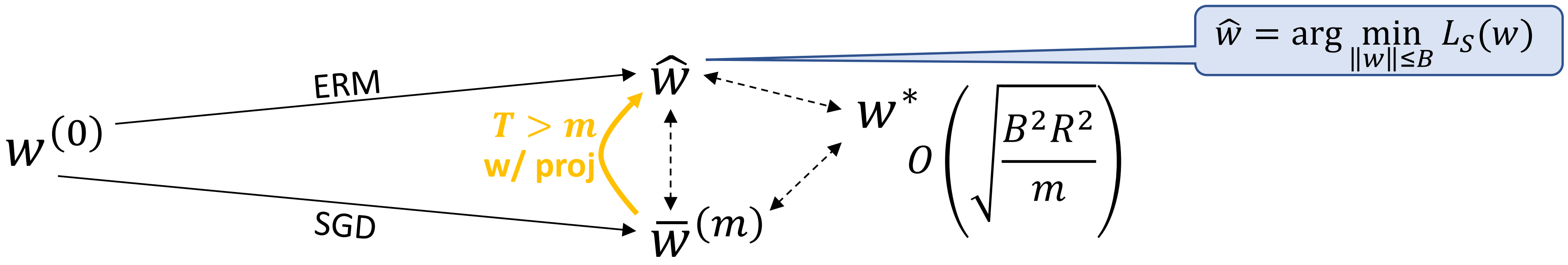
Introduction to Machine Learning
TTIC 31020

Prof. Nati Srebro

Lecture 11:

Multi-Pass SGD
Online Learning
The Inductive Bias of Optimization



Direct SA (SGD) Approach:

$$\min L_S(w)$$

Initialize $w^{(0)} = 0$

At iteration t:

- Draw $x_t, y_t \sim \mathcal{D}$
- If $y_t\langle w^{(t)}, \phi(x_t)\rangle < 1$,
  $w^{(t+1)} \leftarrow w^{(t)} + \eta_t y_t \phi(x_t)$
  else: $w^{(t+1)} \leftarrow w^{(t)}$

Return $\bar{w}^{(T)} = \frac{1}{T}\sum_{t=1}^{T} w^{(t)}$

- **Fresh sample at each iteration**, $m = T$
  ➔ one pass over the data
- No need to project nor require $\|w\| \le B$
- Implicit regularization via early stopping

SGD on ERM:

$$\min_{\|w\|_2 \le B} L_S(w)$$

Draw $(x_1, y_1), \ldots, (x_m, y_m) \sim \mathcal{D}$

Initialize $w^{(0)} = 0$

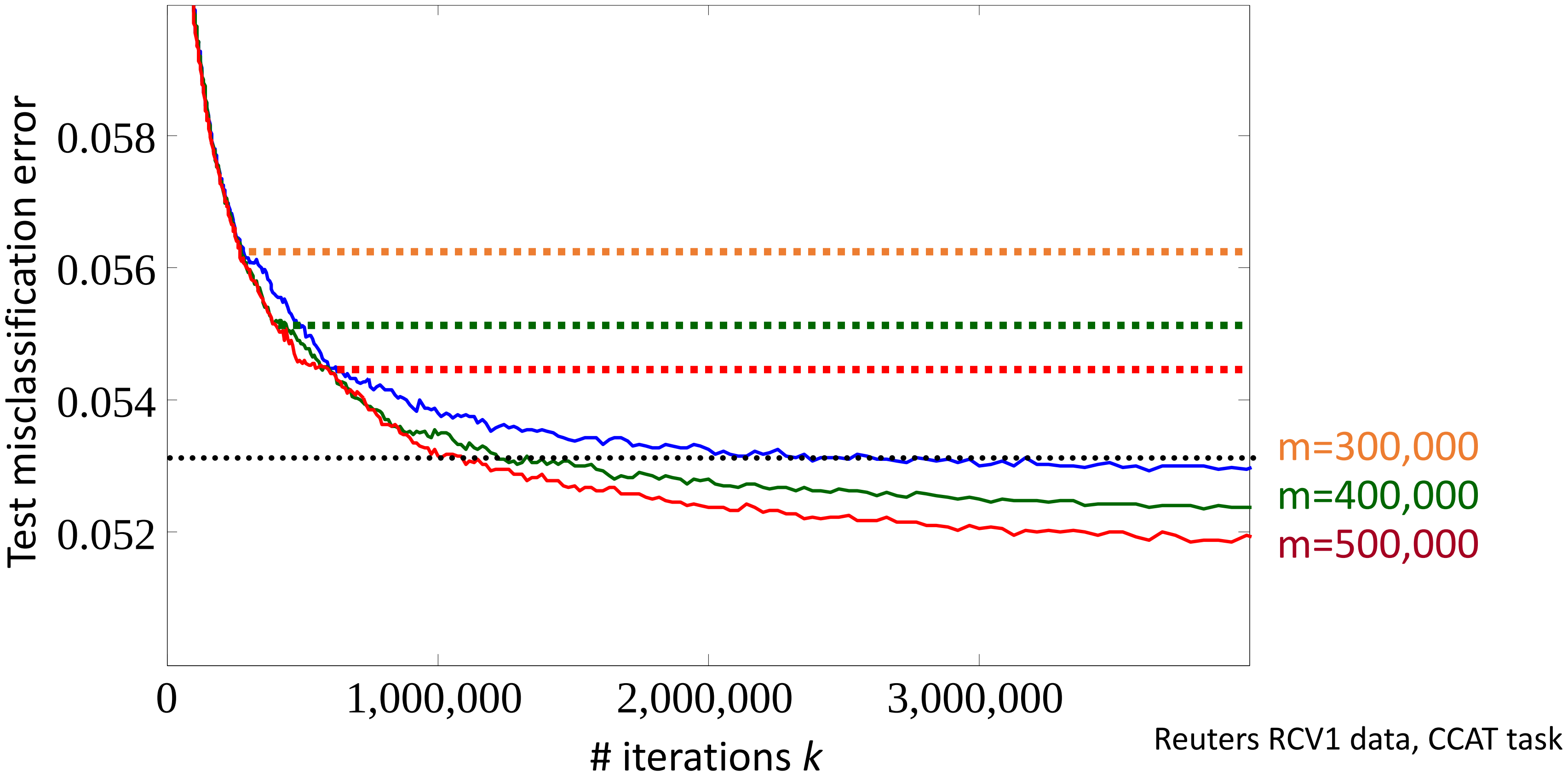
At iteration t:

