Learning at the Turing

The Alan Turing Institute

Mathematics of Machine Learning - Summer School

Lecture 5
Convex Loss Surrogates. Gradient Descent

June 30, 2021

Patrick Rebeschini
Department of Statistics, University of Oxford

Recall Results on Binary Classification

- $Z_i = (X_i, Y_i) \in \mathbb{R}^d \times \{-1, 1\}$
- Admissible action set $\mathcal{A} \subseteq \mathcal{B} := \{a : \mathbb{R}^d \to \{-1, 1\}\}$
- **True** loss function $\ell(a, (x, y)) = 1_{a(x) \neq y} = \varphi^\star(a(x)y)$ with $\varphi^\star(u) := 1_{u \leq 0}$

$$r(a) = \mathbf{P}(a(X) \neq Y) \qquad a^\star \in \operatorname*{argmin}_{a \in \mathcal{A}} r(a) \qquad a^{\star\star} \in \operatorname*{argmin}_{a \in \mathcal{B}} r(a)$$

$$R(a) = \frac{1}{n} \sum_{i=1}^{n} 1_{a(X_i) \neq Y_i} \qquad A^\star \in \operatorname*{argmin}_{a \in \mathcal{A}} R(a)$$

So far we have proved:

$$\boxed{\mathbf{P}\left(r(A^\star) - r(a^\star) \lesssim \sqrt{\frac{\mathtt{VC}(\mathcal{A})}{n}} + \sqrt{\frac{\log(1/\delta)}{n}} \right) \geq 1 - \delta}$$

Problem: In general, computing $A^\star$ is NP hard!

Idea: Define convex relaxation of the original problem

Convexity

Convex function (Definition 8.1)

A function $f : \mathbb{R}^d \to \mathbb{R}$ is *convex* if for every $x, \tilde{x} \in \mathbb{R}^d, \lambda \in [0, 1]$ we have

$$\boxed{f(\lambda x + (1 - \lambda)\tilde{x}) \leq \lambda f(x) + (1 - \lambda) f(\tilde{x})}$$

Convex set (Definition 8.2)

A set $\mathcal{A}$ is *convex* if for every $a, \tilde{a} \in \mathcal{A}, \lambda \in [0, 1]$ we have

$$\boxed{\lambda a + (1 - \lambda)\tilde{a} \in \mathcal{A}}$$

Convex Loss Surrogates

Convex loss surrogate (Definition 8.3)

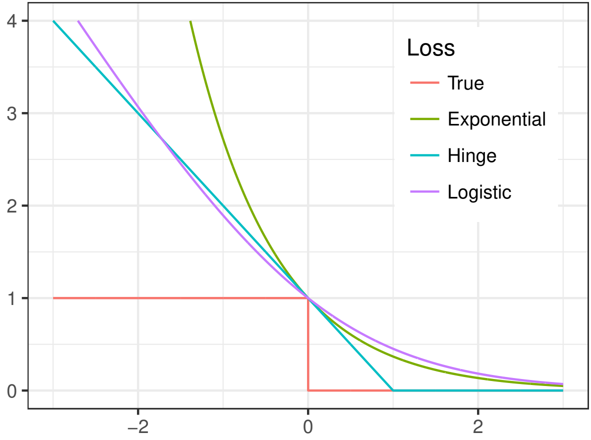
A function $\varphi : \mathbb{R} \to \mathbb{R}_+$ is called a *convex loss surrogate* if:
- convex
- non-increasing
- $\varphi(0) = 1$

True loss:
$\varphi^\star(u) = 1_{u \leq 0}$

Exponential loss:
$\varphi(u) = e^{-u}$

Hinge loss:
$\varphi(u) = \max\{1 - u, 0\}$

Logistic loss:
$\varphi(u) = \log_2(1 + e^{-u})$



Convex Soft Classifiers

- **Soft** classifiers $\mathcal{A}_{\text{soft}} \subseteq \mathcal{B}_{\text{soft}} := \{a : \mathbb{R}^d \to \mathbb{R}\}$
- If $a \in \mathcal{B}_{\text{soft}}$, corresponding **hard** classifier is given by $\text{sign}(a)$

