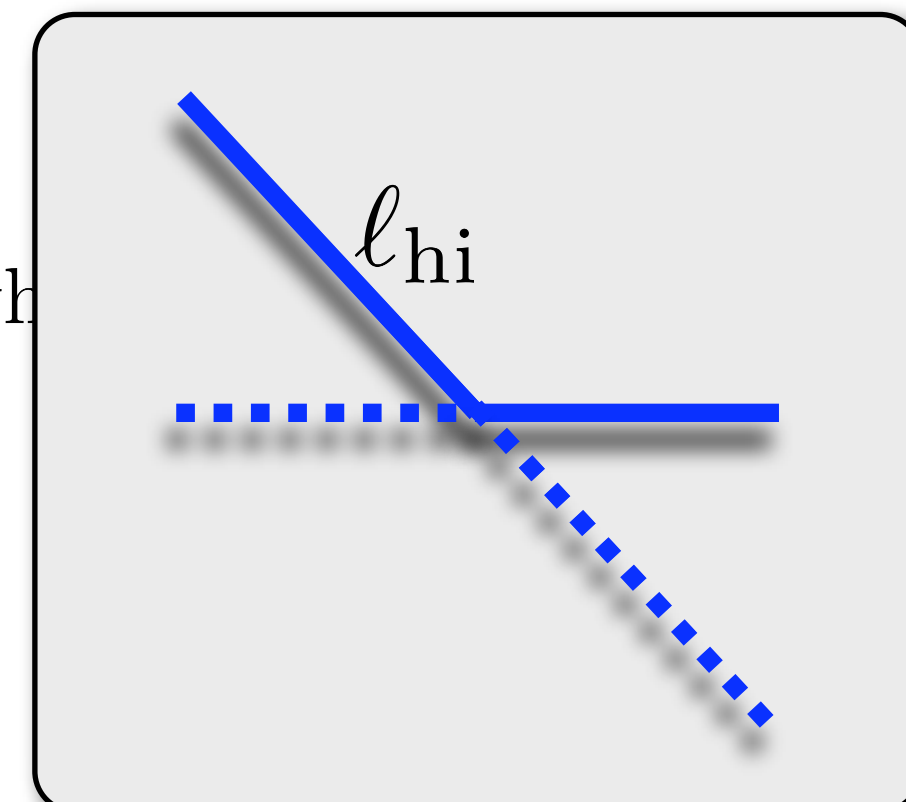
- We can use gradient descent (on the <u>primal</u>) to achieve sufficient increase of the <u>dual</u> objective:

    - Gradient descent:

        1. $\boldsymbol{\lambda}_t = -\mathbf{x}_t$ ($\boldsymbol{\lambda}_t = -\alpha_t \mathbf{x}_t$ with $\alpha_t = 1$) wh
        2. $\boldsymbol{\lambda}_t = 0$ otherwise

    - Dual increase: $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{1}{2\sigma}$

- Can we potentially make faster progress in the dual while maintaining the regret bound?



Online SVM Revisited

- Since $f^\star(\mathbf{v}) = f^\star(-\mathbf{v})$ and $\nabla f^\star(\mathbf{v}) = \mathbf{v}$,

$$\mathbf{w}_{t+1} = \nabla f^\star(-\sum_{i<t+1} \boldsymbol{\lambda}_i) = \sum_{i<t+1} \alpha_i \mathbf{x}_i = \sum_{i<t} \alpha_i \mathbf{x}_i + \alpha_t \mathbf{x}_t = \mathbf{w}_t + \alpha_t \mathbf{x}_t$$

- We saw that obtain a regret bound if $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{L^2}{2\sigma}$ where L is the Lipschitz constant of ℓ_t w.r.t $\|\cdot\|_\star$

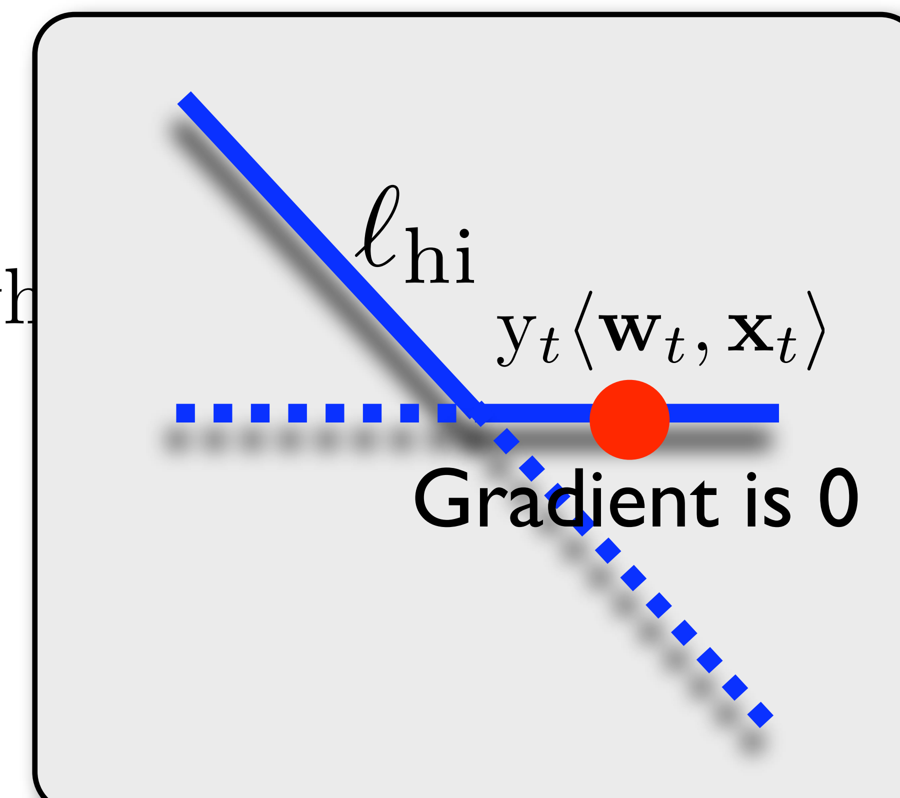
- We can use gradient descent (on the primal) to achieve sufficient increase of the dual objective:

  - Gradient descent:

    1. $\boldsymbol{\lambda}_t = -\mathbf{x}_t$ ($\boldsymbol{\lambda}_t = -\alpha_t \mathbf{x}_t$ with $\alpha_t = 1$) wh
    2. $\boldsymbol{\lambda}_t = 0$ otherwise

  - Dual increase: $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{1}{2\sigma}$

- Can we potentially make faster progress in the dual while maintaining the regret bound?

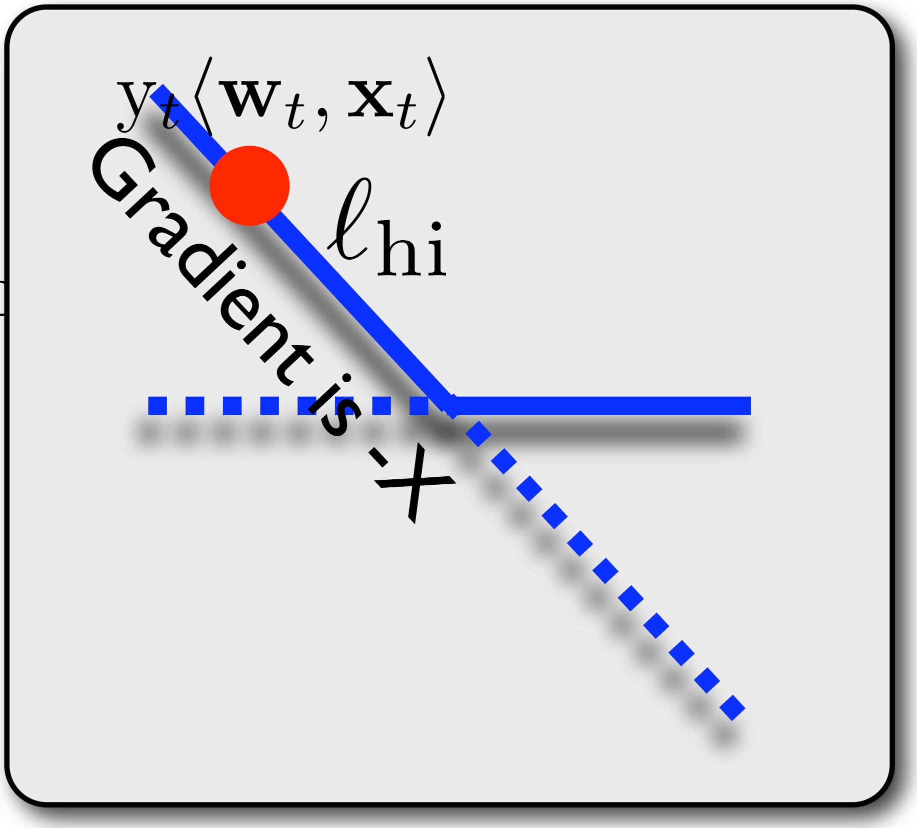


Online SVM Revisited

- Since $f^\star(\mathbf{v}) = f^\star(-\mathbf{v})$ and $\nabla f^\star(\mathbf{v}) = \mathbf{v}$,

$$\mathbf{w}_{t+1} = \nabla f^\star(-\sum_{i<t+1} \boldsymbol{\lambda}_i) = \sum_{i<t+1} \alpha_i \mathbf{x}_i = \sum_{i<t} \alpha_i \mathbf{x}_i + \alpha_t \mathbf{x}_t = \mathbf{w}_t + \alpha_t \mathbf{x}_t$$

- We saw that obtain a regret bound if $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{L^2}{2\sigma}$ where L is the Lipschitz constant of ℓ_t w.r.t $\|\cdot\|_\star$
- We can use gradient descent (on the primal) to achieve sufficient increase of the dual objective:
  - Gradient descent:
    1. $\boldsymbol{\lambda}_t = -\mathbf{x}_t$ ($\boldsymbol{\lambda}_t = -\alpha_t \mathbf{x}_t$ with $\alpha_t = 1$) wh
    2. $\boldsymbol{\lambda}_t = 0$ otherwise
  - Dual increase: $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{1}{2\sigma}$
- Can we potentially make faster progress in the dual while maintaining the regret bound?