- Pick $i \in 1 \ldots m$ at random
- If $y_i\langle w^{(t)}, \phi(x_i)\rangle < 1$,
  $w^{(t+1)} \leftarrow w^{(t)} + \eta_t y_i \phi(x_i)$
  else: $w^{(t+1)} \leftarrow w^{(t)}$
- $w^{(t+1)} \leftarrow proj\ w^{(t+1)}\ to\ \|w\| \le B$

Return $\bar{w}^{(T)} = \frac{1}{T}\sum_{t=1}^{T} w^{(t)}$

- Can have $T > m$ iterations (multiple passes)
- Need to project to $\|w\| \le B$
- Explicit regularization via $\|w\|$

Mixed Approach: SGD on ERM



- The mixed approach (reusing examples) can make sense
- Still: fresh samples are better

  With a larger training set, can reduce generalization error faster

  Larger training set means *less* runtime to get target generalization error

Direct SA/SGD Approach

(Learning ***as*** Stochastic Optimization)

SGD on the objective $L_{\mathcal{D}}(w)$

SGD is the Learning Rule

One pass/“online”, $T = m$
(processes each example once, one “epoch” over the data)

SGD on ERM

(Learning ***using*** Stochastic Optimization)

SGD on $L_S(w)$

Learning rule: $\mathrm{ERM}(S) = \hat{w}_B = \arg\min_{\|w\|_2} L_S(w)$

or $RERM(S) = \hat{w}_\lambda = \arg\min L_S(w) + \lambda\|w\|_2^2$

SGD as an Optimization Algorithm for min

Multiple passes/epochs, can have $T > m$
(can processes examples multiple times)

Online learning:

At each iteration $t = 1,2,\ldots$

- Receive instance x_t
- Predict a label $\hat{y}_t = h^{(t)}(x_t)$
- Receive label y_t,
- Update $h^{(t+1)}$ based on (x_t, y_t)

Stochastic Approximation (e.g. SGD):

At each iteration $t = 1,2,\ldots$

receive (x_t, y_t)

update $h^{(t+1)}$ based on (x_t, y_t)

- Goal in realizable case ($\exists_{h^* \in \mathcal{H}} h^*(x_t) = y_t$): #mistakes (ie $h^{(t)}(x_t) \neq y_t$)

$$\frac{1}{m}\sum_t \ell^{01}\left(h^{(t)}(x_t), y_t\right) \le \underbrace{\qquad\qquad\qquad\qquad}_{0} \quad \epsilon$$

#mistakes$/m$

- Goal in agnostic case: regret versus best $h^* \in \mathcal{H}$ in hindsight

$$\frac{1}{m}\sum_t \ell\left(h^{(t)}(x_t), y_t\right) \le \inf_{h^* \in \mathcal{H}} \frac{1}{m}\sum_t \ell(h^*(x_t), y_t) \;+\; \epsilon$$

regret

Online regret guarantees beyond scope of course

Online Gradient Descent

Online learning:

At each iteration $t = 1,2, \ldots$

- Receive instance x_t
- Predict a label $\hat{y}_t = h_{w^{(t)}}(x_t)$
- Receive label y_t, suffer loss $\ell\left(h_{w^{(t)}}, y_t\right)$
- Update $w^{(t+1)}$ based on (x_t, y_t)

$$\begin{aligned} \boldsymbol{w}^{(t+1)} &\leftarrow \boldsymbol{w}^{(t)} - \boldsymbol{\eta_t \nabla_w \ell(h_{w^{(t)}}(x_t), y_t)} \\ &= w^{(t)} - \eta_t \nabla_w \ell(\langle \boldsymbol{w}^{(t)}, \boldsymbol{\phi}(\boldsymbol{x}_t)\rangle, y_t) \\ &= w^{(t)} - \eta_t \ell'\left(\langle \boldsymbol{w}^{(t)}, \boldsymbol{\phi}(\boldsymbol{x}_t)\rangle, \boldsymbol{y}_t\right)\boldsymbol{\phi}(\boldsymbol{x}_t) \end{aligned}$$

For linear pred
$h_w(x) = \langle w, \phi(x)\rangle$

- If $\ell(h_w(x), y)$ is convex and ρ-Lipschitz in w

$$\frac{1}{m}\sum_t \ell\left(h_{w^{(t)}}, y_t\right) \le \inf_{\|w\|_2 \le \boldsymbol{B}} \frac{1}{m}\sum_t \ell(h_w(x_t), y_t) + \sqrt{\frac{B^2\rho^2}{m}}$$

- If $h_w(x) = \langle w, \phi(x)\rangle$, $\|\phi(x)\|_2 \le R$ and $\ell(z, y)$ is 1-Lipschitz in z:

$$\frac{1}{m}\sum_t \ell\left(\langle w^{(t)}, \phi(x_t)\rangle, y_t\right) \le \inf_{\|w\|_2 \le \boldsymbol{B}} \frac{1}{m}\sum_t \ell(\langle w, \phi(x_t)\rangle, y_t) + \sqrt{\frac{B^2R^2}{m}}$$

Online regret guarantees beyond scope of course

Perceptron as OGD

Online learning:

At each iteration $t = 1,2,\ldots$

- Receive instance x_t
- Predict a label $\hat{y}_t = h_{w^{(t)}}(x_t)$
- Receive label y_t, suffer loss $\ell\left(h_{w^{(t)}}, y_t\right)$
- Update $w^{(t+1)}$ based on (x_t, y_t)

$$\begin{aligned} \boldsymbol{w}^{(t+1)} &\leftarrow \boldsymbol{w}^{(t)} - \boldsymbol{\eta_t \nabla_w} \ell\left(\boldsymbol{h}_{\boldsymbol{w}^{(t)}}(\boldsymbol{x}_t), \boldsymbol{y}_t\right) \\ &= w^{(t)} - \eta_t \nabla_w \ell\left(\langle \boldsymbol{w}^{(t)}, \boldsymbol{\phi}(\boldsymbol{x}_t)\rangle, y_t\right) \\ &= w^{(t)} - \eta_t \ell'\left(\langle \boldsymbol{w}^{(t)}, \boldsymbol{\phi}(\boldsymbol{x}_t)\rangle, \boldsymbol{y}_t\right) \boldsymbol{\phi}(\boldsymbol{x}_t) \end{aligned}$$

???

