Learning Max-Sum classifiers by Structured Output SVM

Vojtěch Franc

April 12, 2022

- Learning Max-Sum classifiers on acyclic graphs
- Learning Max-Sum classifiers with super-modular functions
- Learning generic Max-Sum classifiers via LP relaxation

XEP33SML – Structured Model Learning, Summer 2022

Structured Output Support Vector Machines

- Given $\mathcal{T} = \{(x^i, y^i) \in \mathcal{X} \times \mathcal{Y} \mid i = 1, \ldots, m\}$ and a feature map $\boldsymbol{\phi}\colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$, we want to learn $\boldsymbol{w} \in \mathbb{R}^n$ of a classifier

$$h(\boldsymbol{x}; \boldsymbol{w}) = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}} \langle \boldsymbol{w}, \boldsymbol{\phi}(x, y) \rangle$$

Structured Output Support Vector Machines

- Given $\mathcal{T} = \{(x^i, y^i) \in \mathcal{X} \times \mathcal{Y} \mid i = 1, \ldots, m\}$ and a feature map $\boldsymbol{\phi}\colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$, we want to learn $\boldsymbol{w} \in \mathbb{R}^n$ of a classifier

$$h(\boldsymbol{x}; \boldsymbol{w}) = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}} \langle \boldsymbol{w}, \boldsymbol{\phi}(x, y) \rangle$$

- SO-SVM with margin rescaling loss find $\boldsymbol{w}$ by solving a convex problem

$$\boldsymbol{w}^* = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^n} \left(\frac{\lambda}{2} \|\boldsymbol{w}\|^2 + R^\psi(\boldsymbol{w}) \right)$$

where

$$R^\psi(\boldsymbol{w}) = \frac{1}{m} \sum_{i=1}^{m} \max_{y \in \mathcal{Y}} \Big(\ell(y^i, y) + \langle \boldsymbol{w}, \boldsymbol{\phi}(x^i, y) \rangle - \langle \boldsymbol{w}, \boldsymbol{\phi}(x^i, y^i) \rangle \Big)$$

Structured Output Support Vector Machines

- Given $\mathcal{T} = \{(x^i, y^i) \in \mathcal{X} \times \mathcal{Y} \mid i = 1, \ldots, m\}$ and a feature map $\boldsymbol{\phi}\colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^n$, we want to learn $\boldsymbol{w} \in \mathbb{R}^n$ of a classifier

$$h(\boldsymbol{x}; \boldsymbol{w}) = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}} \langle \boldsymbol{w}, \boldsymbol{\phi}(x, y) \rangle$$

- SO-SVM with margin rescaling loss find $\boldsymbol{w}$ by solving a convex problem

$$\boldsymbol{w}^* = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^n} \left(\frac{\lambda}{2} \|\boldsymbol{w}\|^2 + R^{\psi}(\boldsymbol{w}) \right)$$

where

$$R^{\psi}(\boldsymbol{w}) = \frac{1}{m} \sum_{i=1}^{m} \max_{y \in \mathcal{Y}} \Big(\ell(y^i, y) + \langle \boldsymbol{w}, \boldsymbol{\phi}(x^i, y) \rangle - \langle \boldsymbol{w}, \boldsymbol{\phi}(x^i, y^i) \rangle \Big)$$

- For every loss $\ell\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ such that $\ell(y, y') = 0 \iff y = y'$, it holds that $R^{\psi}(\boldsymbol{w}) \geq R_{\mathcal{T}^m}(h) = \frac{1}{m} \sum_{i=1}^{m} \ell(y^i, h(x^i; \boldsymbol{w}))$.

