First-Order Methods for Distributionally Robust Mixed-Integer Optimization

Hubert Villuendas, Mathieu Besançon & Jérôme Malick
: hubert.villuendas@univ-grenoble.fr

University Grenoble Alpes, France

June 2nd, 2026

SIAM, Edinburgh

Université Grenoble Alpes

From uncertainty to robustness

$$\underset{z \in \mathcal{Z}}{\text{minimize}} \quad \mathbf{E}_{\xi}\left[f(z, \xi)\right] \qquad \mathcal{Z} \subset \mathbf{R}^n \times \mathbf{Z}^{d-n}$$

- Known: samples $\left\{\widehat{\xi}_1, \ldots, \widehat{\xi}_N\right\} \subset \Xi$

From uncertainty to robustness

$$\underset{z \in \mathcal{Z}}{\text{minimize}} \quad \mathbf{E}_{\xi}\left[f(z,\xi)\right] \qquad \mathcal{Z} \subset \mathbf{R}^n \times \mathbf{Z}^{d-n}$$

- Known: samples $\left\{\widehat{\xi}_1, \ldots, \widehat{\xi}_N\right\} \subset \Xi$
- Empirical distribution:

$$\widehat{\mathbf{P}}_N = \frac{1}{N}\sum_{k=1}^{N} \delta_{\widehat{\xi}_k}$$

From uncertainty to robustness

$$\underset{z \in \mathcal{Z}}{\text{minimize}} \quad \mathbf{E}_{\xi}\left[f(z,\xi)\right] \qquad\qquad \mathcal{Z} \subset \mathbf{R}^n \times \mathbf{Z}^{d-n}$$

- Known: samples $\left\{\widehat{\xi}_1, \ldots, \widehat{\xi}_N\right\} \subset \Xi$
- Empirical distribution:

$$\widehat{\mathbf{P}}_N = \frac{1}{N}\sum_{k=1}^{N} \delta_{\widehat{\xi}_k}$$

- Empirical Risk Minimization (ERM):

$$\underset{z \in \mathcal{Z}}{\text{minimize}} \quad \left[\frac{1}{N}\sum_{k=1}^{N} f\left(z, \widehat{\xi}_k\right) = \mathbf{E}_{\xi \sim \widehat{\mathbf{P}}_N}\left[f(z,\xi)\right]\right]$$

From uncertainty to robustness

$$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \mathbf{E}_{\xi}\left[f(z,\xi)\right] \qquad\qquad \mathcal{Z}\subset \mathbf{R}^n\times\mathbf{Z}^{d-n}$$

- Known: samples $\left\{\widehat{\xi}_1,\ldots,\widehat{\xi}_N\right\}\subset\Xi$
- Empirical distribution:

$$\widehat{\mathbf{P}}_N=\frac{1}{N}\sum_{k=1}^{N}\delta_{\widehat{\xi}_k}$$

- Empirical Risk Minimization (ERM):

Subject to distributionnal shift!

$$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \left[\frac{1}{N}\sum_{k=1}^{N} f\left(z,\widehat{\xi}_k\right)=\mathbf{E}_{\xi\sim\widehat{\mathbf{P}}_N}\left[f(z,\xi)\right]\right]$$

From uncertainty to robustness

$$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \mathbf{E}_{\xi}\left[f(z,\xi)\right] \qquad\qquad \mathcal{Z}\subset\mathbf{R}^n\times\mathbf{Z}^{d-n}$$

- Known: samples $\left\{\widehat{\xi}_1,\ldots,\widehat{\xi}_N\right\}\subset\Xi$
- Empirical distribution:

$$\widehat{\mathbf{P}}_N=\frac{1}{N}\sum_{k=1}^{N}\delta_{\widehat{\xi}_k}$$

- Empirical Risk Minimization (ERM):

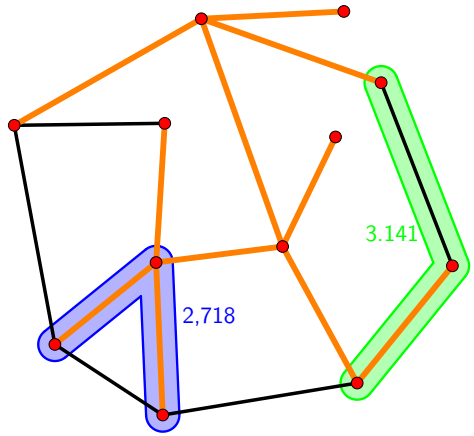
$$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \left[\frac{1}{N}\sum_{k=1}^{N}f\left(z,\widehat{\xi}_k\right)=\mathbf{E}_{\xi\sim\widehat{\mathbf{P}}_N}\left[f(z,\xi)\right]\right]$$