1. **Linear functions with convex parameter space:**

$$\mathcal{A}_{\text{soft}} = \{a(x) = w^\top x + b : w \in \mathcal{C}_1 \subseteq \mathbb{R}^d, b \in \mathcal{C}_2 \subseteq \mathbb{R}\}$$

   $\mathcal{C}_1, \mathcal{C}_2$ are convex sets

2. **Majority votes (Boosting):**

$$\mathcal{A}_{\text{soft}} = \{a(x) = \sum_{i=1}^{m} w_j h_j(x) : w = (w_1, \ldots, w_m) \in \Delta_m\}$$

   Δ_m is the m-dim. simplex and $h_1, \ldots, h_m : \mathbb{R}^d \to \mathbb{R}$ are *base classifiers*

Empirical φ-Risk Minimization

If φ and $\mathcal{A}_{\text{soft}}$ are convex, we are left with a convex problem

$$\boxed{R_\varphi(a) = \frac{1}{n} \sum_{i=1}^{n} \varphi(a(X_i)Y_i)} \qquad \boxed{A^\star_\varphi \in \underset{a \in \mathcal{A}_{\text{soft}}}{\text{argmin}}\, R_\varphi(a)}$$

Zhang's Lemma

$$r_\varphi(a) = \mathbf{E}\,\varphi(a(X)Y) \qquad a_\varphi^{\star\star} \in \operatorname*{argmin}_{a \in \mathcal{B}_{\mathsf{soft}}} r_\varphi(a)$$

$$r(a) = \mathbf{E}\,\varphi^\star(a(X)Y) = \mathbf{P}(a(X) \neq Y) \qquad a^{\star\star} \in \operatorname*{argmin}_{a \in \mathcal{B}} r(a)$$

Zhang's Lemma (Lemma 8.5)

Let $\varphi : \mathbb{R} \to \mathbb{R}_+$ be a convex loss surrogate. For any $\tilde\eta \in [0,1]$, $\tilde a \in \mathbb{R}$, let

$$H_{\tilde\eta}(\tilde a) := \varphi(\tilde a)\tilde\eta + \varphi(-\tilde a)(1-\tilde\eta), \qquad \tau(\tilde\eta) := \inf_{\tilde a \in \mathbb{R}} H_{\tilde\eta}(\tilde\alpha).$$

Assume that there exist $c > 0$ and $\nu \in [0,1]$ such that

$$\boxed{\left|\tilde\eta - \frac{1}{2}\right| \leq c(1-\tau(\tilde\eta))^\nu \qquad \text{for any } \tilde\eta \in [0,1]}$$

Then, for any $a : \mathbb{R}^d \to \mathbb{R}$ we have

$$\boxed{\underbrace{r(\mathrm{sign}(a)) - r(a^{\star\star})}_{\substack{\text{excess risk}\\ \text{hard classifier}}} \leq 2c\big(\underbrace{r_\varphi(a) - r_\varphi(a_\varphi^{\star\star})}_{\substack{\text{excess } \varphi\text{-risk}\\ \text{soft classifier}}}\big)^\nu}$$

Zhang's Lemma: Examples

- **Exponential loss:**
  $\tau(\tilde{\eta}) = 2\sqrt{\tilde{\eta}(1-\tilde{\eta})}$
  $c = 1/\sqrt{2}$
  $\nu = 1/2$

- **Hinge loss:**
  $\tau(\tilde{\eta}) = 1 - |1 - 2\tilde{\eta}|$
  $c = 1/2$
  $\nu = 1$