Aggressive Dual Ascend Schemes (I)

- Locally aggressive update:

  1. Leave $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{t-1}$ intact from previous rounds
  2. $\boldsymbol{\lambda}_{t+1} = \ldots = \boldsymbol{\lambda}_T = 0$: yet to observe future examples
  3. Set $\boldsymbol{\lambda}_t = -\alpha_t \mathbf{x}_t$ to maximize the increase in the dual

- Maximizing the "instantaneous" dual w.r.t α_t is a scalar optimization problem that often can be solved analytically

- Increase in dual is at least as large as increase due to gradient descent. The locally aggressive scheme achieves at least as good a regret bound as the aggressive Perceptron

Aggressive Dual Ascend Schemes (I)

$$\boldsymbol{\lambda}_t = \arg\min_{\boldsymbol{\mu}} \mathcal{D}(\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{t-1}, \boldsymbol{\mu}, 0, \ldots, 0)$$

- Locally aggressive update:

    1. Leave $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{t-1}$ intact from previous rounds
    2. $\boldsymbol{\lambda}_{t+1} = \ldots = \boldsymbol{\lambda}_T = 0$: yet to observe future examples
    3. Set $\boldsymbol{\lambda}_t = -\alpha_t \mathbf{x}_t$ to maximize the increase in the dual

- Maximizing the "instantaneous" dual w.r.t α_t is a scalar optimization problem that often can be solved analytically

- Increase in dual is at least as large as increase due to gradient descent. The locally aggressive scheme achieves at least as good a regret bound as the aggressive Perceptron

Aggressive Dual Ascend Schemes (II)

- Follow the regularized leader (Forel):

  1. $\boldsymbol{\lambda}_{t+1} = \ldots = \boldsymbol{\lambda}_T = 0$ as before
  2. Set $\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_t$ so as to maximize the resulting dual

- Primal of dual with $\boldsymbol{\lambda}_{t+1} = \ldots, \boldsymbol{\lambda}_T = 0$ is

$$\mathcal{P}_t(\mathbf{w}) = \sigma f(\mathbf{w}) + \sum_{i=1}^{\mathbf{t}} \ell_i(\mathbf{w})$$

- Strong duality: $\mathcal{D}(\boldsymbol{\lambda}_1^\star, \ldots, \boldsymbol{\lambda}_t^\star) = \mathcal{P}_t(\mathbf{w}^\star)$

- Thus, on round t we set $\mathbf{w}_t$ to be the optimum of an instantaneous primal problem: $\mathbf{w}_t = \arg\min_{\mathbf{w}} \sigma f(\mathbf{w}) + \sum_{i=1}^{\mathbf{t}} \ell_i(\mathbf{w})$

- Increase in dual is at least as large as increase of locally aggressive update. Forel is at least as good as scheme I

Locally Aggressive Update for Online SVM

- The Fenchel dual of SVM is $\mathcal{D}(\boldsymbol{\alpha}) = \sum_{t=1}^{T} \alpha_t - \frac{1}{2\,\sigma} \left\| \sum_{t=1}^{T} \alpha_t y_t \mathbf{x}_t \right\|^2$
- We saw that obtain a regret bound if $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{L^2}{2\sigma}$ where L is the Lipschitz constant of ℓ_t w.r.t $\|\cdot\|_\star$
- We can use gradient descent (on the primal) to achieve sufficient increase of the dual objective:
  - Gradient descent: $\alpha_t = 1$ if $[1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle]_+ > 0$
  - Dual increase: $\mathcal{D}_{t+1} - D_t \geq \ell_t(\mathbf{w}_t) - \frac{1}{2\sigma}$
- Aggressively increase the dual by choosing α_t to maximize $\Delta_t = \mathcal{D}_{t+1} - \mathcal{D}_t$

Passive-Aggressive: Locally Aggr. Online SVM

- Recall once more SVM's dual: $\mathcal{D}(\boldsymbol{\alpha}) = \sum_{t=1}^{T} \alpha_t - \frac{1}{2\,\sigma} \left\| \sum_{t=1}^{T} \alpha_t y_t \mathbf{x}_t \right\|^2$
- The change in the dual due to a change of α_t

$$\begin{aligned} \Delta_t &= \left(\sum_{i \leq t} \alpha_t - \tfrac{1}{2\,\sigma} \left\| \sigma \mathbf{w}_t + \alpha_t y_t \mathbf{x}_t \right\|^2 \right) - \left(\sum_{i<t} \alpha_i - \tfrac{1}{2\,\sigma} \left\| \sigma \mathbf{w}_t \right\|^2 \right) \\ &= \alpha_t (1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle) - \alpha_t^2 \frac{\|\mathbf{x}_t\|^2}{2\,\sigma} \end{aligned}$$

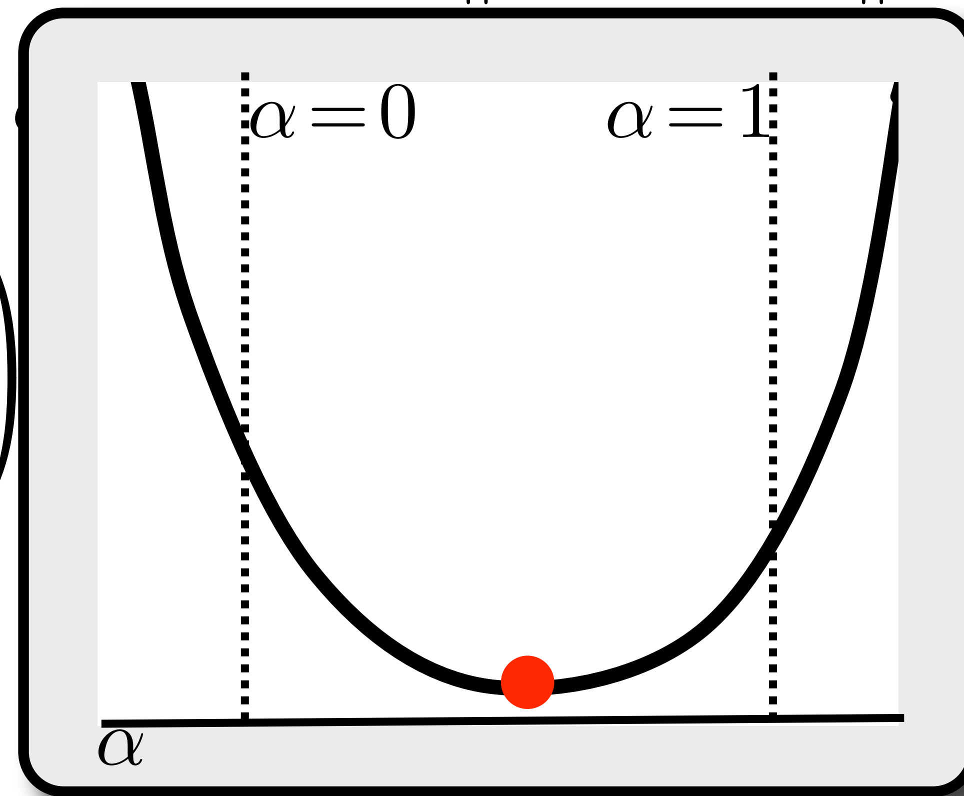
- Quadratic equation in α_t with boundary constraints $\alpha_t \in [0, 1]$

$$\alpha_t^\star = \max \left\{ 0, \min \left\{ 1, \sigma \frac{1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle)}{\|\mathbf{x}_t\|^2} \right\} \right\}$$

- Passive-Aggressive: if margin ≥ 1 do nothing otherwise use $\alpha_t^\star$

Passive-Aggressive: Locally Aggr. Online SVM

- Recall once more SVM's dual: $\mathcal{D}(\boldsymbol{\alpha}) = \sum_{t=1}^{T} \alpha_t - \frac{1}{2\,\sigma} \left\| \sum_{t=1}^{T} \alpha_t y_t \mathbf{x}_t \right\|^2$
- The change in the dual due to a change of



$$
\begin{aligned}
\Delta_t &= \left(\sum_{i \leq t} \alpha_t - \tfrac{1}{2\,\sigma} \|\sigma \mathbf{w}_t + \alpha_t y_t \mathbf{x}_t\|^2 \right) \\
&= \alpha_t (1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle) - \alpha_t^2 \frac{\|\mathbf{x}_t\|^2}{2\,\sigma}
\end{aligned}
$$

- Quadratic equation in α_t with boundary constraints $\alpha_t \in [0, 1]$

$$\alpha_t^\star = \max \left\{ 0, \min \left\{ 1, \sigma \frac{1 - y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle)}{\|\mathbf{x}_t\|^2} \right\} \right\}$$

- Passive-Aggressive: if margin ≥ 1 do nothing otherwise use $\alpha_t^\star$

Online SVM by Following the Leader

- Instantaneous primal $\mathcal{P}_t(\mathbf{w}) = \sigma/2\|\mathbf{w}\|^2 + \sum_{i=1}^{t} \ell_i(\mathbf{w})$
- Dual of $\mathcal{P}_t(\mathbf{w})$

$$\mathcal{D}(\alpha_1, \ldots, \alpha_t | \alpha_{t+1} = \ldots = 0) = \sum_{i=1}^{t} \alpha_i - \frac{1}{2\sigma} \left\| \sum_{i=1}^{t} \alpha_i y_i \mathbf{x}_i \right\|^2$$

- Follow the regularized leader - **Forel**:
  $(\alpha_1^\star, \ldots, \alpha_t^\star) = \arg \min_{\alpha_1, \ldots, \alpha_t} \mathcal{D}(\alpha_1, \ldots, \alpha_t | \alpha_{t+1} = \ldots = 0)$
- From strong duality

$$\mathbf{w}_t^\star = \arg\min_{\mathbf{w}} \mathcal{P}_t(\mathbf{w}) \;\Leftrightarrow\; \mathbf{w}_t^\star = \sum_{i=1}^{t} \alpha_i^\star y_i \mathbf{x}_i$$

- The regret of FOREL is at least as good as PA's regret

Entropic Regularization

Motivation – Prediction with expert advice:

- Learner receives a vector $\mathbf{x}_t = (x_t^1, \ldots, x_t^d) \in [-1,1]^d$ of experts advice
- Learner needs to predict a target $\hat{y}_t \in \mathbb{R}$
- Environment gives correct target $y_t \in \mathbb{R}$
- Learner suffers loss $|y_t - \hat{y}_t|$
- Goal: predict almost as well as best committee of experts $\sum_t |y_t - \hat{y}_t| - \sum_t |y_t - \langle \mathbf{w}^\star, \mathbf{x}^t \rangle| \stackrel{!}{=} o(T)$

Modeling:

- S is the d-dimensional probability simplex
- Loss functions: $\ell_t(\mathbf{w}) = |y_t - \langle \mathbf{w}, \mathbf{x}_t \rangle|$

Entropic Regularization (cont.)