SO-SVM solved via Cutting Plane Method

- The first order oracle computes the risk

$$R^{\psi}(\boldsymbol{w}) = \frac{1}{m}\sum_{i=1}^{m} \max_{y\in\mathcal{Y}} \Big(\ell(y^i, y) + \langle \boldsymbol{w}, \boldsymbol{\phi}(x^i, y)\rangle - \langle \boldsymbol{w}, \boldsymbol{\phi}(x^i, y^i)\rangle \Big)$$

and one of its sub-gradient $\boldsymbol{g} \in \partial R^{\psi}(\boldsymbol{w})$ at any $\boldsymbol{w} \in \mathbb{R}^n$, e.g.

$$\boldsymbol{g} = \frac{1}{m}\sum_{i=1}^{m} \Big(\boldsymbol{\phi}(x^i, \hat{y}^i) - \boldsymbol{\phi}(x^i, y^i) \Big)$$

SO-SVM solved via Cutting Plane Method

- The first order oracle computes the risk

$$R^{\psi}(\boldsymbol{w}) = \frac{1}{m}\sum_{i=1}^{m} \max_{y\in\mathcal{Y}} \Big(\ell(y^i, y) + \langle \boldsymbol{w}, \boldsymbol{\phi}(x^i, y)\rangle - \langle \boldsymbol{w}, \boldsymbol{\phi}(x^i, y^i)\rangle \Big)$$

and one of its sub-gradient $\boldsymbol{g} \in \partial R^{\psi}(\boldsymbol{w})$ at any $\boldsymbol{w} \in \mathbb{R}^n$, e.g.

$$\boldsymbol{g} = \frac{1}{m}\sum_{i=1}^{m} \Big(\boldsymbol{\phi}(x^i, \hat{y}^i) - \boldsymbol{\phi}(x^i, y^i) \Big)$$

- To this end, we need to solve the loss augmented classification problem

$$\hat{y}^i = \operatorname*{argmax}_{y\in\mathcal{Y}} \Big(\ell(y^i, y) + \langle \boldsymbol{w}, \boldsymbol{\phi}(x^i, y)\rangle \Big)$$

Max-sum classifier and Hamming loss

- The max-sum classifier

$$\hat{\boldsymbol{y}} = \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \left\langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}) \right\rangle := \sum_{v\in\mathcal{V}} q_v(x_v, y_v) + \sum_{\{v,v'\}\in\mathcal{E}} g_{vv'}(y_v, y_{v'})$$

where $q_v(x,y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_v(x,y)\rangle$ and $g_{vv'}(y,y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_{vv'}(y,y')\rangle$

Max-sum classifier and Hamming loss

- The max-sum classifier

$$\hat{\boldsymbol{y}} = \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \big\langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y})\big\rangle := \sum_{v\in\mathcal{V}} q_v(x_v,y_v) + \sum_{\{v,v'\}\in\mathcal{E}} g_{vv'}(y_v,y_{v'})$$

where $q_v(x,y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_v(x,y)\rangle$ and $g_{vv'}(y,y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_{vv'}(y,y')\rangle$

- Hamming loss $\ell(\boldsymbol{y},\boldsymbol{y}') = \sum_{v\in\mathcal{V}} [\![y_v \neq y'_{v'}]\!]$

Max-sum classifier and Hamming loss

- The max-sum classifier

$$\hat{\boldsymbol{y}} = \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \left\langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y}) \right\rangle := \sum_{v\in\mathcal{V}} q_v(x_v, y_v) + \sum_{\{v,v'\}\in\mathcal{E}} g_{vv'}(y_v, y_{v'})$$

where $q_v(x,y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_v(x,y)\rangle$ and $g_{vv'}(y,y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_{vv'}(y,y')\rangle$

- Hamming loss $\ell(\boldsymbol{y},\boldsymbol{y}') = \sum_{v\in\mathcal{V}} [\![y_v \neq y'_{v'}]\!]$

- The loss Augmented Classification Problem

$$\begin{aligned} \hat{\boldsymbol{y}}^i &= \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \left[\ell(\boldsymbol{y}^i, \boldsymbol{y}) + \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}^i, \boldsymbol{y})\rangle \right] \\ &= \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \left[\sum_{v\in\mathcal{V}} \Big([\![y^i_v \neq y_v]\!] + q_v(x^i, y_v) \Big) + \sum_{\{v,v'\}\in\mathcal{E}} g_{vv'}(y_v, y_{v'}) \right] \end{aligned}$$