- **Logistic loss:**
  $\tau(\tilde{\eta}) = -\tilde{\eta}\log_2\tilde{\eta} - (1-\tilde{\eta})\log_2(1-\tilde{\eta})$
  $c = 1/\sqrt{2}$
  $\nu = 1/2$

Zhang's Lemma shows that we can reliably focus on convex problems

Elements of Convex Theory

Subgradients (Definition 8.8)

Let $f : \mathcal{C} \subset \mathbb{R}^d \to \mathbb{R}$. A vector $g \in \mathbb{R}^d$ is a *subgradient* of f at $x \in \mathcal{C}$ if

$$\boxed{f(x) - f(y) \leq g^T(x - y) \qquad \text{for any } y \in \mathcal{C}}$$

The set of subgradients of f at x is denoted $\partial f(x)$.

Subgradients yield **global** information (**uniform** lower bounds)

Convexity and subgradients (Theorem 8.9)

Let $f : \mathcal{C} \subseteq \mathbb{R}^d \to \mathbb{R}$ with $\mathcal{C}$ convex:

f is convex $\implies$ for any $x \in \mathrm{int}(\mathcal{C}), \partial f(x) \neq \emptyset$

f is convex $\impliedby$ for any $x \in \mathcal{C}, \partial f(x) \neq \emptyset$

If f is convex and differentiable at x, then $\nabla f(x) \in \partial f(x)$

Convex functions that are differentiable allow to infer **global** information (i.e., subgradients) from **local** information (i.e., gradients)

This is why convex problems are "typically" amenable to computations...
To prove algorithms converge we need additional local-to-global properties

Are Convex Problems Easy to Solve?

- *Convex hull*: $\mathrm{conv}(\mathcal{T}) := \left\{ \sum_{j=1}^{m} w_j t_j : w \in \Delta_m, t_1, \ldots, t_m \in \mathcal{T}, m \in \mathbb{N} \right\}$
- *Epigraph*: $\mathrm{epi}(f) := \{(x,t) \in \mathcal{D} \times \mathbb{R} : f(x) \le t\}$.

Proposition 8.6

$$\min_{t\in\mathcal{T}} c^\top t = \min_{t\in\mathrm{conv}(\mathcal{T})} c^\top t, \qquad \max_{t\in\mathcal{T}} c^\top t = \max_{t\in\mathrm{conv}(\mathcal{T})} c^\top t.$$

Proof: As $\mathcal{T} \subseteq \mathrm{conv}(\mathcal{T})$, we have $\min_{t\in\mathcal{T}} c^\top t \ge \min_{t\in\mathrm{conv}(\mathcal{T})} c^\top t$. Other direction:

$$\begin{aligned}
\min_{t\in\mathrm{conv}(\mathcal{T})} c^\top t &= \min_{m\in\mathbb{N}} \min_{t_1,\ldots,t_m\in\mathcal{T}} \min_{(w_1,\ldots,w_m)\in\Delta_m} c^\top \left(\sum_{j=1}^{m} w_j t_j \right) \\
&= \min_{m\in\mathbb{N}} \min_{t_1,\ldots,t_m\in\mathcal{T}} \min_{(w_1,\ldots,w_m)\in\Delta_m} \sum_{j=1}^{m} w_j c^\top t_j \ge \min_{t\in\mathcal{T}} c^\top t.
\end{aligned}$$

Proposition 8.7

For any $f : \mathcal{D} \subseteq \mathbb{R}^d \to \mathbb{R}$, $\min_{x\in\mathcal{D}} f(x) = \min_{(x,t)\in\mathcal{C}} t$ with $\mathcal{C} = \mathrm{conv}(\mathrm{epi}(f))$.

Any minimization problem can be written in a convex form!