Prediction with expert advice – regret:

- Consider working with $f(\mathbf{w}) = \frac{\sigma}{2} \|\mathbf{w}\|^2$
- Regret is $L\, W\, \sqrt{T}$ where:
  - S is the probability simplex and thus $W = \max_{\mathbf{w} \in \Delta} \|\mathbf{w}\| = 1$
  - Lipschitz constant is $L = \max \|\mathbf{x}\| = \sqrt{d}$
  - Regret is $O(\sqrt{d\, T})$
- Is this the best we can do in terms of dependency in d?

Entropic Regularization (cont.)

Prediction with expert advice – Entropic regularization:

- Consider working with
$$f(\mathbf{w}) = \sum_{j=1}^{n} w_j \log\left(\frac{w_j}{1/n}\right) = \log(n) + \sum_j w_j \log(w_j)$$
- $\|\cdot\|_1, \|\cdot\|_\infty$ for assessing convexity and Lipschitz constants
- f is 1-strongly convex w.r.t. $\|\cdot\|_1$
- Regret is $L\,W\,\sqrt{T}$ where:
  - S is the probability simplex and thus $W = \max_{\mathbf{w}\in\Delta} f(\mathbf{w}) = \log(n)$
  - Lipschitz constant of $\ell_t(\mathbf{w}) = |y_t - \langle \mathbf{w}, \mathbf{x}_t\rangle|$ is $L = 1$ since $\|\mathbf{x}_t\|_\infty \le 1$
  - Regret is $O(\sqrt{\log(d)\,T})$

Entropic Regularization → Multiplicative PA

- Generalized hinge loss $[\gamma - y_t \langle \mathbf{w}, \mathbf{x}_t \rangle]_+$
- Use $f(\mathbf{w}) = \log(n) + \sum_{j=1}^{n} w_j \log(w_j)$ ($\mathbf{w}$ in prob. simplex)

  Fenchel dual of f: $f^\star(\boldsymbol{\lambda}) = \log \left(\frac{1}{n} \sum_{j=1}^{n} e^{\lambda_j} \right)$

- Primal problem

$$\mathcal{P}(\mathbf{w}) = \sigma \left(\log(n) + \sum_{j=1}^{n} w_j \log(w_j) \right) + \sum_{t=1}^{T} [\gamma - y_t \langle \mathbf{w}, \mathbf{x}_t \rangle]_+$$

- Define $\boldsymbol{\theta} = \sum_i \boldsymbol{\lambda}_i = \frac{1}{\sigma} \sum_i \alpha_i y_i \mathbf{x}_i$ to write dual problem

$$\mathcal{D}(\boldsymbol{\alpha}) = \gamma \sum_i \alpha_i - \sigma \log \left(\frac{1}{n} \sum_{j=1}^{n} e^{\theta_j} \right) \quad \text{s.t. } \alpha_i \in [0, 1]$$

PA Update with Entropic Regularization

- Find α_t with maximal local dual increase (closed form for maximal increase if $\mathbf{x}_t \in \{-1, 0, 1\}^n$)

$$\alpha_t^\star = \arg\max_{\alpha \in [0,1]} \quad \gamma\alpha - \sigma \log\left(\frac{1}{n}\sum_{i=1}^{t-1} \alpha_i y_i \mathbf{x}_i + \alpha y_t \mathbf{x}_t\right)$$

- Define $\boldsymbol{\theta}_t = \dfrac{1}{\sigma}\displaystyle\sum_{i=1}^{t} \alpha_i^\star y_i \mathbf{x}_i$

- Update $\mathbf{w}_t = \nabla f^\star(\boldsymbol{\theta}_t)$

$$\mathbf{w}_{t,j} = \exp(\theta_{t,j})/Z_t \quad \text{where} \quad Z_t = \sum_r \exp(\theta_{t,r})$$

- Use $\mathbf{w}_{t,j} \sim \exp(\theta_{t,j})$ to obtain a multiplicative update

$$w_{t+1,j} = w_{t,j} \exp(\alpha_t^\star y_t x_{t,j})/\tilde{Z}_t$$

PA Update with Entropic Regularization

- Find α_t with maximal local dual increase (closed form for maximal increase if $\mathbf{x}_t \in \{-1, 0, 1\}^n$)

$$\alpha_t^\star = \arg \max_{\alpha \in [0,1]} \quad \gamma\alpha - \sigma \log \left(\frac{1}{n} \sum_{i=1}^{t-1} \alpha_i y_i \mathbf{x}_i + \alpha y_t \mathbf{x}_t \right)$$

- Define $\boldsymbol{\theta}_t = \dfrac{1}{\sigma} \displaystyle\sum_{i=1}^{t} \alpha_i^\star y_i \mathbf{x}_i$

Regret is $O(\sqrt{\log(d)\,T})$

- Update $\mathbf{w}_t = \nabla f^\star(\boldsymbol{\theta}_t)$

$$\mathbf{w}_{t,j} = \exp(\theta_{t,j})/Z_t \quad \text{where} \quad Z_t = \sum_r \exp(\theta_{t,r})$$

- Use $\mathbf{w}_{t,j} \sim \exp(\theta_{t,j})$ to obtain a multiplicative update

$$w_{t+1,j} = w_{t,j} \exp(\alpha_t^\star y_t x_{t,j})/\tilde{Z}_t$$

Online Logistic Regression

- Loss: $\log\left(1 + \exp(-y_t \langle \mathbf{w}, \mathbf{x}_t \rangle)\right)$
- Primal problem
$$\sigma f(\mathbf{w}) + \sum_{t=1}^{T} \log\left(1 + \exp(-y_t \langle \mathbf{w}, \mathbf{x}_t \rangle)\right)$$
- Define $\boldsymbol{\theta} = \sum_i (\alpha_i / \sigma) y_i \mathbf{x}_i$
- Dual problem (for $f(\mathbf{w}) = D_{\mathrm{KL}}(\mathbf{w} \| \mathbf{u})$)
$$\mathcal{D}(\boldsymbol{\alpha}) = \sum_t H(\alpha_t) - \sigma \log\left(\frac{1}{n} \sum_{j=1}^{n} e^{\theta_j}\right)$$
- Find α_t with sufficient dual increase using binary search for $\alpha_t \in [0,1]$
- Update (Z_t ensures $\mathbf{w}_{t+1} \in \Delta^n$)
$$w_{t+1,j} \;=\; w_{t,j}\, e^{(\alpha_t/\sigma)\, y_t\, x_{t,j}} / Z_t$$

Online Logistic Regression

- Loss: $\log\left(1+\exp(-y_t\langle \mathbf{w},\mathbf{x}_t\rangle)\right)$
- Primal problem

$$\sigma f(\mathbf{w}) + \sum_{t=1}^{T} \log\left(1+\exp(-y_t\langle \mathbf{w},\mathbf{x}_t\rangle)\right)$$

- Define $\boldsymbol{\theta} = \sum_i (\alpha_i/\sigma) y_i \mathbf{x}_i$
- Dual problem (for $f(\mathbf{w}) = D_{\mathrm{KL}}(\mathbf{w}\|\mathbf{u})$)

$$\mathcal{D}(\boldsymbol{\alpha}) = \sum_t H(\alpha_t) - \sigma \log\left(\frac{1}{n}\sum_{j=1}^{n} e^{\theta_j}\right)$$

- Find α_t with sufficient dual increase using binary search for $\alpha_t \in [0,1]$
- Update (Z_t ensures $\mathbf{w}_{t+1} \in \Delta^n$)

$$w_{t+1,j} \;=\; w_{t,j}\; e^{(\alpha_t/\sigma)\, y_t\, x_{t,j}} / Z_t$$

Same update form as multiplicative PA for SVM

Online Logistic Regression

- Loss: $\log\left(1 + \exp(-y_t \langle \mathbf{w}, \mathbf{x}_t \rangle)\right)$
- Primal problem
$$\sigma f(\mathbf{w}) + \sum_{t=1}^{T} \log\left(1 + \exp(-y_t \langle \mathbf{w}, \mathbf{x}_t \rangle)\right)$$
- Define $\boldsymbol{\theta} = \sum_i (\alpha_i/\sigma) y_i \mathbf{x}_i$
- Dual problem (for $f(\mathbf{w}) = D_{\mathrm{KL}}(\mathbf{w} \| \mathbf{u})$)
$$\mathcal{D}(\boldsymbol{\alpha}) = \sum_t H(\alpha_t) - \sigma \log\left(\frac{1}{n} \sum_{j=1}^{n} e^{\theta_j}\right)$$
- Find α_t with sufficient dual increase using binary search for $\alpha_t \in [0, 1]$
- Update (Z_t ensures $\mathbf{w}_{t+1} \in \Delta^n$)
$$w_{t+1,j} \;=\; w_{t,j}\, e^{(\alpha_t/\sigma)\, y_t\, x_{t,j}} / Z_t$$

Online Logistic Regression

- Loss: $\log\left(1+\exp(-y_t\langle \mathbf{w},\mathbf{x}_t\rangle)\right)$
- Primal problem

$$\sigma f(\mathbf{w}) + \sum_{t=1}^{T} \log\left(1+\exp(-y_t\langle \mathbf{w},\mathbf{x}_t\rangle)\right)$$

Regret is $O(\sqrt{\log(d)\,T})$

- Define $\boldsymbol{\theta} = \sum_i (\alpha_i/\sigma) y_i \mathbf{x}_i$
- Dual problem (for $f(\mathbf{w}) = D_{\mathrm{KL}}(\mathbf{w}\|\mathbf{u})$)

$$\mathcal{D}(\boldsymbol{\alpha}) = \sum_t H(\alpha_t) - \sigma \log\left(\frac{1}{n}\sum_{j=1}^{n} e^{\theta_j}\right)$$