At iteration t:

- Receive x_t
- Predict $\hat{y}_t = sign\left(\langle w^{(t)}, \phi(x_t)\rangle\right)$
- Receive y_t
- **If** $\boldsymbol{y_t \neq \hat{y}_t}$,
  $\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} + \boldsymbol{y_t \phi(x_t)}$
  else: $\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)}$

$$\ell'\left(\langle w^{(t)}, \phi(x_t)\rangle, y_t\right) = \begin{cases} -1, & y_t \neq \hat{y}_t = sign(\langle w^{(t)}, \phi(x_t)\rangle) \\ 0, & y_t = sign(\langle w^{(t)}, \phi(x_t)\rangle) \end{cases}$$

Perceptron as OGD

Online learning:

At each iteration $t = 1,2,\ldots$

- Receive instance x_t
- Predict a label $\hat{y}_t = h_{w^{(t)}}(x_t)$
- Receive label y_t, suffer loss $\ell\left(h_{w^{(t)}}, y_t\right)$
- Update $w^{(t+1)}$ based on (x_t, y_t)

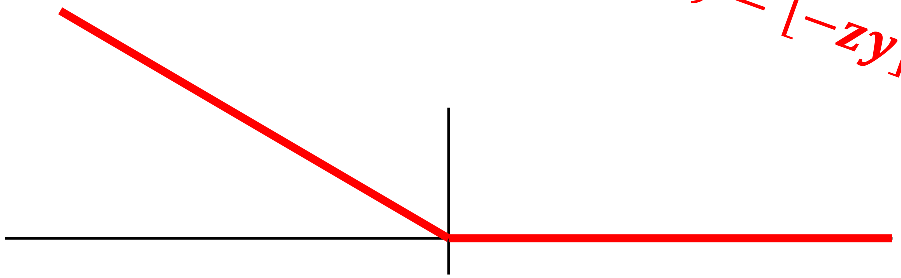
$$\begin{aligned} \boldsymbol{w}^{(t+1)} &\leftarrow \boldsymbol{w}^{(t)} - \boldsymbol{\eta}_t \boldsymbol{\nabla}_w \ell\left(\boldsymbol{h}_{w^{(t)}}(\boldsymbol{x}_t), \boldsymbol{y}_t\right) \\ &= w^{(t)} - \eta_t \nabla_w \ell\left(\langle \boldsymbol{w}^{(t)}, \boldsymbol{\phi}(\boldsymbol{x}_t)\rangle, y_t\right) \\ &= w^{(t)} - \eta_t \ell'\left(\langle \boldsymbol{w}^{(t)}, \boldsymbol{\phi}(\boldsymbol{x}_t)\rangle, \boldsymbol{y}_t\right)\boldsymbol{\phi}(\boldsymbol{x}_t) \end{aligned}$$

At iteration t:

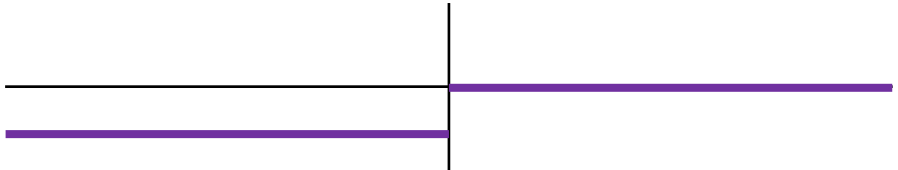
- Receive x_t
- Predict $\hat{y}_t = sign\left(\langle w^{(t)}, \phi(x_t)\rangle\right)$
- Receive y_t
- If $\boldsymbol{y}_t \neq \hat{\boldsymbol{y}}_t$, $\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)} + \boldsymbol{y}_t \boldsymbol{\phi}(\boldsymbol{x}_t)$

  else: $\boldsymbol{w}^{(t+1)} \leftarrow \boldsymbol{w}^{(t)}$

$\ell(\boldsymbol{z}, \boldsymbol{y}) = [-\boldsymbol{z}\boldsymbol{y}]_+$



$$\ell'(z, y) = \begin{cases} -1, & y \neq sign(z) \text{ i.e., } yz < 0 \\ 0, & y = sign(z) \text{ i.e., } yz > 0 \end{cases}$$



Frank Rosenblatt

$$\frac{1}{m}\sum_t \ell(h_{w^{(t)}}, y_t) \le \inf_w \frac{1}{m}\sum_t \ell(h_w(x_t), y_t) + Regret$$

Online algorithm A

e.g. Online Gradient Descent:

$$w^{(t+1)} \leftarrow w^{(t)} - \eta \nabla_w \left(h_{w^{(t)}}(x_t, y_t)\right)$$

or online Perceptron

Realizable Online-to-Batch

(if $\exists w^* \; L_S(w^*) = 0$)

Input: $S = (x_1, y_1)..(x_m, y_m) \sim \mathcal{D}^m$

While $y_i w^{(t)}(x_i) < 0$,

feed (x_i, y_i) into A to get $w^{(t+1)}$

Output $w^{(T)}$

Empirical Optimization: $L_S\left(w^{(T)}\right) = 0$

Generalization: $\mathbb{E}\left[L_{\mathcal{D}}\left(w^{(T)}\right)\right] \le \frac{\#\text{mistakes}}{m} = Regret$

One-Pass Online-to-Batch

Input: $S = (x_1, y_1)..(x_m, y_m) \sim \mathcal{D}^m$

For $t = 1 \ldots \boldsymbol{m}$,

feed (x_t, y_t) into A to get $w^{(t+1)}$

Output $\bar{w} = \frac{1}{m}\sum w^{(t)}$

Generalization:

$$\mathbb{E}[L_{\mathcal{D}}(\bar{w})] \le \inf_{w^*} L_{\mathcal{D}}(w^*) + Regret$$