Max-sum classifier and Hamming loss

- The max-sum classifier

$$\hat{\boldsymbol{y}} = \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y})\rangle := \sum_{v\in\mathcal{V}} q_v(x_v,y_v) + \sum_{\{v,v'\}\in\mathcal{E}} g_{vv'}(y_v,y_{v'})$$

where $q_v(x,y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_v(x,y)\rangle$ and $g_{vv'}(y,y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_{vv'}(y,y')\rangle$

- Hamming loss $\ell(\boldsymbol{y},\boldsymbol{y}') = \sum_{v\in\mathcal{V}} [\![y_v \neq y'_{v'}]\!]$
- The loss Augmented Classification Problem

$$\begin{aligned}\hat{\boldsymbol{y}}^i &= \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \Big[\ell(\boldsymbol{y}^i,\boldsymbol{y}) + \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}^i,\boldsymbol{y})\rangle\Big] \\ &= \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \Big[\sum_{v\in\mathcal{V}} \Big([\![y_v^i \neq y_v]\!] + q_v(x^i,y_v)\Big) + \sum_{\{v,v'\}\in\mathcal{E}} g_{vv'}(y_v,y_{v'})\Big]\end{aligned}$$

- The ACP is tractable for acyclic graph $(\mathcal{V},\mathcal{E})$.

Super-modular Max-sum classifier and Hamming loss

- The max-sum classifier

$$\hat{\boldsymbol{y}} = \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x},\boldsymbol{y})\rangle := \sum_{v\in\mathcal{V}} q_v(x_v, y_v) + \sum_{\{v,v'\}\in\mathcal{E}} g_{vv'}(y_v, y_{v'})$$

where $g_{vv'}(y,y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_{vv'}(y,y')\rangle$ is super-modular.

Super-modular Max-sum classifier and Hamming loss

- The max-sum classifier

$$\hat{\boldsymbol{y}} = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} \big\langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}) \big\rangle := \sum_{v \in \mathcal{V}} q_v(x_v, y_v) + \sum_{\{v,v'\} \in \mathcal{E}} g_{vv'}(y_v, y_{v'})$$

where $g_{vv'}(y, y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_{vv'}(y, y') \rangle$ is super-modular.

- SO-SVM leads to

$$\boldsymbol{w}^* = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^n} \left(\frac{\lambda}{2} \|\boldsymbol{w}\|^2 + R^{\psi}(\boldsymbol{w}) \right)$$

subject to

$$g_{vv'}(y, y') + g_{vv'}(y+1, y'+1) - g_{vv'}(y, y'+1) - g_{vv'}(y+1, y') \geq 0\,,$$
$$\{v, v'\} \in \mathcal{E}, y, y' \in \{1, \ldots, K-1\}$$

Super-modular Max-sum classifier and Hamming loss

- The max-sum classifier

$$\hat{\boldsymbol{y}} = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} \left\langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}) \right\rangle := \sum_{v \in \mathcal{V}} q_v(x_v, y_v) + \sum_{\{v,v'\} \in \mathcal{E}} g_{vv'}(y_v, y_{v'})$$

  where $g_{vv'}(y, y) = \langle \boldsymbol{w}, \boldsymbol{\phi}_{vv'}(y, y') \rangle$ is super-modular.

- SO-SVM leads to

$$\boldsymbol{w}^* = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^n} \left(\frac{\lambda}{2} \|\boldsymbol{w}\|^2 + R^{\psi}(\boldsymbol{w}) \right)$$

subject to

$$g_{vv'}(y, y') + g_{vv'}(y+1, y'+1) - g_{vv'}(y, y'+1) - g_{vv'}(y+1, y') \geq 0\,,$$
$$\{v, v'\} \in \mathcal{E}, y, y' \in \{1, \ldots, K-1\}$$

- Provided the solver maintains intermediate solution $\boldsymbol{w}$ feasible the ACPs are sub-modular and thus tractable.