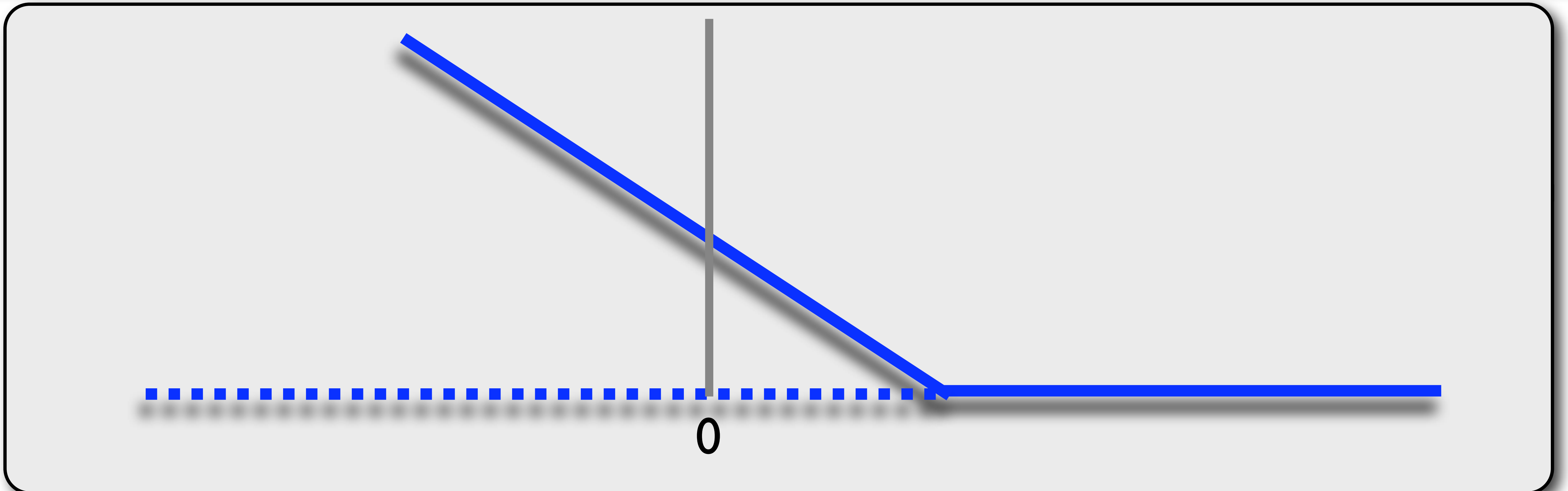
- Find α_t with sufficient dual increase using binary search for $\alpha_t \in [0,1]$
- Update (Z_t ensures $\mathbf{w}_{t+1} \in \Delta^n$)

$$w_{t+1,j} \;=\; w_{t,j}\; e^{(\alpha_t/\sigma)\, y_t\, x_{t,j}} / Z_t$$

Back to "Classical" Perceptron

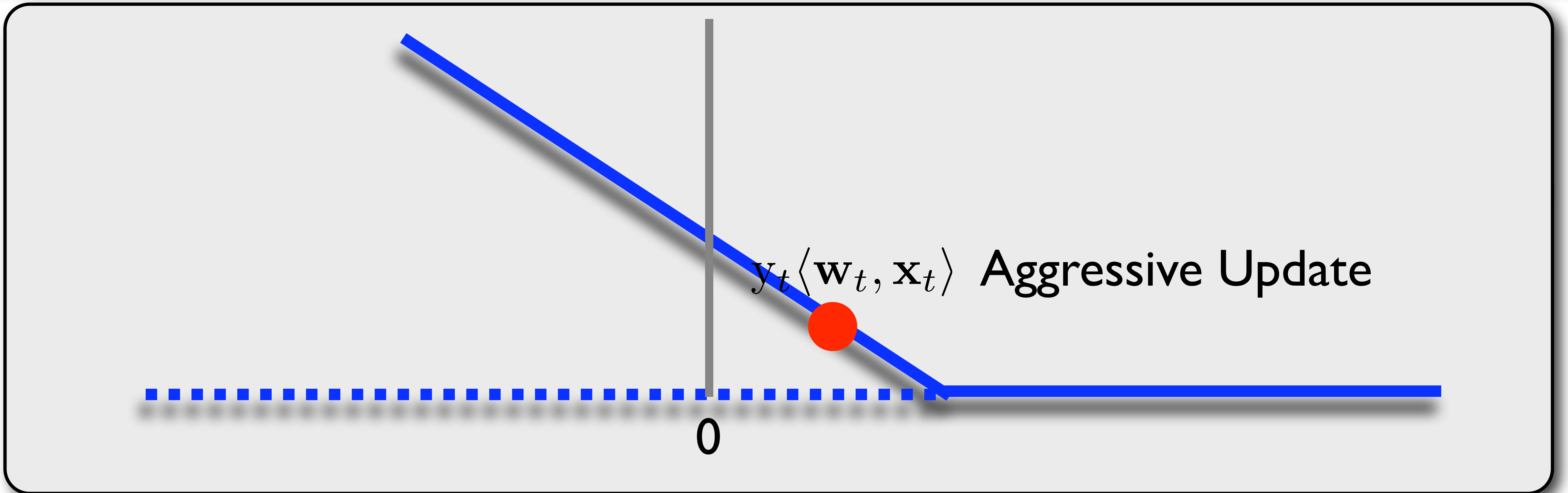
- Focus on rounds with mistakes $(y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle \leq 0)$
- Assume norm of instances bounded by 1 $(\forall t : \|\mathbf{x}_t\| \leq 1)$
- Recall $\Delta_t = \alpha_t - \frac{1}{2}(\alpha_t y_t \langle \mathbf{w}_t, \mathbf{x}_t \rangle + \alpha_t^2 \|\mathbf{x}_t\|^2 / \sigma)$
- From assumptions $\Delta_t \geq \alpha_t - \frac{1}{2\sigma}\alpha_t^2$
- Two version of the Perceptron:
    - Aggressive Perceptron:
      $\alpha_t = 1$ whenever $\ell_t(\mathbf{w}_t) > 0$
    - Scaled version of classical Perceptron:
      $\alpha_t = 1$ only when $\ell_t(\mathbf{w}_t) \geq 1$
- Upon an update $\Delta_t \geq 1 - \frac{1}{2\sigma}$ for both versions

Back to "Classical" Perceptron



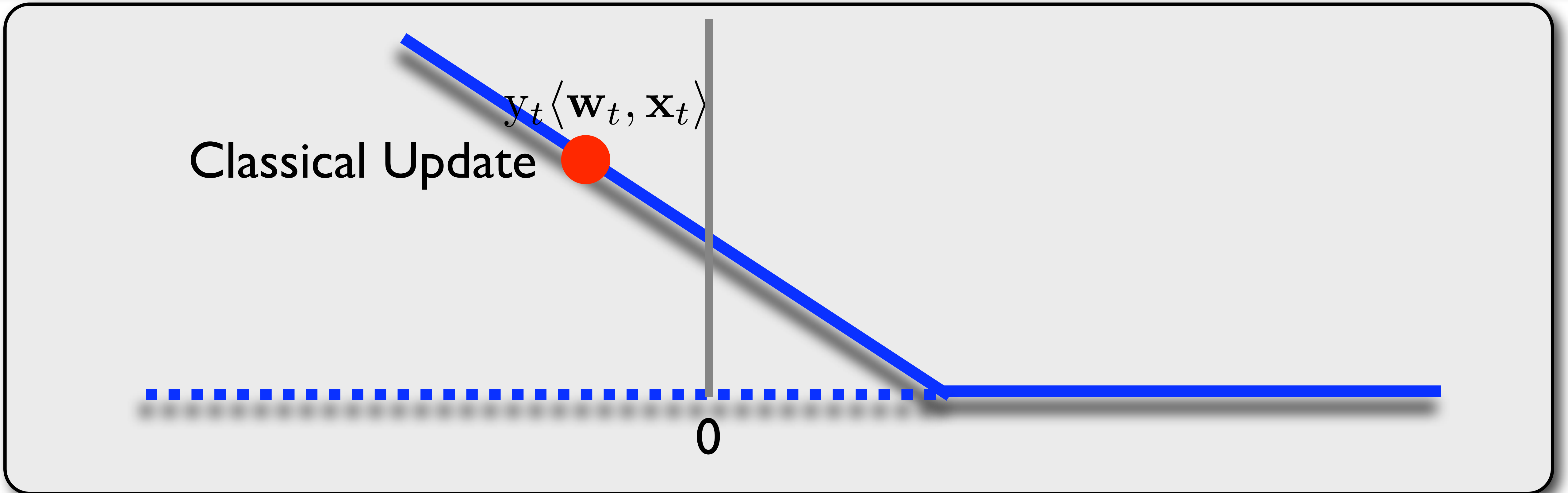
- Two version of the Perceptron:
    - Aggressive Perceptron:
      $\alpha_t = 1$ whenever $\ell_t(\mathbf{w}_t) > 0$
    - Scaled version of classical Perceptron:
      $\alpha_t = 1$ only when $\ell_t(\mathbf{w}_t) \geq 1$
- Upon an update $\Delta_t \geq 1 - \frac{1}{2\sigma}$ for both versions

Back to "Classical" Perceptron



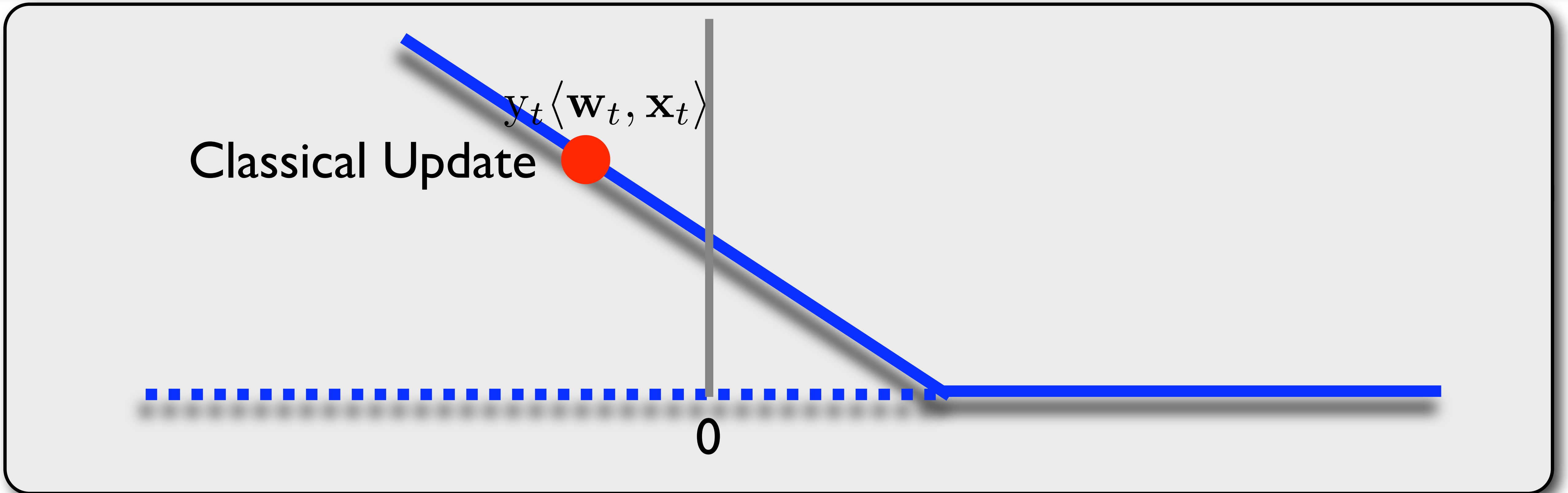
- Two version of the Perceptron:
  - Aggressive Perceptron:
    $\alpha_t = 1$ whenever $\ell_t(\mathbf{w}_t) > 0$
  - Scaled version of classical Perceptron:
    $\alpha_t = 1$ only when $\ell_t(\mathbf{w}_t) \geq 1$
- Upon an update $\Delta_t \geq 1 - \frac{1}{2\sigma}$ for both versions