- Wasserstein Distributionally Robust Optimization (WDRO) **[*Esfahani and Kuhn, 2018*]**

$$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \sup_{W\left(\mathbf{Q},\widehat{\mathbf{P}}_N\right)\leq\rho} \mathbf{E}_{\zeta\sim\mathbf{Q}}\left[f(z,\zeta)\right]$$

Running example — Quadratic Spanning Tree

$G = (V, E)$ on m edges, $\mathcal{C} \in \mathbf{R}_+^{m \times m}$ cost matrix.

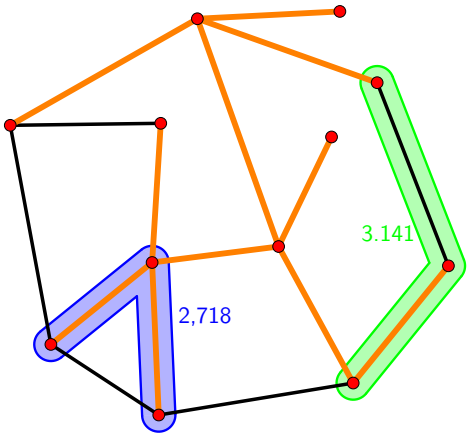


$$\underset{z \in \mathcal{Z}}{\text{minimize}} \quad z^\top \mathcal{C} z + c^\top z$$

$$z \in \{0,1\}^m,\ z_e = \begin{cases} \mathbf{1} & \text{if } e \in \mathscr{T} \\ 0 & \text{otherwise.} \end{cases}$$

Running example — Quadratic Spanning Tree

$G = (V, E)$ on m edges, $\mathcal{C} \in \mathbf{R}_+^{m \times m}$ cost matrix.



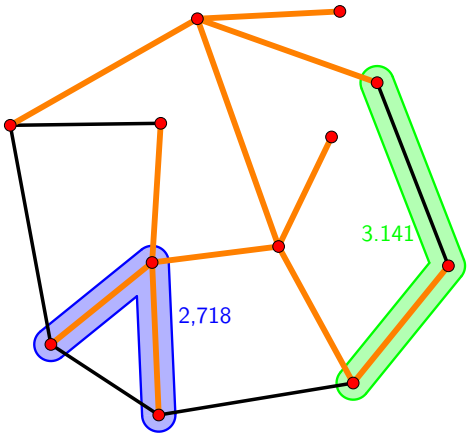
$$\underset{z \in \mathcal{Z}}{\text{minimize}} \quad z^\top \mathcal{C} z + c^\top z$$

$$z \in \{0,1\}^m,\ z_e = \begin{cases} \mathbf{1} & \text{if } e \in \mathscr{T} \\ 0 & \text{otherwise.} \end{cases}$$

- *Determinimistic Quadratic* minimimum spanning tree problem is $\mathscr{NP}$-hard **[*Assad and Xu, 1992*]**

Running example — Quadratic Spanning Tree

$G = (V, E)$ on m edges, $\mathcal{C} \in \mathbf{R}_+^{m \times m}$ cost matrix.



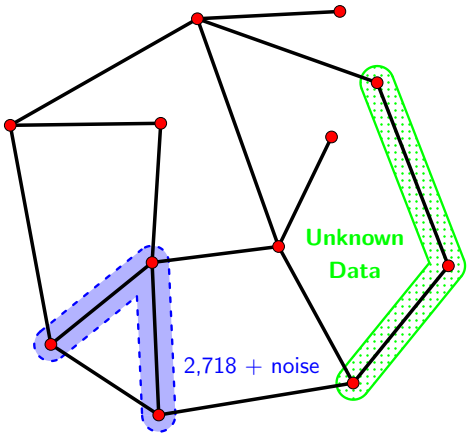
$$\underset{z \in \mathcal{Z}}{\text{minimize}} \quad z^\top \mathcal{C} z + c^\top z$$

$$z \in \{0,1\}^m,\ z_e = \begin{cases} 1 & \text{if } e \in \mathscr{T} \\ 0 & \text{otherwise.} \end{cases}$$