Local-to-Global Properties

- **Convex:** $\boxed{f(y) \geq f(x) + \nabla f(x)^T (y-x) \quad \forall x, y \in \mathbb{R}^d}$
- **α-Strongly Convex:**

$$\boxed{\exists \alpha > 0 \text{ such that } f(y) \geq f(x) + \nabla f(x)^T (y-x) + \frac{\alpha}{2}\|y-x\|_2^2 \quad \forall x, y \in \mathbb{R}^d}$$

- **β-Smooth:**

$$\boxed{\exists \beta > 0 \text{ such that } f(y) \leq f(x) + \nabla f(x)^T (y-x) + \frac{\beta}{2}\|y-x\|_2^2 \quad \forall x, y \in \mathbb{R}^d}$$

- **γ-Lipschitz:**

$$\boxed{\exists \gamma > 0 \text{ such that } f(x) - \gamma\|y-x\|_2 \leq f(y) \leq f(x) + \gamma\|y-x\|_2 \ \forall x, y \in \mathbb{R}^d}$$

	Strongly convex?	**Smooth?**	**Lipschitz?**
Exponential loss (in $\mathbb{R}$)	NO	NO	NO
Hinge loss (in $\mathbb{R}$)	NO	NO	YES
Logistic loss (in $\mathbb{R}$)	NO	YES	YES

However, we typically only need the domain to be a compact set of $\mathbb{R}$

Recap

- Training data: $(X_1, Y_1), \dots, (X_n, Y_n) \in \mathcal{X} \times \{-1, 1\}$, with $\mathcal{X} \subseteq \mathbb{R}^d$
- Loss function: $\varphi : \mathbb{R} \to \mathbb{R}_+$ (**convex**: reasonable by Zhang's lemma)
- Predictors $\mathcal{A} = \{x \in \mathbb{R}^d \to a_w(x) : w \in \mathcal{W}\}$ ($\mathcal{W}$ **convex** in many cases)
  NB. There are many settings where $\mathcal{A}$ is **not** convex (e.g., neural networks)

Risk minimization:

$$\begin{aligned} &\underset{w}{\text{minimize}} && r(w) = \mathbf{E}\varphi(a_w(X)Y) \\ &\text{subject to} && w \in \mathcal{W} \end{aligned} \quad \Longrightarrow \quad \text{Let } w^\star \text{ be a minimizer}$$

Empirical risk minimization:

$$\begin{aligned} &\underset{w}{\text{minimize}} && R(w) = \frac{1}{n}\sum_{i=1}^{n} \varphi(a_w(X_i)Y_i) \\ &\text{subject to} && w \in \mathcal{W} \end{aligned} \quad \Longrightarrow \quad \text{Let } W^\star \text{ be a minimizer}$$

$$\boxed{r(W) - r(w^\star) \le \underbrace{R(W) - R(W^\star)}_{\texttt{Optimization}} + \underbrace{\sup_{w \in \mathcal{W}} \{r(w) - R(w)\} + \sup_{w \in \mathcal{W}} \{R(w) - r(w)\}}_{\texttt{Statistics}}}$$

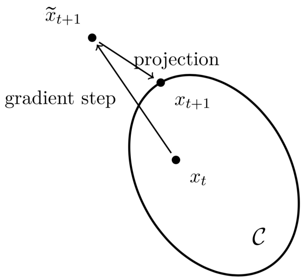
Projected Subgradient Method

Goal: $\boxed{\min_{x \in \mathcal{C}} f(x)}$ with f convex, $\mathcal{C}$ convex and compact

Projected Subgradient Method

$$\boxed{\begin{aligned} &\tilde{x}_{t+1} = x_t - \eta_t g_t, \text{where } g_t \in \partial f(x_t) \\ &x_{t+1} = \Pi_{\mathcal{C}}(\tilde{x}_{t+1}) \end{aligned}}$$

with the projection operator $\Pi_{\mathcal{C}}(y) = \operatorname{argmin}_{x \in \mathcal{C}} \|x - y\|_2$.