Back to "Classical" Perceptron



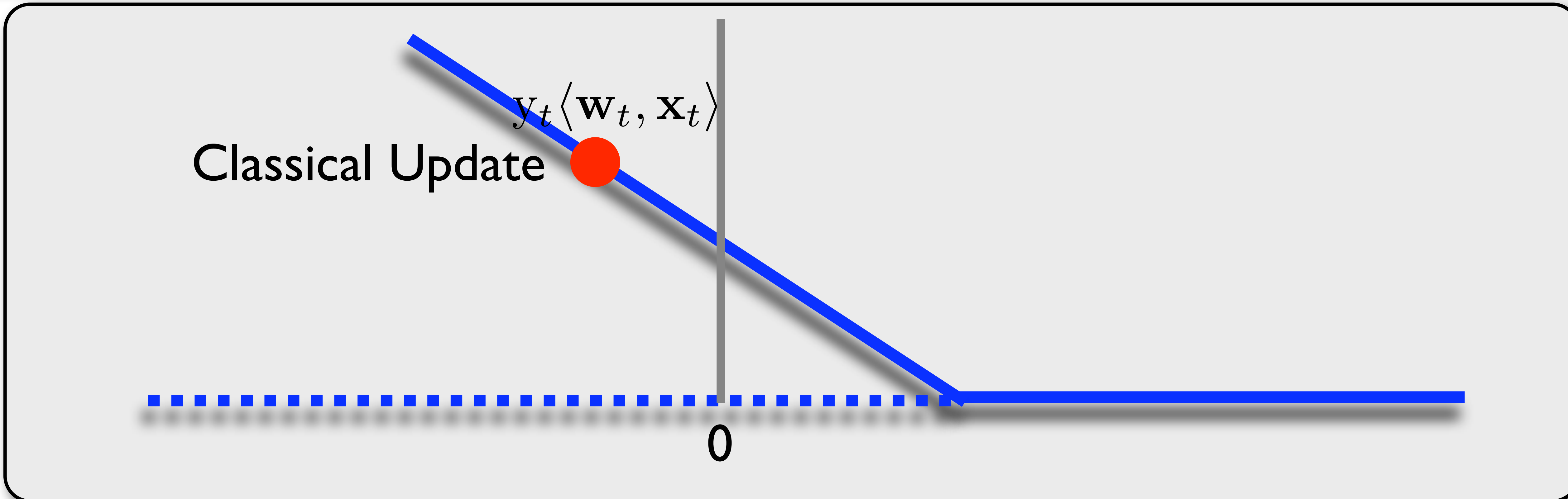
- Two version of the Perceptron:
    - Aggressive Perceptron:
      $\alpha_t = 1$ whenever $\ell_t(\mathbf{w}_t) > 0$
    - Scaled version of classical Perceptron:
      $\alpha_t = 1$ only when $\ell_t(\mathbf{w}_t) \geq 1$
- Upon an update $\Delta_t \geq 1 - \frac{1}{2\sigma}$ for both versions

Back to "Classical" Perceptron



- Two version of the Perceptron:
    - Aggressive Perceptron: Achieves a Regret Bound
      $\alpha_t = 1$ whenever $\ell_t(\mathbf{w}_t) > 0$
    - Scaled version of classical Perceptron:
      $\alpha_t = 1$ only when $\ell_t(\mathbf{w}_t) \geq 1$
- Upon an update $\Delta_t \geq 1 - \frac{1}{2\sigma}$ for both versions

Back to "Classical" Perceptron



- Two version of the Perceptron:
  - Aggressive Perceptron:
    $\alpha_t = 1$ whenever $\ell_t(\mathbf{w}_t) > 0$

    Achieves a Mistake Bound
  - Scaled version of classical Perceptron:
    $\alpha_t = 1$ only when $\ell_t(\mathbf{w}_t) \geq 1$
- Upon an update $\Delta_t \geq 1 - \frac{1}{2\sigma}$ for both versions

Universality of Classical Perceptron

- Resulting update - "scaled" Perceptron:

$$\mathbf{w}_{t+1} = \begin{cases} \mathbf{w}_t + \frac{1}{\sigma}\, y_t \mathbf{x}_t & \text{if } \langle \mathbf{w}_t, \mathbf{x}_t \rangle y_t \leq 0 \\ \mathbf{w}_t & \text{otherwise} \end{cases}$$

- Use weak duality to obtain that $\varepsilon\left(1 - \frac{1}{2\sigma}\right) \leq \sum_t \Delta_t \leq \mathcal{P}(\mathbf{w}^\star)$
- Performance the same regardless of choice of σ
- Choose σ so as to minimize regret bound

$$\varepsilon(T) \leq \sum_{t=1}^{T} \ell_{\text{hi}}(\langle \mathbf{u}, \mathbf{x}_t \rangle, y_t) \; + \; \|\mathbf{u}\| \sqrt{\varepsilon(T)}$$

- Bound implies that

$$\varepsilon(T) \leq \mathcal{L}^\star + \|\mathbf{u}\| \sqrt{\mathcal{L}^\star} + \|\mathbf{u}\|^2 \;\; \text{where} \;\; \mathcal{L}^\star = \sum_t \ell_{\text{hi}}\left(\langle \mathbf{u}, \mathbf{x}_t \rangle, y_t\right)$$

Universality of Classical Perceptron

- Resulting update - "scaled" Perceptron:

$$\mathbf{w}_{t+1} = \begin{cases} \mathbf{w}_t + \frac{1}{\sigma} y_t \mathbf{x}_t & \text{if } \langle \mathbf{w}_t, \mathbf{x}_t \rangle y_t \le 0 \\ \mathbf{w}_t & \text{otherwise} \end{cases}$$

- Use weak duality to obtain that $\varepsilon \left(1 - \frac{1}{2\sigma}\right) \le \sum_t \Delta_t \le \mathcal{P}(\mathbf{w}^\star)$
- Performance the same regardless of choice of σ
- Choose σ so as to minimize regret bound

$$\varepsilon(T) \le \sum_{t=1}^{T} \ell_{\mathrm{hi}}(\langle \mathbf{u}, \mathbf{x}_t \rangle, y_t) \; + \; \|\mathbf{u}\| \sqrt{\varepsilon(T)}$$

Perceptron is approximate **universal** online SVM

- Bound implies that

$$\varepsilon(T) \le \mathcal{L}^\star + \|\mathbf{u}\| \sqrt{\mathcal{L}^\star} + \|\mathbf{u}\|^2 \quad \text{where} \quad \mathcal{L}^\star = \sum_t \ell_{\mathrm{hi}} \left(\langle \mathbf{u}, \mathbf{x}_t \rangle, y_t\right)$$

Universality of Classical Perceptron

- Resulting update - "scaled" Perceptron:

$$\mathbf{w}_{t+1} = \begin{cases} \mathbf{w}_t + \frac{1}{\sigma}\, y_t \mathbf{x}_t & \text{if } \langle \mathbf{w}_t, \mathbf{x}_t \rangle y_t \le 0 \\ \mathbf{w}_t & \text{otherwise} \end{cases}$$

- Use weak duality to obtain that $\varepsilon \left(1 - \frac{1}{2\sigma}\right) \le \sum_t \Delta_t \le \mathcal{P}(\mathbf{w}^\star)$
- Performance the same regardless of choice of σ
- Choose σ so as to minimize regret bound

$$\varepsilon(T) \le \sum_{t=1}^{T} \ell_{\mathrm{hi}}(\langle \mathbf{u}, \mathbf{x}_t \rangle, y_t) \; + \; \|\mathbf{u}\| \sqrt{\varepsilon(T)}$$

- Bound implies that

$$\varepsilon(T) \le \mathcal{L}^\star + \|\mathbf{u}\| \sqrt{\mathcal{L}^\star} + \|\mathbf{u}\|^2 \quad \text{where} \quad \mathcal{L}^\star = \sum_t \ell_{\mathrm{hi}} \left(\langle \mathbf{u}, \mathbf{x}_t \rangle, y_t \right)$$

Part IV:
A Case Study:
Online Email Categorization

The Task - Email Categorization

- On each round:
  - Receive an email message
  - Recommend the user a folder to which this email should go
  - Pay a unit loss if user does not agree with prediction
  - Learn the "true" folder the email should go to
- Goal
  - Minimize cumulative loss

Modeling (highlights)

- Feature representation
  - Represent email as bag-of-words (d-dimensional binary vectors)
  - Multi-vector multiclass construction
- The loss function
  - The 0-1 loss function is not convex. Use hinge-loss as surrogate
- The regularization
  - Euclidean & Entropic
- Dual update
  - Three dual update schemes

Modeling: Multiple Vector Construction

Email

... Brush the eggplant slices with olive oil and season with pepper.
Toss the peppers with a little olive oil. Place both on the ...