Onlined Gradient Descent [Zinkevich 03] $\xrightarrow[\text{[Cesa-Binachi et al 04]}]{\text{online2stochastic}}$ Stochastic Gradient Descent [Nemirovski Yudin 78]

Online Learning vs Stochastic Approximation

- In both Online Setting and Stochastic Approximation
  - Receive samples sequentially
  - Update predictor after each sample

- But, in Online Setting:
  - Objective is empirical regret, i.e. behavior on observed instances
  - Every point is both a training point and a test point
  - (x_t, y_t) chosen arbitrarily (no distribution involved), could be non stationary, non independent, adapt based on predictor, anything goes
- Whereas in Stochastic Approximation:
  - Objective is $L(h) = \mathbb{E}_{x,y}[loss(h(x), y)]$, i.e. behavior on "future" samples $(x, y) \sim \mathcal{D}$
  - i.i.d. *training* samples $(x_t, y_t) \sim \mathcal{D}$
  - Have same source distribution $\mathcal{D}$ for train and test crucial

- Stochastic Approximation is a computational approach, Online Learning is an analysis setup
  - E.g. "Majority" is a valid online algorithm and makes sense to analyze as such

Direct SA/SGD Approach

(Learning ***as*** Stochastic Optimization)

SGD on the objective $L_{\mathcal{D}}(w)$

SGD as a Learning Rule

One pass/epoch: "online", $T = m$
(processes each example once)

Generalization from SGD regret guarantee

$$L_{\mathcal{D}}\left(\overline{w}^T\right) \le L_{\mathcal{D}}(w^*) + O\left(\sqrt{\frac{\|w^*\|_2^2 \|\phi\|_2^2}{T}}\right)$$

What is the inductive bias?

How and where is it specified or used in SGD?

SGD on ERM

(Learning ***using*** Stochastic Optimization)

SGD on $L_S(w)$

Learning rule: $\mathrm{ERM}(S) = \hat{w}_B = \arg\min_{\|w\|_2} L_S(w)$

or $RERM(S) = \hat{w}_\lambda = \arg\min L_S(w) + \lambda \|w\|_2^2$

SGD as an Optimization Algorithm for min

Multiple passes/epochs, can have $T > m$
(can processes examples multiple times)

Explicit complexity control: $\|w\|_2 \le B$ or $+\lambda\|w\|_2^2$

Generalization from explicit complexity control:

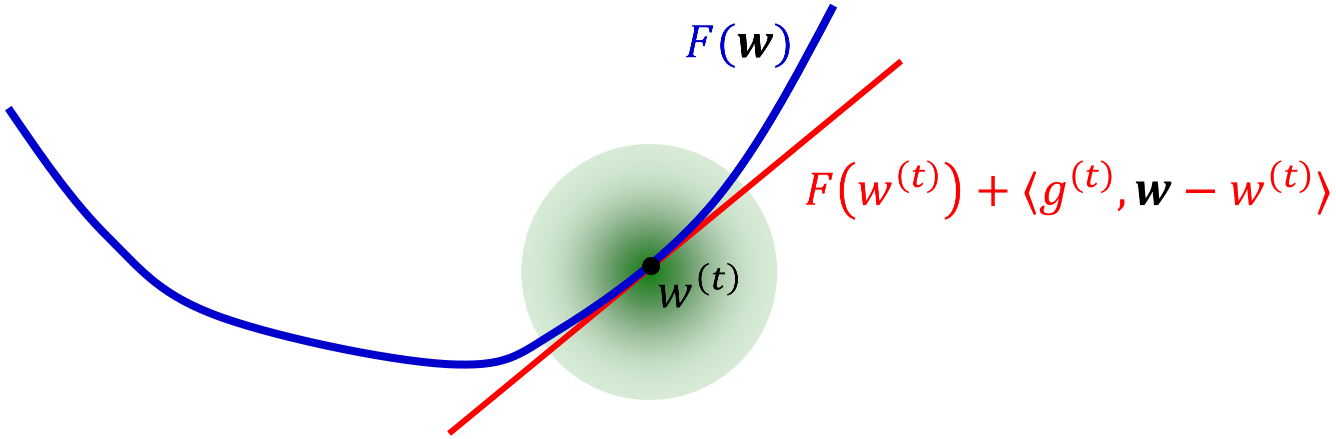
$$L_{\mathcal{D}}(\hat{w}_B) \le L_{\mathcal{D}}(w^*) + O\left(\sqrt{\frac{\|w^*\|_2^2 \|\phi\|_2^2}{m}}\right)$$

Explicit inductive bias: $\|w\|_2$

Where's the Regularization

- Gradient Descent seems to be regularizing with $\|w\|_2$. How?

$$w^{(t+1)} \leftarrow \arg\min_{w} \underbrace{F\left(w^{(t)}\right) + \langle g^{(t)}, w - w^{(t)} \rangle}_{\substack{\text{1st order model of } F(\mathbf{w}) \\ \text{around } \mathbf{w}^{(t)}\text{, based on } \mathbf{g}^{(t)}}} + \underbrace{\frac{1}{2\eta}\left\|w - w^{(t)}\right\|_2}_{\substack{\text{only valid near } w^{(t)}, \\ \text{so don't go too far.} \\ \text{And stochastic,} \\ \text{so don't trust it too much}}}$$