Non-Expansivity of Projections

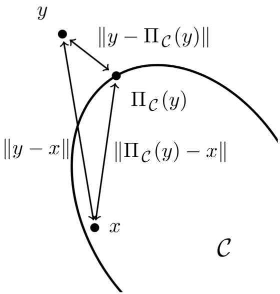
Non-expansivity (Proposition 9.2)

Let $x \in \mathcal{C}$ and $y \in \mathbb{R}^d$. Then,

$$\left(\Pi_{\mathcal{C}}(y) - x\right)^\top \left(\Pi_{\mathcal{C}}(y) - y\right) \leq 0$$

which implies $\|\Pi_{\mathcal{C}}(y) - x\|_2^2 + \|y - \Pi_{\mathcal{C}}(y)\|_2^2 \leq \|y - x\|_2^2$ and, in particular,

$$\boxed{\|\Pi_{\mathcal{C}}(y) - x\|_2 \leq \|y - x\|_2}$$



First Order Optimality Condition

First Order Optimality Condition (Proposition 8.10)

Let f be convex, and $\mathcal{C}$ be a closed set on which f is differentiable. Then,

$$x^\star \in \operatorname*{argmin}_{x \in \mathcal{C}} f(x) \quad \Longleftrightarrow \quad \nabla f(x^\star)^\top (x^\star - x) \leq 0 \quad \text{for any } x \in \mathcal{C}$$

Proof of Proposition 9.2. This is a direct consequence of Proposition 8.10 since $\Pi_{\mathcal{C}}(y)$ is a minimizer of the function $z \to f_y(z) = \|y - z\|_2$, and $\nabla f_y(z) = (z - y)/\|z - y\|_2$.

Results for Lipschitz Functions

A function f is γ-**Lipschitz on** $\mathcal{C}$ if there exists $\gamma > 0$ such that (equivalent)

- For every $x, y \in \mathcal{C}$, $f(x) - \gamma\|x - y\|_2 \leq f(y) \leq f(x) + \gamma\|x - y\|_2$
- For every $x, y \in \mathcal{C}$, $|f(y) - f(x)| \leq \gamma\|x - y\|_2$
- For every $x \in \mathcal{C}$, any subgradient $g \in \partial f(x)$ satisfies $\|g\|_2 \leq \gamma$

Projected Subgradient Method—Lipschitz (Theorem 9.3)

- Function f is γ-Lipschitz
- Assume $\|x_1 - x^\star\|_2 \leq b$

Then, the projected subgradient method with $\eta_s \equiv \eta = \frac{b}{\gamma\sqrt{t}}$ satisfies

$$\boxed{f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^\star) \leq \frac{\gamma b}{\sqrt{t}}}$$

It is not a descent method: the value function can increase in one time step

Proof of Theorem 9.3)

- Convexity yields:

$$f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^\star) \le \frac{1}{t}\sum_{s=1}^{t} f(x_s) - f(x^\star) \le \frac{1}{t}\sum_{s=1}^{t} g_s^\top (x_s - x^\star)$$

- Using $2a^\top b = \|a\|_2^2 + \|b\|_2^2 - \|a-b\|_2^2$ and $g_s = \frac{1}{\eta}(x_s - \tilde{x}_{s+1})$:

$$\begin{aligned} g_s^\top (x_s - x^\star) &= \frac{1}{\eta}(x_s - \tilde{x}_{s+1})^\top (x_s - x^\star) \\ &= \frac{1}{2\eta}\left(\|x_s - x^\star\|_2^2 + \|x_s - \tilde{x}_{s+1}\|_2^2 - \|\tilde{x}_{s+1} - x^\star\|_2^2\right) \\ &= \frac{1}{2\eta}\left(\|x_s - x^\star\|_2^2 - \|\tilde{x}_{s+1} - x^\star\|_2^2\right) + \frac{\eta}{2}\|g_s\|_2^2 \\ &\le \frac{1}{2\eta}\left(\|x_s - x^\star\|_2^2 - \|x_{s+1} - x^\star\|_2^2\right) + \frac{\eta}{2}\|g_s\|_2^2 \end{aligned}$$

  where we used that $\|\tilde{x}_{s+1} - x^\star\|_2 \ge \|x_{s+1} - x^\star\|_2$ by Proposition 9.2.