- *Determinimistic Quadratic* minimimum spanning tree problem is $\mathscr{NP}$-hard **[*Assad and Xu, 1992*]**
- *Determinimistic Linear* minimum spanning tree problem ($\mathcal{C} = 0$) is easy! Solved in $\mathcal{O}(m \log(m))$ by Kruskal algorithm **[*Schrijver, 2002*]**

Running example — Quadratic Spanning Tree

$G = (V, E)$ on m edges, $\mathcal{C} \in \mathbf{R}_+^{m \times m}$ cost matrix.



$$\underset{z \in \mathcal{Z}}{\text{minimize}} \quad z^\top \mathcal{C} z + c^\top z$$

$$z \in \{0,1\}^m,\ z_e = \begin{cases} 1 & \text{if } e \in \mathscr{T} \\ 0 & \text{otherwise.} \end{cases}$$

- *Determinimistic Quadratic* minimimum spanning tree problem is $\mathscr{NP}$-hard **[*Assad and Xu, 1992*]**
- *Determinimistic Linear* minimum spanning tree problem ($\mathcal{C} = 0$) is easy! Solved in $\mathcal{O}(m \log(m))$ by Kruskal algorithm **[*Schrijver, 2002*]**
- Here: *uncertain* costs! $\rightarrow$ really hard

Solving WDRO

$$\min_{z\in\mathcal{Z}} \sup_{W(\mathbf{Q},\widehat{\mathbf{P}}_N)\leq\rho} \mathbf{E}_{\zeta\sim\mathbf{Q}}\left[f(z,\zeta)\right]$$

$$= \min_{z\in\mathcal{Z}} \inf_{\lambda\in\mathbf{R}_+} \lambda\rho + \mathbf{E}_{\widehat{\xi}\sim\widehat{\mathbf{P}}_N}\left[\sup_{\zeta\in\Xi}\left\{f(z,\zeta) - \lambda c\left(\widehat{\xi},\zeta\right)\right\}\right]$$

[*Esfahani and Kuhn, 2018*]

Solving WDRO

$$\min_{z\in\mathcal{Z}} \sup_{W\left(\mathbf{Q},\widehat{\mathbf{P}}_N\right)\leq\rho} \mathbf{E}_{\zeta\sim\mathbf{Q}}\left[f(z,\zeta)\right]$$

$$= \min_{z\in\mathcal{Z}} \inf_{\lambda\in\mathbf{R}_+} \lambda\rho + \mathbf{E}_{\widehat{\xi}\sim\widehat{\mathbf{P}}_N}\left[\sup_{\zeta\in\Xi}\left\{f(z,\zeta) - \lambda c\left(\widehat{\xi},\zeta\right)\right\}\right]$$

[*Esfahani and Kuhn, 2018*]

Existing work for DRO in Operations Research:

- Chance-constrained vehicle routing **[*Ghosal and Wiesemann, 2020*]**
- Surgery allocation **[*Chow et al., 2022*]**
- Ressource allocation during a pandemic **[*Sun et al., 2023*]**

Solving WDRO

$$\min_{z\in\mathcal{Z}} \sup_{W\left(\mathbf{Q},\widehat{\mathbf{P}}_N\right)\leq\rho} \mathbf{E}_{\zeta\sim\mathbf{Q}}\left[f(z,\zeta)\right]$$

$$= \min_{z\in\mathcal{Z}} \inf_{\lambda\in\mathbf{R}_+} \lambda\rho + \mathbf{E}_{\widehat{\xi}\sim\widehat{\mathbf{P}}_N}\left[\sup_{\zeta\in\Xi}\left\{f(z,\zeta) - \lambda c\left(\widehat{\xi},\zeta\right)\right\}\right]$$

[*Esfahani and Kuhn, 2018*]

Existing work for DRO in Operations Research:

- Chance-constrained vehicle routing **[*Ghosal and Wiesemann, 2020*]**
- Surgery allocation **[*Chow et al., 2022*]**
- Ressource allocation during a pandemic **[*Sun et al., 2023*]**

→ our (*general*) approach: **Smoothing technique** and **First-order**

Solving WDRO

$$\min_{z\in\mathcal{Z}} \sup_{W\left(\mathbf{Q},\widehat{\mathbf{P}}_N\right)\leq\rho} \mathbf{E}_{\zeta\sim\mathbf{Q}}\left[f(z,\zeta)\right]$$

$$= \min_{z\in\mathcal{Z}} \inf_{\lambda\in\mathbf{R}_+} \lambda\rho + \mathbf{E}_{\widehat{\xi}\sim\widehat{\mathbf{P}}_N}\left[\sup_{\zeta\in\Xi}\left\{f(z,\zeta) - \lambda c\left(\widehat{\xi},\zeta\right)\right\}\right]$$