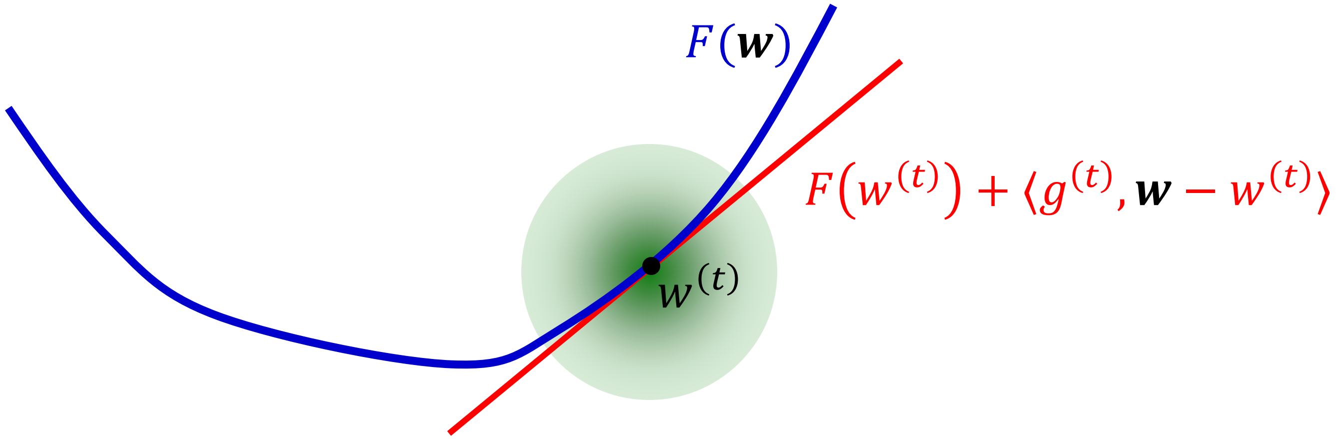
Where's the Regularization

- Gradient Descent seems to be regularizing with $\|w\|_2$. How?

$$w^{(t+1)} \leftarrow \arg\min_{w} F\left(w^{(t)}\right) + \langle g^{(t)}, w - w^{(t)} \rangle \ + \frac{1}{2\eta}\left\|w - w^{(t)}\right\|_2$$

$$= \ \arg\min_{w} \langle g^{(t)}, w \rangle + \frac{1}{2\eta}\left\|w - w^{(t)}\right\|_2$$

$$= w^{(t)} - \eta g^{(t)}$$



- SGD (at least on convex problems) implicitly regularizes using $\|w\|_2$
  - #iterations $T \approx$ sample complexity $m \propto \|w\|_2^2$
  - Generalization/suboptimality controlled in terms of $\|w\|_2$ ➔ this is the inductive bias
  - Alternative to $\|w\|_2 \leq B$ or adding $\lambda\|w\|_2$ for injecting $\|w\|_2$ inductive bias (same guarantee)

- What about other regularizers $R(w)$ / inductive biases??
  - Can apply SGD to regularized or constrained ERM:

$$\min_{R(w)\leq B} L_S(w) \quad \text{or} \quad \min L_S(w) + \lambda R(w)$$

Sample complexity m controlled by $R(w^*)$,

…but #iterations T controlled by $\|w^*\|_2$

  - Other optimization methods related to other regularizers / inductive biases

(generic answer for convex $R(w)$ and convex (ie linear) learning problems: Stochastic Mirror Descent with potential function corresponding to $R(w)$—beyond scope of this course)

- Stochastic Gradient Descent as a Learning Algorithm:
  - One pass over the data!

- What if we do multiple passes over the data?
- Or what about batch gradient descent?

Can Batch Gradient Descent also help generalization (inject inductive bias)?

$$\min_w L_S(w) \qquad \text{using } w^{(t+1)} \leftarrow w^{(t)} - \eta_t \nabla L_S\left(w^{(t)}\right)$$

$$w^{(t)} \xrightarrow{t\to\infty} \arg\min L_S(w) \qquad \text{, but which minimizer??}$$

- Consider $h_w(x) = \langle w, \phi(x) \rangle$, $\phi(x) \in \mathbb{R}^D$, $D \gg m$, $\ell(h_w(x), y) = |h_w(x) - y|$
- If data in ``general position'': $\exists w\ L_S(w) = 0$, in fact an entire $D - m$ dim space of minimizers!

Claim: starting from $w^{(0)} = 0$, $w^{(t)} \xrightarrow{t\to\infty} \arg\min \|w\|_2$ s.t. $L_S(w) = 0$

Proof:

(1) $w^{(t)} \in span\left(\phi(x_1), \dots, \phi(x_m)\right)$

$$\nabla L_S(w) = \sum \ell'(\dots)\phi(x_i) \in span\left(\phi(x_1), \dots, \phi(x_m)\right)$$

$$w^{(t)} = -\sum \eta_t \nabla L_S\left(w^{(j)}\right) \in span\left(\nabla L_S\left(w^{(j)}\right)\right) \subseteq span\left(\phi(x_1), \dots, \phi(x_m)\right)$$

(2) If $w \in span\left(\phi(x_1), \dots, \phi(x_m)\right)$ and $\langle w, \phi(x_i) \rangle = y_i$, then it's the min norm solution

consider $w + w_\parallel + w_\perp$. Any $w_\perp \neq 0$ would violate constraints, and any $w_\parallel \neq 0$ would increase norm

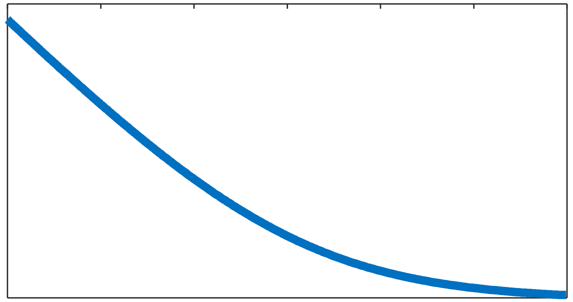
Can Batch Gradient Descent also help generalization (inject inductive bias)?

$\min_w L_S(w)$ using $w^{(t+1)} \leftarrow w^{(t)} - \eta_t \nabla L_S^{lgstc}\left(w^{(t)}\right)$

$w^{(t)} \xrightarrow{t\to\infty} \arg\min L_S(w)$, but which minimizer??