- Summing from $s = 1$ to t:

$$f\left(\frac{1}{t}\sum_{s=1}^{t} x_s\right) - f(x^\star) \le \frac{1}{2\eta t}\left(\|x_1 - x^\star\|_2^2 - \|x_{t+1} - x^\star\|_2^2\right) + \frac{\eta\gamma^2}{2} \le \frac{b^2}{2\eta t} + \frac{\eta\gamma^2}{2}$$

  Minimizing the right-hand side we have $\eta = \frac{b}{\gamma\sqrt{t}}$ which yields the result.

Results for Smooth Functions

A function f is β-**smooth on** $\mathcal{C}$ if there exists $\beta > 0$ such that (equivalent)

- For every $x, y \in \mathcal{C}$, $f(y) \leq f(x) + \nabla f(x)^\top (y - x) + \frac{\beta}{2}\|y - x\|_2^2$
- For every $x, y \in \mathcal{C}$, $|\nabla f(y) - \nabla f(x)| \leq \beta \|x - y\|_2$ (gradient is β-Lipschitz)
- For every $x \in \mathcal{C}$, $\nabla^2 f(x) \preccurlyeq \beta I$ (if f is twice-differentiable)

Projected Gradient Descent—Smooth (Theorem 9.4)

- Function f is β-smooth
- Assume $\|x_1 - x^\star\|_2 \leq b$

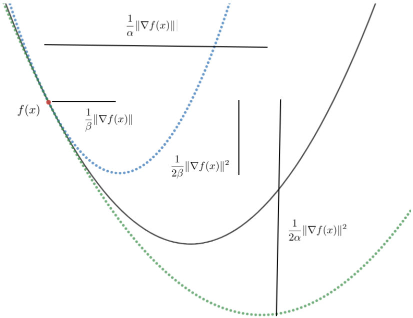
Then, projected gradient descent with $\eta_s \equiv \eta = 1/\beta$ satisfies

$$\boxed{f(x_t) - f(x^\star) \leq \frac{3\beta b^2 + f(x_1) - f(x^\star)}{t}}$$

In the case of smooth functions, gradient descent is a natural algorithm...

Interpretation for Smooth Functions

... it is the algorithm that at each time step moves to the point in $\mathcal{C}$ that maximizes the guaranteed local decrease given by the quadratic function that uniformly upper-bounds the function f at the current location



$$\operatorname*{argmin}_{y\in\mathcal{C}}\left\{f(x)+\nabla f(x)^\top(y-x)+\frac{\beta}{2}\|y-x\|_2^2\right\}=\operatorname*{argmin}_{y\in\mathcal{C}}\left\{\left\|\left(x-\frac{1}{\beta}\nabla f(x)\right)-y\right\|_2^2\right\}$$
$$\equiv\Pi_{\mathcal{C}}\left(x-\frac{1}{\beta}\nabla f(x)\right)$$

Results for Smooth and Strongly Convex Functions

A function f is **α-strongly convex on $\mathcal{C}$** if there is $\alpha > 0$ such that (equivalent)

- For every $x, y \in \mathcal{C}$, $f(y) \geq f(x) + \nabla f(x)^\top (y - x) + \frac{\alpha}{2} \|y - x\|_2^2$
- For every $x \in \mathcal{C}$, $\nabla^2 f(x) \succcurlyeq \alpha I$ (if f is twice-differentiable)

Gradient Descent—Smooth and Strongly Convex (Theorem 9.5)

- Assume $\mathcal{C} = \mathbb{R}^d$ (same type of result holds for projected gradient descent)
- Function f is α-strongly convex and β-smooth