BMRM with constraints

- Constrained regularized convex risk minimization

$$\boldsymbol{w}^* \in \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\frac{\lambda}{2}\|\boldsymbol{w}\|^2 + R(\boldsymbol{w}) \right) \quad \text{s.t.} \quad \mathbf{A}\boldsymbol{w} \leq \boldsymbol{b}$$

BMRM with constraints

- Constrained regularized convex risk minimization

$$\boldsymbol{w}^* \in \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\frac{\lambda}{2}\|\boldsymbol{w}\|^2 + R(\boldsymbol{w})\right) \quad \text{s.t.} \quad \mathbf{A}\boldsymbol{w} \le \boldsymbol{b}$$

- The BMRM algorithm:

1. Init: $t \leftarrow 0$, $\boldsymbol{w}_0 \in \mathbb{R}^n$
2. Compute $R(\boldsymbol{w}_t)$ and $\boldsymbol{g}_t \in \partial R(\boldsymbol{w}_t)$
3. Solve the constrained reduced problem

$$\boldsymbol{w}_{t+1} = \operatorname*{argmin}_{\boldsymbol{w}\in\mathbb{R}^n} \left(\frac{\lambda}{2}\|\boldsymbol{w}\|^2 + R_t(\boldsymbol{w})\right) \quad \text{s.t.} \quad \mathbf{A}\boldsymbol{w} \le \boldsymbol{b}$$

   where

$$R_t(\boldsymbol{w}) = \max_{i=0,\ldots,t} \left[R(\boldsymbol{w}_i) + \langle \boldsymbol{g}_i, \boldsymbol{w} - \boldsymbol{w}_i\rangle\right]$$

4. if $\min_{i=1,\ldots,t} F(\boldsymbol{w}_i) - F_t(\boldsymbol{w}_{t+1}) \le \varepsilon$ stop else $t \leftarrow t+1$ go to 2.

General max-sum classifier learned via LP relaxation

- The ACP leads to

$$\hat{\boldsymbol{y}}^i = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} f^i(\boldsymbol{y}, \boldsymbol{w}) := \sum_{v \in \mathcal{V}} \Big([\![y_v^i \neq y_v]\!] + q_v(x^i, y_v) \Big) + \sum_{\{v,v'\} \in \mathcal{E}} g_{vv'}(y_v, y_{v'})$$

General max-sum classifier learned via LP relaxation

- The ACP leads to

$$\hat{\boldsymbol{y}}^i = \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} f^i(\boldsymbol{y},\boldsymbol{w}) := \sum_{v\in\mathcal{V}} \Big([\![y_v^i \neq y_v]\!] + q_v(x^i, y_v) \Big) + \sum_{\{v,v'\}\in\mathcal{E}} g_{vv'}(y_v, y_{v'})$$

- The value of ACP can be upper bounded via the LP relaxation:

$$\max_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} f^i(\boldsymbol{y},\boldsymbol{w}) \leq \min_{\boldsymbol{\varphi}} E^i(\boldsymbol{\varphi},\boldsymbol{w})$$

General max-sum classifier learned via LP relaxation

- The ACP leads to

$$\hat{\boldsymbol{y}}^i = \operatorname*{argmax}_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} f^i(\boldsymbol{y},\boldsymbol{w}) := \sum_{v\in\mathcal{V}} \Big([\![y_v^i \neq y_v]\!] + q_v(x^i, y_v) \Big) + \sum_{\{v,v'\}\in\mathcal{E}} g_{vv'}(y_v, y_{v'})$$