- Consider $h_w(x) = \langle w, \phi(x)\rangle$, $\phi(x) \in \mathbb{R}^D$, $D \gg m$, $\ell^{lgstc}(h_w(x), y) = \log\left(1 + e^{-y h_w(x)}\right)$
- Data linear separable: $\exists w\ \forall_i y_i \langle w, \phi(x_i)\rangle > 0$

$L_S^{lgstc}\left(w^{(t)}\right) \to 0$



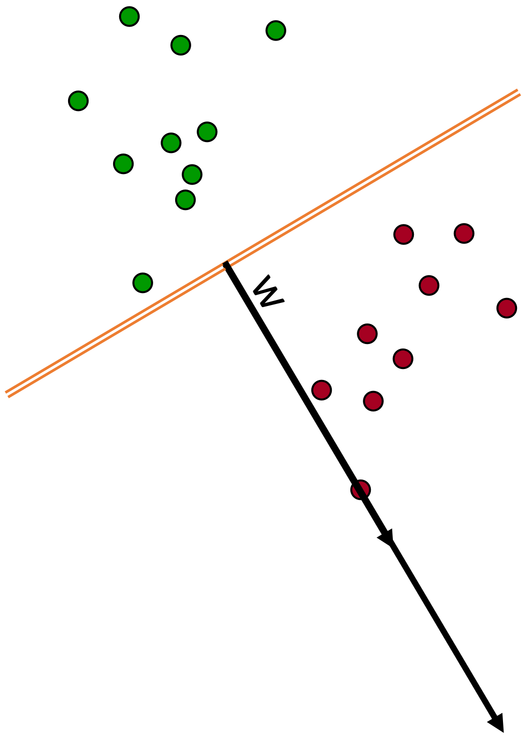
$w^{(t)} \xrightarrow{t\to\infty} ??$

$w^{(t)} \to \infty!$

But in what direction?

$sign\left(\left\langle w^{(t)}, \phi(x)\right\rangle\right) \to ??$

$\frac{w^{(t)}}{\left\|w^{(t)}\right\|} \to ??$



Can Batch Gradient Descent also help generalization (inject inductive bias)?

$\min_w L_S(w)$ using $w^{(t+1)} \leftarrow w^{(t)} - \eta_t \nabla L_S^{lgstc}\left(w^{(t)}\right)$

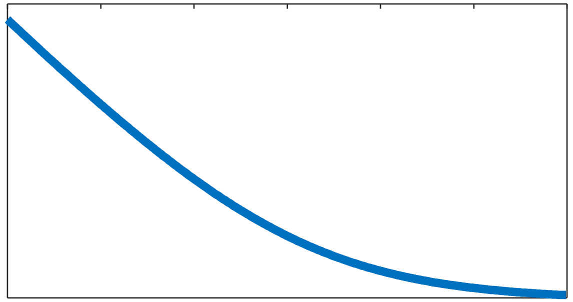
$w^{(t)} \xrightarrow{t\to\infty} \arg\min L_S(w)$, but which minimizer??

- Consider $h_w(x) = \langle w, \phi(x)\rangle$, $\phi(x) \in \mathbb{R}^D$, $D \gg m$, $\ell^{lgstc}(h_w(x), y) = \log\left(1 + e^{-yh_w(x)}\right)$
- Data linear separable: $\exists w\ \forall_i y_i \langle w, \phi(x_i)\rangle > 0$

$L_S^{lgstc}\left(w^{(t)}\right) \to 0$

$w^{(t)} \xrightarrow{t\to\infty} ??$

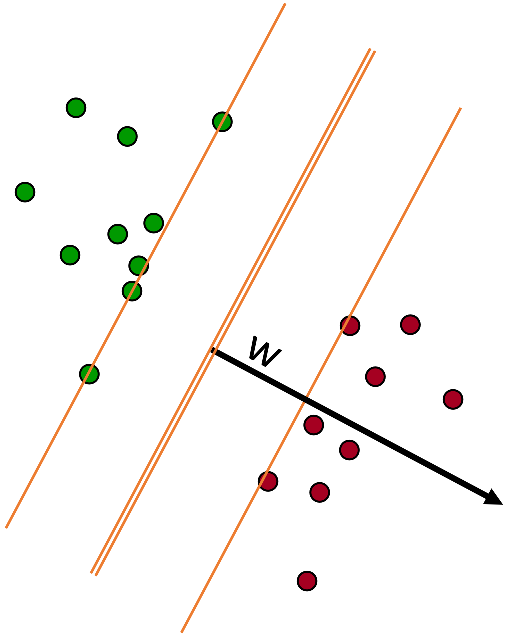
$w^{(t)} \to \infty$!



But in what direction?

$sign\left(\langle w^{(t)}, \phi(x)\rangle\right) \to ??$

$\frac{w^{(t)}}{\|w^{(t)}\|} \to ??$



Claim: $\frac{w(t)}{\|w(t)\|_2} \xrightarrow{t\to\infty} \frac{\hat{w}}{\|\hat{w}\|_2}$

$\hat{w} = \arg\min \|w\|_2 \ \ s.t. \forall_i y_i \langle w, x_i\rangle \geq 1$

- **Gradient Descent (or Multi-Pass SGD) on $L_S(w)$ converges** to $\arg\min \|w\|_2 \; s.t. \; L_S(w) = 0$

(with $\ell^{abs}(h_w(x), y) = |h_w(x) - y|$ or $\ell^{sq}(h_w(x), y) = (h_w(x) - y)^2$)

or $\propto \arg\min \|w\|_2 \; s.t. \; L_s^{\text{margin}}(w) = 0$ (with ℓ^{lgstc})