$$\mathbf{x}_t \;=\; [1\;,\;0\;,\;0\;,\;1\;,\;0\;,\;\ldots\;]$$

oil

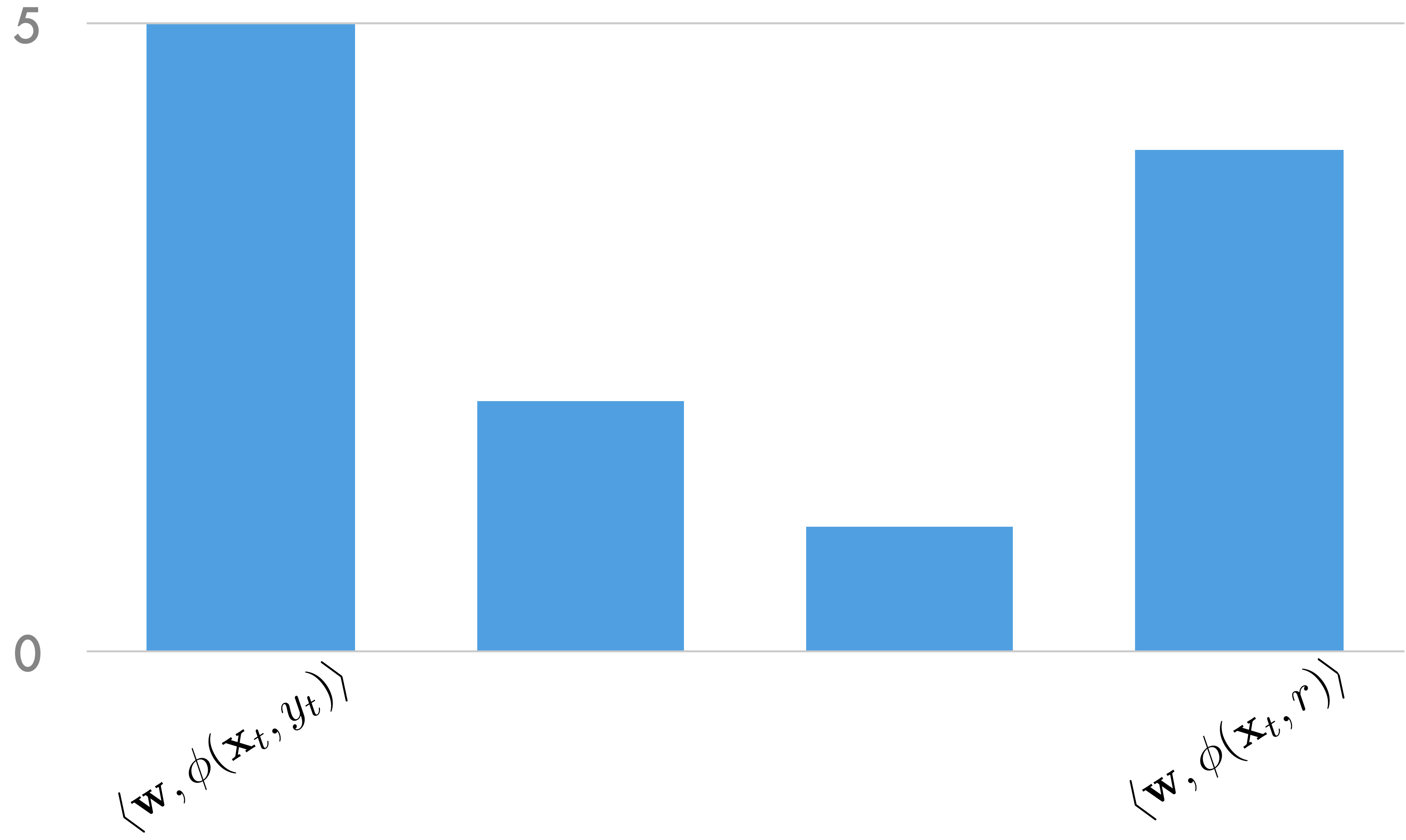
$$\phi(\mathbf{x}_t, r) \;=\; [\mathbf{0}\;,\;\ldots\;,\;\mathbf{0}\;,\;\mathbf{x}_t\;,\;\mathbf{0}\;,\;\ldots\;,\mathbf{0}]$$

r block

Prediction : $\quad \hat{y}_t \;=\; \max_r \langle \mathbf{w}, \phi(\mathbf{x}_t, r) \rangle$

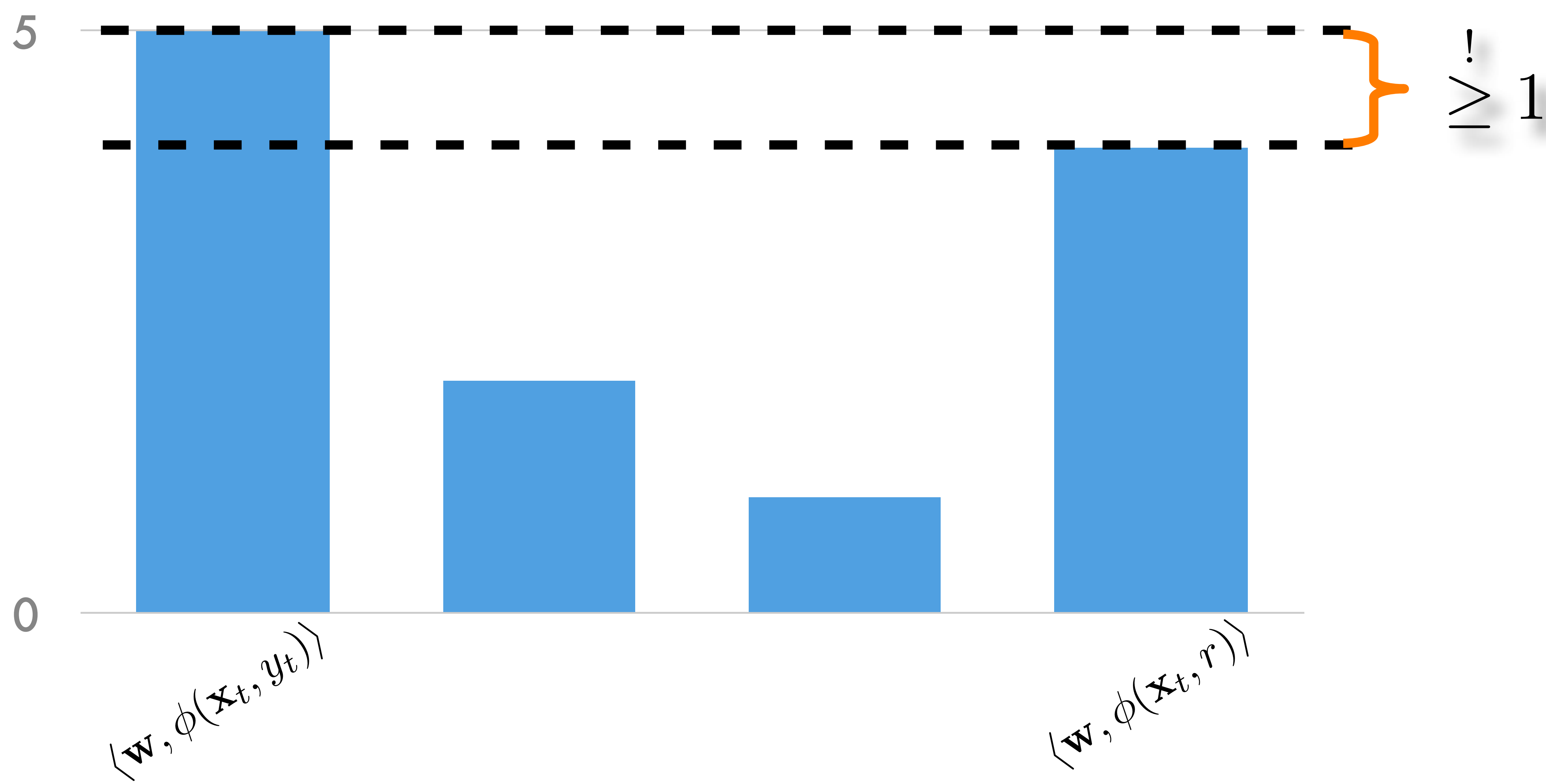
Modeling: Loss Functions

$$\ell_t(\mathbf{w}) \;=\; \max_{r \neq y_t} 1 - \langle \mathbf{w}, \phi(\mathbf{x}_t, y_t) - \phi(\mathbf{x}_t, r)\rangle \;\geq\; \ell_{0-1}(\hat{y}_t, y_t)$$



Modeling: Loss Functions

$$\ell_t(\mathbf{w}) \;=\; \max_{r \neq y_t} 1 - \langle \mathbf{w}, \phi(\mathbf{x}_t, y_t) - \phi(\mathbf{x}_t, r) \rangle \;\geq\; \ell_{0-1}(\hat{y}_t, y_t)$$



Modeling: Regularization

- Euclidean regularization $f(\mathbf{w}) = \frac{\sigma}{2}\|\mathbf{w}\|_2^2$
- Entropic regulaization $f(\mathbf{w}) = \sigma \sum_i w_i \log(d\, w_i)$

Expected Performance

- Recall the regret bounds we derived
    - Euclidean: $(\max_t \|\mathbf{x}_t\|_2)\, \|\mathbf{w}^\star\|_2 \sqrt{T}$
    - Entropic: $(\max_t \|\mathbf{x}_t\|_\infty)\, \|\mathbf{w}^\star\|_1 \sqrt{\log(d)\, T}$
- Let s be the length of the longest email
- Let r be the number of non-zero elements of $\mathbf{w}^\star$
- Then, $\frac{\text{Entropic}}{\text{Euclidean}} \leq \sqrt{\frac{r\ \log(d)}{s}}$