- The value of ACP can be upper bounded via the LP relaxation:

$$\max_{\boldsymbol{y}\in\mathcal{Y}^{\mathcal{V}}} f^i(\boldsymbol{y},\boldsymbol{w}) \leq \min_{\boldsymbol{\varphi}} E^i(\boldsymbol{\varphi},\boldsymbol{w})$$

where $\boldsymbol{\varphi} \in \mathbb{R}^{2|\mathcal{E}||\mathcal{Y}|}$ is composed of $\varphi_{vv'}, \varphi_{v'v} \colon \mathcal{Y} \to \mathbb{R}$, $\{v,v'\} \in \mathcal{E}$ and

$$E^i(\boldsymbol{\varphi},\boldsymbol{w}) = \sum_{v\in\mathcal{V}} \max_{y\in\mathcal{Y}} q_v^{\boldsymbol{\varphi},\boldsymbol{w}}(y, x^i, y_v^i) + \sum_{\{v,v'\}\in\mathcal{E}} \max_{(y,y')\in\mathcal{Y}^2} g_{vv'}^{\boldsymbol{\varphi},\boldsymbol{w}}(y,y')$$

$$\begin{aligned} q_v^{\boldsymbol{\varphi},\boldsymbol{w}}(y, x^i, y_v^i) &= [\![y_v^i \neq y_v]\!] + q_v(x^i, y_v) - \sum_{v'\in\mathcal{N}(v)} \varphi_{vv'}(y), \quad v\in\mathcal{V}, y\in\mathcal{Y} \\ g_{vv'}^{\boldsymbol{\varphi},\boldsymbol{w}}(y,y') &= g_{vv'}(y,y') + \varphi_{vv'}(y) + \varphi_{v'v}(y'), \quad \{v,v'\}\in\mathcal{E}, y,y'\in\mathcal{Y} \end{aligned}$$

General max-sum classifier learned via LP relaxation

- The LP-relaxed margin-rescaling loss:

$$
\begin{aligned}
\psi(\boldsymbol{x}^i, \boldsymbol{y}^i, \boldsymbol{w}) &= \max_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} \Big(\ell(\boldsymbol{y}^i, \boldsymbol{y}) + \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}^i, \boldsymbol{y}) \rangle \Big) - \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}^i, \boldsymbol{y}^i) \rangle \\
&\leq \min_{\boldsymbol{\varphi}} E^i(\boldsymbol{\varphi}, \boldsymbol{w}) - \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}^i, \boldsymbol{y}^i) \rangle \\
&= \psi_{\mathrm{LP}}(\boldsymbol{x}^i, \boldsymbol{y}^i, \boldsymbol{w})
\end{aligned}
$$

General max-sum classifier learned via LP relaxation

- The LP-relaxed margin-rescaling loss:

$$
\begin{aligned}
\psi(\boldsymbol{x}^i, \boldsymbol{y}^i, \boldsymbol{w}) &= \max_{\boldsymbol{y} \in \mathcal{Y}^{\mathcal{V}}} \Big(\ell(\boldsymbol{y}^i, \boldsymbol{y}) + \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}^i, \boldsymbol{y}) \rangle \Big) - \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}^i, \boldsymbol{y}^i) \rangle \\
&\leq \min_{\boldsymbol{\varphi}} E^i(\boldsymbol{\varphi}, \boldsymbol{w}) - \langle \boldsymbol{w}, \boldsymbol{\phi}(\boldsymbol{x}^i, \boldsymbol{y}^i) \rangle \\
&= \psi_{\mathrm{LP}}(\boldsymbol{x}^i, \boldsymbol{y}^i, \boldsymbol{w})
\end{aligned}
$$

- SO-SVM leads to

$$
\boldsymbol{w}^* = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^n} \left(\frac{\lambda}{2} \|\boldsymbol{w}\|^2 + R^{\psi}(\boldsymbol{w}) \right)
$$

where

$$
R^{\psi}(\boldsymbol{w}) = \frac{1}{m} \sum_{i=1}^{m} \psi_{\mathrm{LP}}(\boldsymbol{x}^i, \boldsymbol{y}^i, \boldsymbol{w})
$$

Stochastic Gradient Descent

- Let us consider a convex constrained problem

$$\boldsymbol{w}^* \in \operatorname*{argmin}_{\boldsymbol{w}\in\mathcal{W}} F(\boldsymbol{w})$$

  where $\mathcal{W} \subset \mathbb{R}^n$ is a closed convex set and $F\colon \mathcal{W} \to \mathbb{R}$ is convex.