$\equiv$ **MDL for $\|w\|_2$**

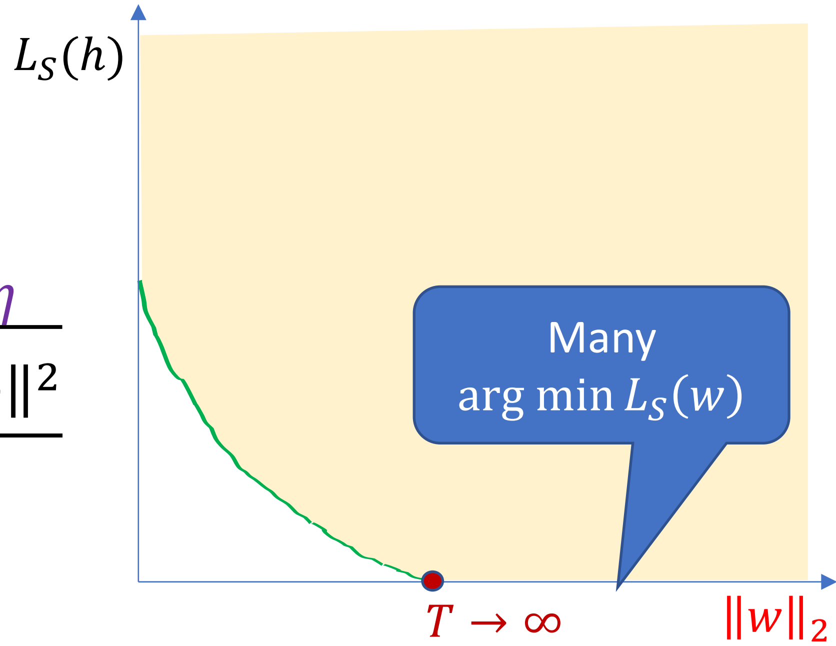
- **One-Pass ("Online") Stochastic Gradient Descent**

Learning with $\|w\|_2$ inductive bias

complexity/fit tradeoff controlled by stepsize ("learning rate") η

$$L(w) \le \inf_{\|w^*\|_2 \le \eta \|\phi\| m} L(w^*) + \eta \|\phi\|^2 \le \inf_{\|w^*\|_2 \le B} L(w^*) + \sqrt{\frac{B^2 \|\phi\|^2}{m}}$$

with $\eta = \frac{B}{\|\phi\|\sqrt{m}}$



- **Gradient Descent (or Multi-Pass SGD) on $L_S(w)$ converges** to $\arg\min\|w\|_2 \; s.t.\; L_S(w) = 0$

  or $\propto \arg\min\|w\|_2 \; s.t.\; L_S^{\text{margin}}(w) = 0$ (with ℓ^{lgstc})

  (with $\ell^{abs}(h_w(x), y) = |h_w(x) - y|$ or $\ell^{sq}(h_w(x), y) = (h_w(x) - y)^2$)

  $\equiv$ MDL for $\|w\|_2$

- Gradient Descent or Multi-Pass SGD with Early Stopping

  provides complexity control related to $\|w\|_2$

  generalization similar to RERM, $\arg\min L_S(w) + \lambda\|w\|_2$

  tradeoff controlled by **stepsize** *and* **stopping time (#iterations)**

- **One-Pass ("Online") Stochastic Gradient Descent**

  Learning with $\|w\|_2$ inductive bias

  complexity/fit tradeoff controlled by stepsize ("learning rate") η

Draw $(x_1, y_1), \ldots, (x_m, y_m) \sim \mathcal{D}$
Initialize $w^{(0)} = 0$
At iteration $t=1..T$:
- Pick $i \in 1 \ldots m$ at random
- If $y_i\langle w^{(t)}, \phi(x_i)\rangle < 1$,
  $w^{(t+1)} \leftarrow w^{(t)} + \eta_t y_i \phi(x_i)$
  else: $w^{(t+1)} \leftarrow w^{(t)}$
- $w^{(t+1)} \leftarrow proj\; w^{(t+1)}\; to\; \|w\| \leq B$

Return $\bar{w}^{(T)} = \frac{1}{T}\sum_{t=1}^{T} w^{(t)}$

$$L(w) \leq \inf_{\|w^*\|_2 \leq \eta\|\phi\|m} L(w^*) + \eta\|\phi\|^2 \leq \inf_{\|w^*\|_2 \leq B} L(w^*) + \sqrt{\frac{B^2\|\phi\|^2}{m}}$$

with $\eta = \frac{B}{\|\phi\|\sqrt{m}}$

- **Gradient Descent (or Multi-Pass SGD) on $L_S(w)$ converges** to $\mathbf{\arg\min \|w\|_2 \; s.t. \; L_S(w) = 0}$

  or $\propto \mathbf{\arg\min \|w\|_2 \; s.t. \; L_s^{\text{margin}}(w) = 0}$ (with ℓ^{lgstc})

  (with $\ell^{abs}(h_w(x), y) = |h_w(x) - y|$ or $\ell^{sq}(h_w(x), y) = (h_w(x) - y)^2$)