Modeling: Dual Update Schemes

- DA1: Fixed sub-gradient

$$\boldsymbol{\lambda}_t = \mathbf{v}_t \in \partial \ell_t(\mathbf{w}_t)$$

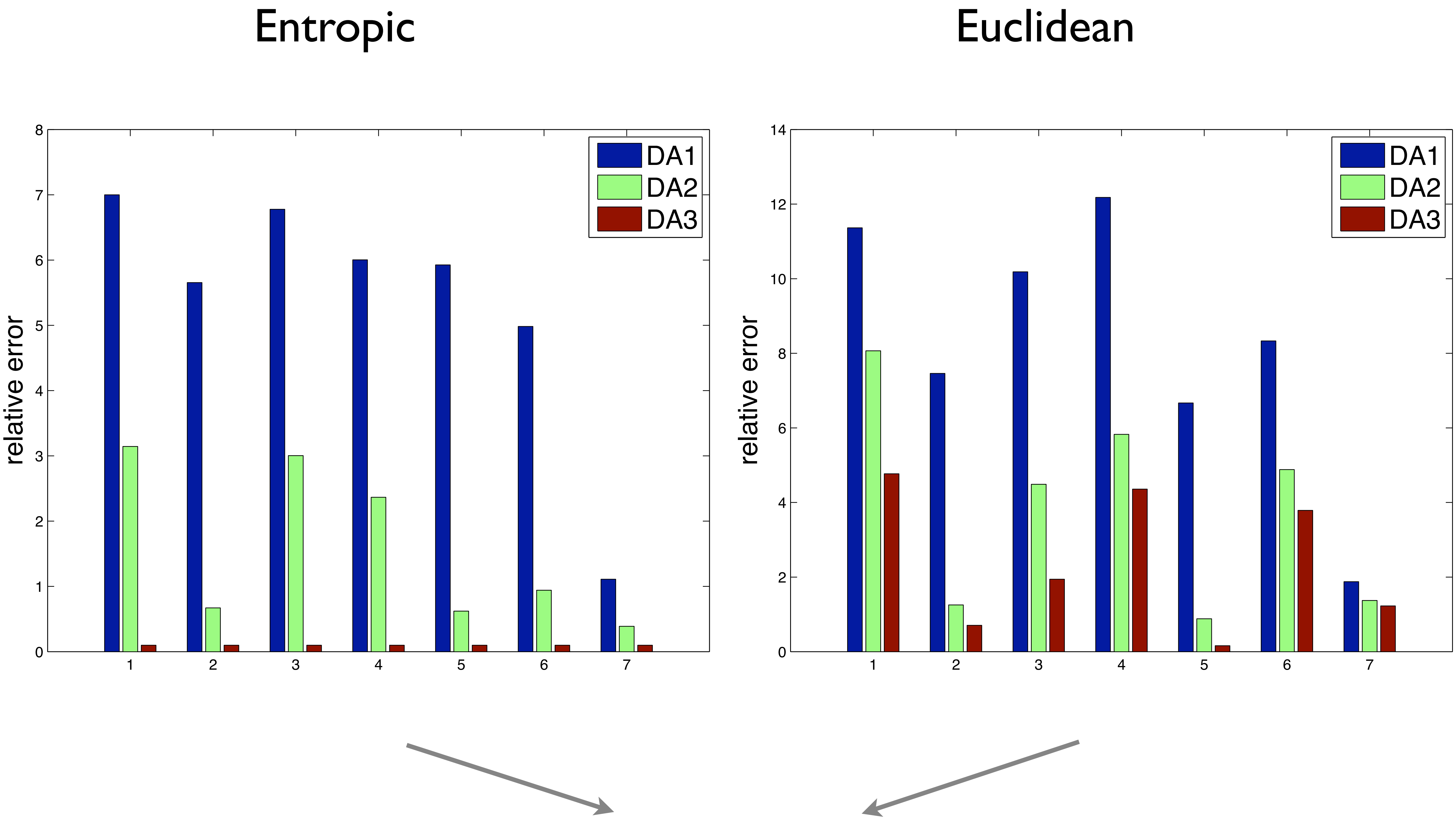
- DA2: Sub-gradient with optimal step size

$$\boldsymbol{\lambda}_t = \alpha_t \mathbf{v}_t \quad \text{where} \quad \alpha_t = \underset{\alpha}{\operatorname{argmax}}\, \mathcal{D}(\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{t-1}, \alpha\, \mathbf{v}_t, 0, \ldots)$$

- DA3: Optimizing current dual vector

$$\boldsymbol{\lambda}_t = \arg\max_{\boldsymbol{\lambda}} \mathcal{D}(\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_{t-1}, \boldsymbol{\lambda}, 0, \ldots)$$

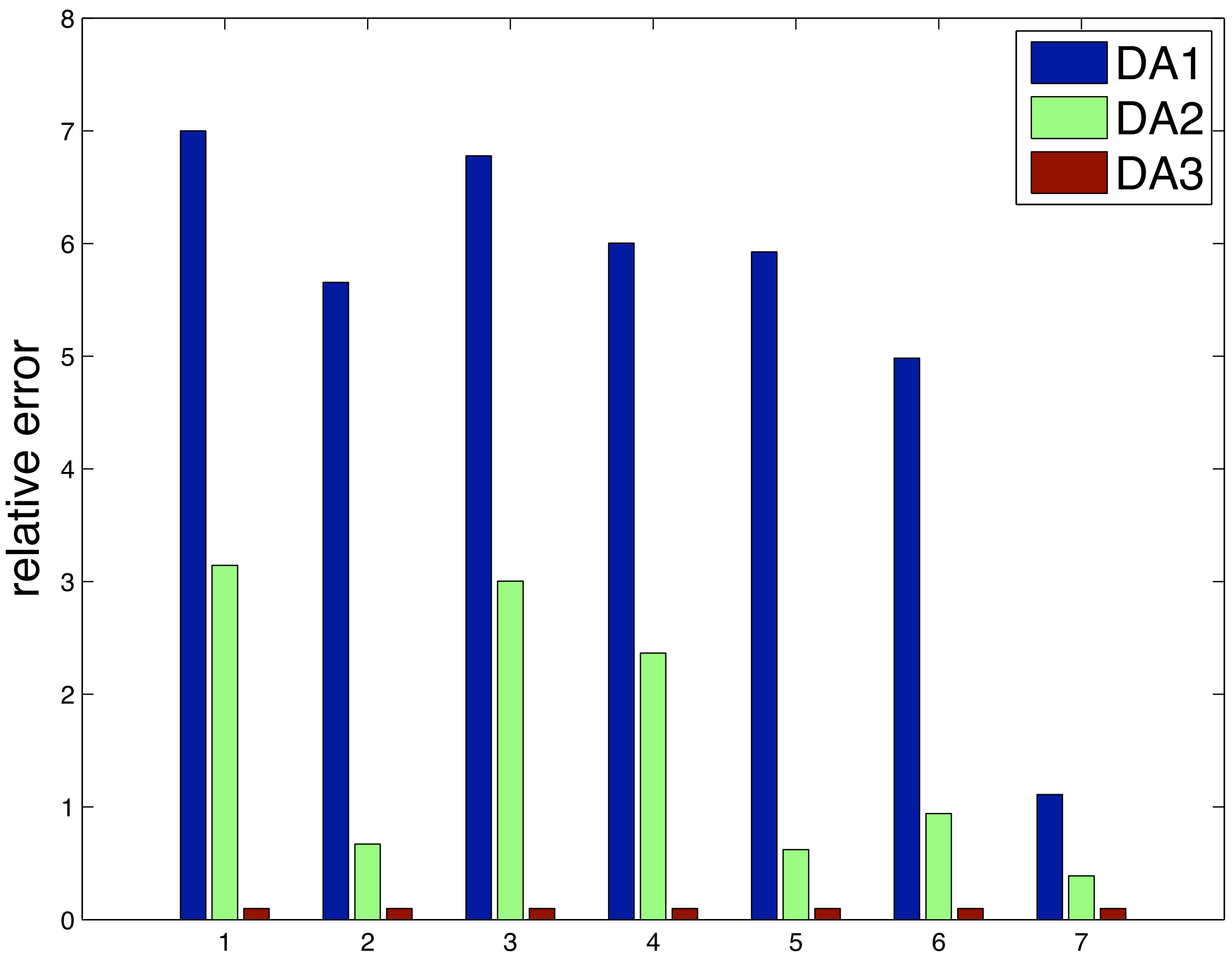
Results: 3 dual updates



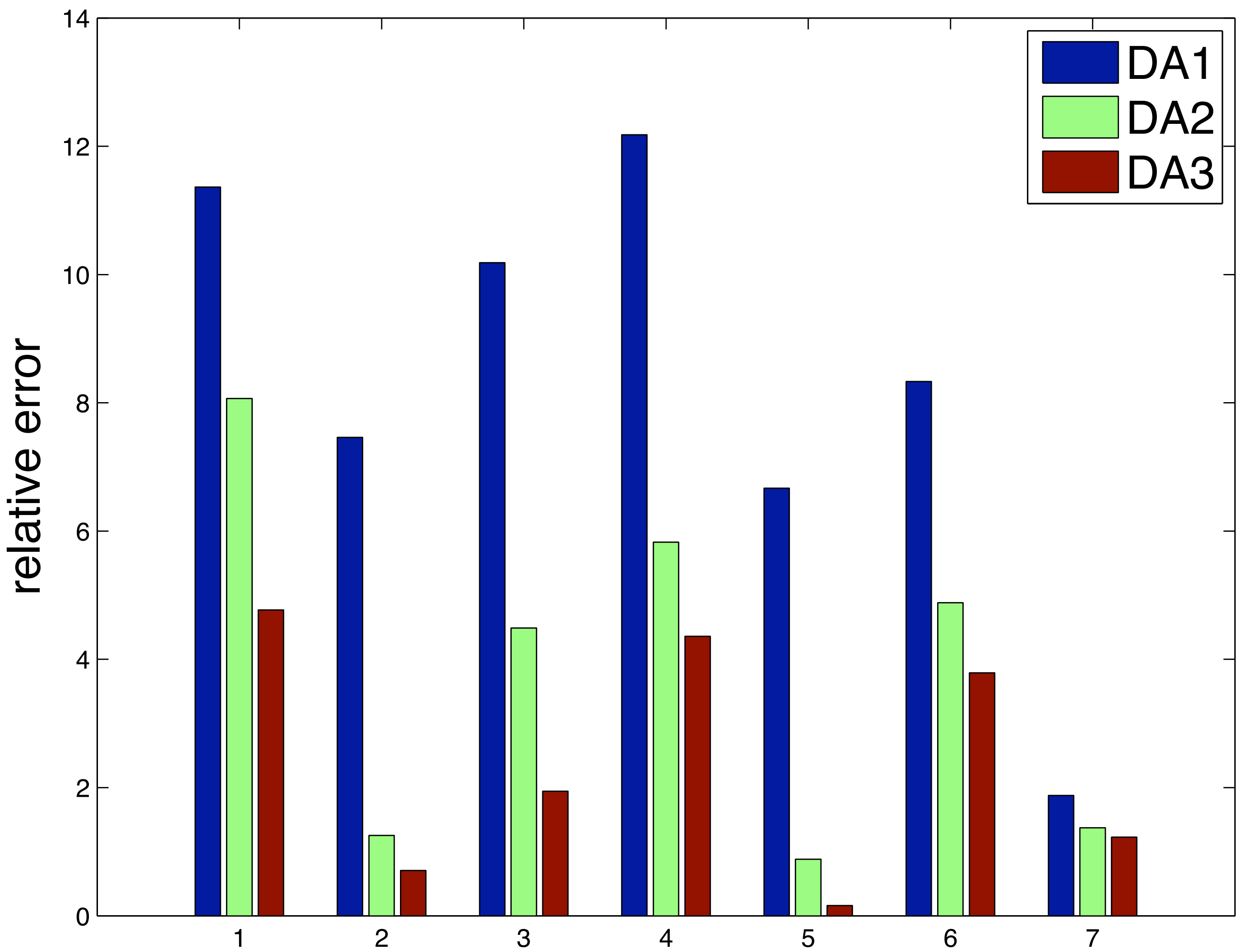
7 different users from the Enron data set