- SGD uses oracle which for given $\boldsymbol{w}^t$ provides a stochastic estimate $\hat{\boldsymbol{g}}^t$ of the sub-gradient $\boldsymbol{g}^t \in \partial F(\boldsymbol{w}^t)$ such that

$$\mathbb{E}\hat{\boldsymbol{g}}^t = \boldsymbol{g}^t$$

- For example, in our setting

$$F(\boldsymbol{w}) = \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \frac{1}{m}\sum_{i\in\mathcal{I}} \ell_i(\boldsymbol{w}) = \frac{1}{m}\sum_{i\in\mathcal{I}} \left(\frac{\lambda}{2}\|\boldsymbol{w}\|^2 + \ell_i(\boldsymbol{w})\right) = \frac{1}{m}\sum_{i\in\mathcal{I}} F_i(\boldsymbol{w})$$

  the oracle picks $i \in \mathcal{I}$ uniformly at randomly and provides a sub-gradient

$$\hat{\boldsymbol{g}}^t \in \partial F_i(\boldsymbol{w}^t)$$

Stochastic Gradient Descent

- The SGD algorithm: starting from $\boldsymbol{w}^1 = \boldsymbol{0}$, SGD computes new iterates recursively as follows

$$\boldsymbol{w}^{t+1} = \Pi_{\mathcal{W}}\Big(\boldsymbol{w}^t - \eta^t \hat{\boldsymbol{g}}^t\Big)$$

  where $\Pi_{\mathcal{W}} \colon \mathbb{R}^n \to \mathcal{W}$ denotes projection on $\mathcal{W}$, i.e.

$$\Pi_{\mathcal{W}}(\boldsymbol{w}) = \operatorname*{argmin}_{\boldsymbol{w}' \in \mathcal{W}} \|\boldsymbol{w}' - \boldsymbol{w}\|$$

  and η^t is a sequence of step-sizes.

- The theoretical results require a fixed step size, typically, $\sum_{t=0}^{\infty} \eta^t = \infty$ and $\lim_{t\to\infty} \eta^t = 0$.

- No stopping condition which would provide a certificate of optimality, instead, SGD is stoped based on monotorring the validation error.

Stochastic Gradient Descent - convergence guarantees

- Definition: function $F\colon \mathcal{W} \to \mathbb{R}$ is λ-strictly convex iff the function $F(\boldsymbol{w}) - \frac{\lambda}{2}\|\boldsymbol{w}\|^2$ is convex. E.g. $F(\boldsymbol{w}) = \frac{\lambda}{2}\|\boldsymbol{w}\|^2 + R(\boldsymbol{w})$ is λ-strictly convex iff $R(\boldsymbol{w})$ is convex.
- λ**-strictly convex functions:** Suppose F is λ-strictly convex, and that $\mathbb{E}[\|\hat{\boldsymbol{g}}^t\|^2] \leq G^2$, $\forall t$. Consider SGD with step sizes $\eta^t = \frac{1}{\lambda t}$. Then for any $t > 1$, it holds that

$$\mathbb{E}[F(\boldsymbol{w}^t) - F(\boldsymbol{w}^*)] \leq \frac{17G^2(1+\log(t))}{\lambda t}$$

- **Convex functions:** Assume that F is convex and that for some constants D, G it holds that $\mathbb{E}[\|\hat{\boldsymbol{g}}^t\|] \leq G$, $\forall t$, and $\sup_{\boldsymbol{w},\boldsymbol{w}' \in \mathcal{W}} \|\boldsymbol{w} - \boldsymbol{w}'\| \leq D$. Consider SGD with step size $\eta^t = \frac{c}{\sqrt{t}}$ where $c > 0$ is a constant. Then for any $t > 1$ it holds that

$$\mathbb{E}[F(\boldsymbol{w}^t) - F(\boldsymbol{w}^*)] \leq \left(\frac{D^2}{c} + cG^2\right)\frac{2+\log(t)}{\sqrt{t}}$$