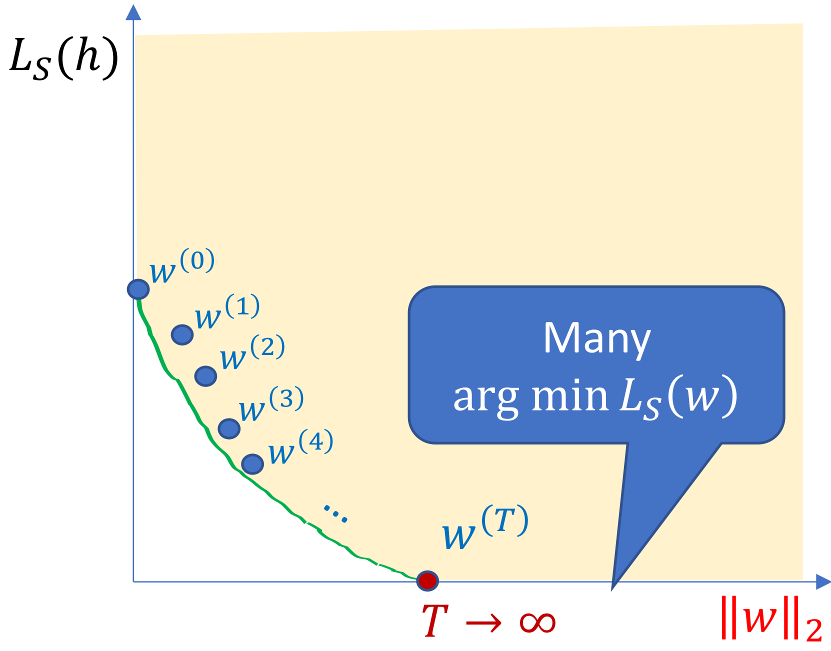
  $\equiv$ MDL for $\|w\|_2$

- Gradient Descent or Multi-Pass SGD with Early Stopping

  provides complexity control related to $\|w\|_2$

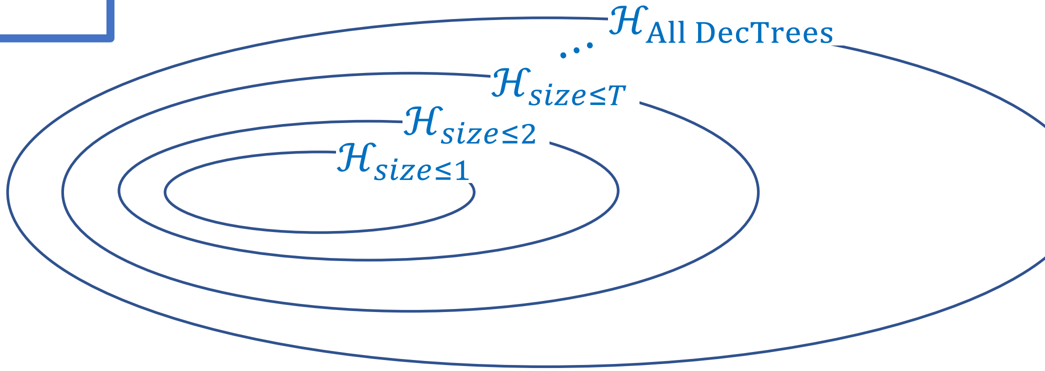
  generalization similar to RERM, $\arg\min L_S(w) + \lambda \|w\|_2$

  tradeoff controlled by **stepsize** *and* **stopping time (#iterations)**



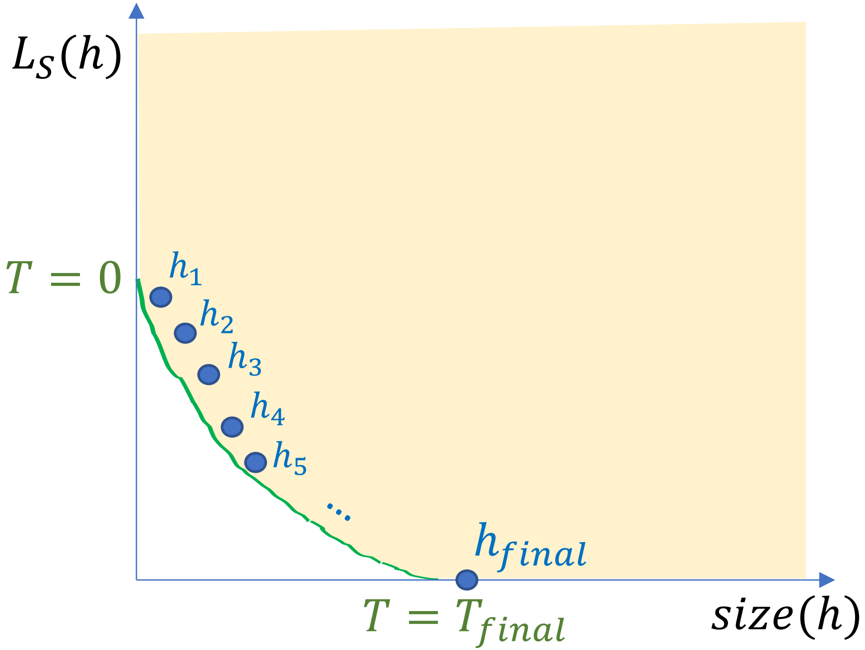
Greedy Decision Tree Construction, minimizing $L_S(h)$

```
Init empty decision tree h_0
While some nodes in h_t are impure (have ≥ 1 train label):
    Pick node v and feature that maxs train error reduction
    Split v according to predicate to obtain h_{t+1}
```



$$h_{final} \approx \arg\min_{L_S(h)=0} size(h_T)$$

- But early stopping after T iterations: $size(h_T) \le T$
- Early stopping corresponds to controlling the inductive bias "decision tree size"
- How early we step $\equiv$ balance between fit and complexity $\equiv$ where we are on regularization path



- One-Pass (“Online”) Stochastic Gradient Descent

  Learning with $\|w\|_2$ inductive bias

  complexity/fit tradeoff controlled by stepsize (“learning rate”) η

- Multi-Pass SGD or Batch Gradient Descent with Early Stopping

  provides complexity control related to $\|w\|_2$

  generalization properties similar to RERM, $\arg\min L_S(w) + \lambda\|w\|_2$

  tradeoff controlled by stepsize *and* stopping time (#iterations)

- Multi-Pass SGD or Batch Gradient Descent to Convergence

  $\approx$ MDL, $\arg\min\|w\|_2$

- When $D \gg m$, for $h_w(x) = \langle w, \phi(x) \rangle$, there are MANY $\mathbf{arg\,min}\, \boldsymbol{L_s(w)}$

- **Gradient Descent on $\boldsymbol{L_S(w)}$** converges to $\mathbf{arg\,min} \|\boldsymbol{w}\|_2 \; \boldsymbol{s.t.\, L_s(w) = 0}$

  (with $\ell^{abs}(h_w(x), y) = |h_w(x) - y|$ or $\ell^{sq}(h_w(x), y) = (h_w(x) - y)^2$)

  or $\propto \mathbf{arg\,min} \|\boldsymbol{w}\|_2 \; \boldsymbol{s.t.\, L_s^{\mathbf{margin}}(w) = 0}$ (with ℓ^{lgstc})

  $\equiv$ MDL for $\|\boldsymbol{w}\|_2$

- **This is specific to the optimization method!**

Instead:

- **Coordinate descent:**

$$i^{(t)} = \arg\max \left| \partial_i L_S\left(w^{(t)}\right) \right|$$

$$w^{(t+1)} = \arg\min L(w) \quad w = w^{(t)} + \eta e_i$$

Bias towards sparser solutions!

With logistic loss, $\rightarrow \propto$ $\mathbf{arg\,min} \|\boldsymbol{w}\|_1 \; s.t.\, L_S^{\mathrm{margin}}(w) = 0$