Then, gradient descent with $\eta_s \equiv \eta = 1/\beta$ satisfies

$$\boxed{f(x_t) - f(x^\star) \leq \left(1 - \frac{\alpha}{\beta}\right)^{t-1} (f(x_1) - f(x^\star))}$$

Proof: (see illustration on the previous slide)

- Guaranteed progress in one step: $f(x_{s+1}) \leq f(x_s) - \frac{1}{2\beta} \|\nabla f(x_s)\|_2^2$
- Lower bound on objective function: $f(x^\star) \geq f(x_s) - \frac{1}{2\alpha} \|\nabla f(x_s)\|_2^2$

Oracle Complexity, Lower Bounds, Accelerated Methods

- **Convergence rates:**

	L-Lipschitz	β-smooth
Convex	$O(\gamma b/\sqrt{t})$	$O((\beta b^2 + c)/t)$
α-strongly convex	$O(\gamma^2/(\alpha t))$	$O(e^{-t\alpha/\beta} c)$

where $\|x_1 - x^\star\|_2 \leq b$ and $f(x_1) - f(x^\star) \leq c$

- **Oracle complexities:**

	L-Lipschitz	β-smooth
Convex	$O(\gamma^2 b^2/\varepsilon^2)$	$O((\beta b^2 + c)/\varepsilon)$
α-strongly convex	$O(\gamma^2/(\alpha\varepsilon))$	$O((\beta/\alpha)\log(c/\varepsilon))$

- **Optimal rates (lower bounds)**

	L-Lipschitz	β-smooth
Convex	$\Omega(\gamma a/(1+\sqrt{t}))$	$\Omega(\tilde{b}^2\beta/(t+1)^2)$
α-strongly convex	$\Omega(\gamma^2/(\alpha t))$	$\Omega(\alpha\tilde{b}^2 e^{-t\sqrt{\alpha/\beta}})$

where $a := \max_{x\in\mathcal{C}} \|x\|_2$ and $\tilde{b} := \max_{x,y\in\mathcal{C}} \|x - y\|_2$

Apart from Lipschitz, optimal rates are achieved only by **accelerated** algorithms

NB. Quantities α, β, γ and $a, b, c, \tilde{b}$ depend implicitly on dimension d

Back to Learning: Linear Predictors with ℓ_2 Ball

Risk minimization:

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & r(w) = \mathbf{E}\varphi(w^\top XY) \\ \text{subject to} \quad & \|w\|_2 \le c_2^{\mathcal{W}} \end{aligned} \quad \Longrightarrow \quad \text{Let } w^\star \text{ be a minimizer}$$

Empirical risk minimization:

$$\begin{aligned} \underset{w}{\text{minimize}} \quad & R(w) = \frac{1}{n}\sum_{i=1}^{n} \varphi(w^\top X_i Y_i) \\ \text{subject to} \quad & \|w\|_2 \le c_2^{\mathcal{W}} \end{aligned} \quad \Longrightarrow \quad \text{Let } W^\star \text{ be a minimizer}$$

$$\boxed{r(\overline{W}_t) - r(w^\star) \le \underbrace{R(\overline{W}_t) - R(W^\star)}_{\texttt{Optimization}} + \underbrace{\sup_{w \in \mathcal{W}} \{r(w) - R(w)\} + \sup_{w \in \mathcal{W}} \{R(w) - r(w)\}}_{\texttt{Statistics}}}$$

$$\boxed{\mathbf{E}\,\texttt{Statistics} \le \frac{4 c_2^{\mathcal{X}} c_2^{\mathcal{W}} \gamma_\varphi}{\sqrt{n}}} \qquad \boxed{\texttt{Optimization} \le \frac{2 c_2^{\mathcal{X}} c_2^{\mathcal{W}} \gamma_\varphi}{\sqrt{t}}}$$

Principled approach: Enough to run algorithm for $t \sim n$ time steps
(ONLY BASED ON UPPER BOUNDS!)