Results: 3 dual updates

Entropic



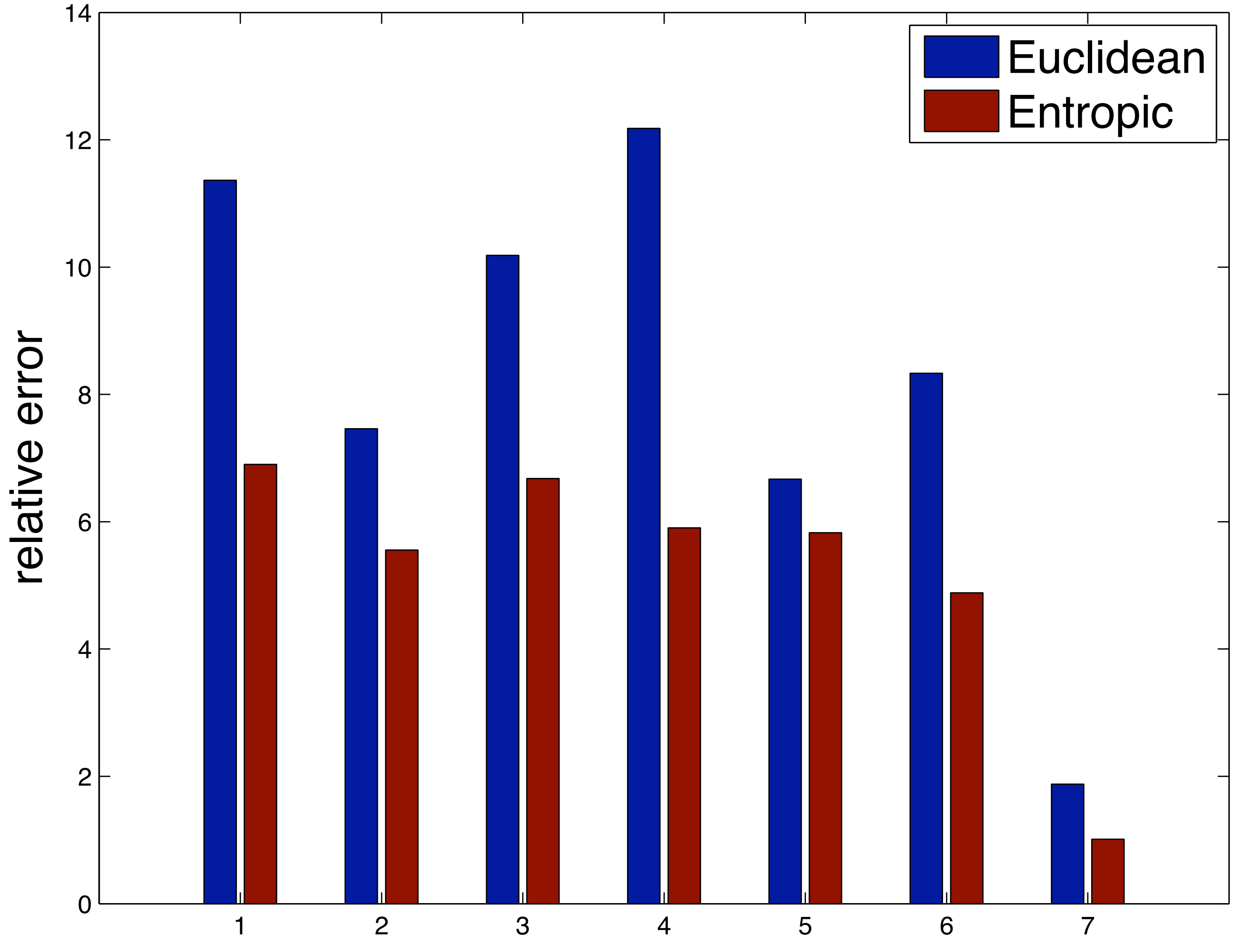
Euclidean



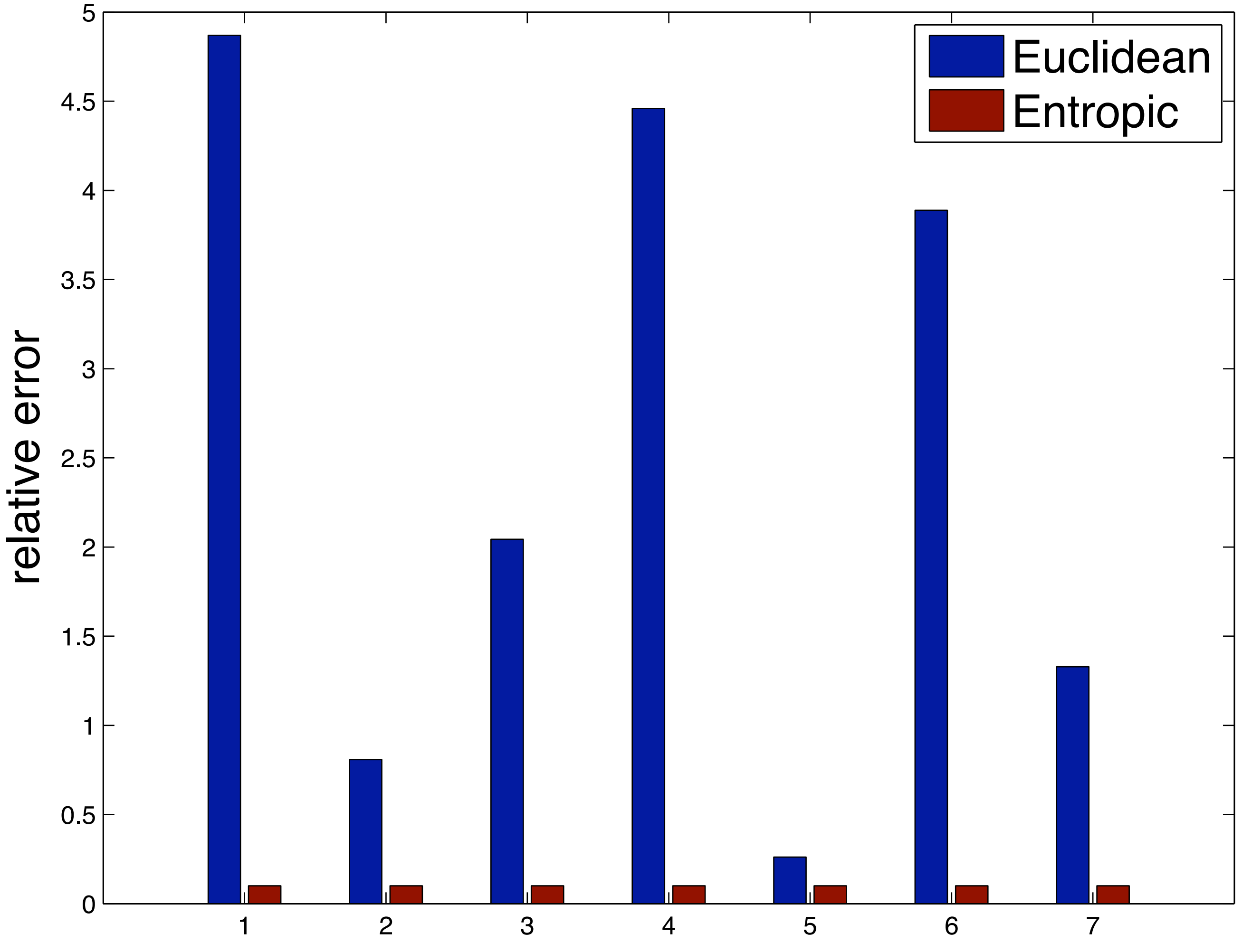
7 different users from the Enron data set

Results: 2 regulairzation

DA1



DA3



Part V:

Further directions

(not covered)

Self-Tuned parameters

- Our algorithmic framework relies on the strong convexity parameter σ
- The optimal choice of σ depends on unknown parameters such as the horizon T and the Lipschitz constants of $\ell_t(\cdot)$
- It is possible to infer these parameters "on-the-fly"

Logarithmic Regret for Strongly Convex

- The dependence of the regret on T in the bounds we derived is $O(\sqrt{T})$
- This dependency tight in a minimax sense
- It is possible to obtain $O(\log(T))$ regret if the loss function is strongly convex
- Main idea: when the loss function is strongly convex additional regularization is not required (the function f can be omitted) and by taking diminishing steps $\sim 1/t$

Online-to-Batch conversions

- Online algorithms can be used in batch settings
- Main idea: if an online algorithm performs well on a sequence of i.i.d. examples then an ensemble of online hypotheses should generalize well
- Thus, we need to construct a single hypothesis from the sequence of online generated hypotheses
- This process is called "Online-to-Batch" conversions
- Popular conversions: pick the averaged hypothesis, the majority vote, use a validation set for choosing a good hypothesis, or simply pick at random a hypothesis from the ensemble

References

- There are numerous relevant papers by:
  - Littlestone, Warmuth, Kivinen, Vovk, Azoury, Freund, Schapire, Gentile, Auer, Grove, Schurmmanns, Long, Smola, Williamson, Herbster, Kalai, Vempala, Hazan ...
- A comprehensive book on online prediction that also covers the connections to game theory and information theory
  - **Prediction Learning and Games.**
    N. Cesa-Bianchi and G. Lugosi. Cambridge university press, 2006.
- The "online convex optimization" model was introduced by Zinkevich
- Use of duality for online learning due to Shalev-Shwartz and Singer
- Most of the topics covered in the tutorial can be found in
  - **Online Learning: Theory, Algorithms, and Applications.**
    S. Shalev-Shwartz. PhD Thesis, The Hebrew University, 2007.
    Advisor: Yoram Singer