[*Esfahani and Kuhn, 2018*]

Existing work for DRO in Operations Research:

- Chance-constrained vehicle routing **[*Ghosal and Wiesemann, 2020*]**
- Surgery allocation **[*Chow et al., 2022*]**
- Ressource allocation during a pandemic **[*Sun et al., 2023*]**

→ our (*general*) approach: **Smoothing technique** and **First-order**

$$\underset{\varepsilon\to\mathbf{0}}{\simeq} \min_{z\in\mathcal{Z}} \inf_{\lambda\in\mathbf{R}_+} \lambda\rho + \varepsilon\mathbf{E}_{\widehat{\xi}\sim\widehat{\mathbf{P}}_N}\left[\log\left(\mathbf{E}_{\zeta\sim\mathcal{N}\left(\widehat{\xi},\sigma^2\mathbf{I}\right)}\left[\exp\left(\frac{f(z,\zeta) - \lambda c\left(\widehat{\xi},\zeta\right)}{\varepsilon}\right)\right]\right)\right]$$

[*Azizian et al., 2023*]

Introduced for unconstrained problems ($\mathcal{Z} = \mathbf{R}^d$)

We make it for constrained problems

Gradient estimate

Let $\mathcal{I} \subseteq [\![N]\!]$ "mini-batch"

Samples : $\zeta_1^{(k)}, \ldots, \zeta_S^{(k)} \sim \mathcal{N}\left(\widehat{\xi}_k, \sigma^2 \mathbf{I}\right) \quad \forall k \in \mathcal{I}$ **[*Vincent et al., 2024*]**

$$\mathcal{G}_z^{\mathcal{I}}(z, \lambda) \overset{\text{def}}{=} \frac{1}{|\mathcal{I}|} \sum_{k \in \mathcal{I}} \frac{\frac{1}{S} \sum_{j=1}^{S} \nabla_z f(z, \zeta_j^{(k)}) \exp\left(f(z, \zeta_j^{(k)}) - \lambda c(\zeta_j^{(k)}, \widehat{\xi}_k)/\varepsilon\right)}{\frac{1}{S} \sum_{j=1}^{S} \exp\left(f(z, \zeta_j^{(k)}) - \lambda c(\zeta_j^{(k)}, \widehat{\xi}_k)/\varepsilon\right)}$$

Gradient estimate

Let $\mathcal{I} \subseteq [\![N]\!]$ "mini-batch"

Samples : $\zeta_1^{(k)}, \dots, \zeta_S^{(k)} \sim \mathcal{N}\left(\widehat{\xi}_k, \sigma^2 \mathbf{I}\right) \quad \forall k \in \mathcal{I}$ **[*Vincent et al., 2024*]**

$$\mathcal{G}_z^{\mathcal{I}}(z,\lambda) \overset{\text{def}}{=} \frac{1}{|\mathcal{I}|} \sum_{k \in \mathcal{I}} \frac{\frac{1}{S}\sum_{j=1}^{S} \nabla_z f(z, \zeta_j^{(k)}) \exp\left(f(z, \zeta_j^{(k)}) - \lambda c(\zeta_j^{(k)}, \widehat{\xi}_k)/\varepsilon\right)}{\frac{1}{S}\sum_{j=1}^{S} \exp\left(f(z, \zeta_j^{(k)}) - \lambda c(\zeta_j^{(k)}, \widehat{\xi}_k)/\varepsilon\right)}$$

Proposition

Let $\mathcal{I}$: random subset. $\forall k \in \mathcal{I}$, let $\zeta_1^{(k)}, \dots, \zeta_S^{(k)} \sim \mathcal{N}\left(\widehat{\xi}_k, \sigma^2 \mathbf{I}\right)$ indenpendant samples. Then $\mathbf{E}\left[\mathcal{G}^{\mathcal{I}}(z,\lambda)\right] \underset{S \to +\infty}{\longrightarrow} \nabla F(z,\lambda)$ almost surely.

Stochastic Frank-Wolfe algorithm

Algorithm 1: *Momentum stochastic Frank-Wolfe* **[Braun et al., 2022]**

Input: Starting point $y_0 \in \mathcal{Y}$, step sizes γ_t and momentum terms $\beta_t \in [0, 1]$.

for $t = 0$ to $\ldots$ **do**

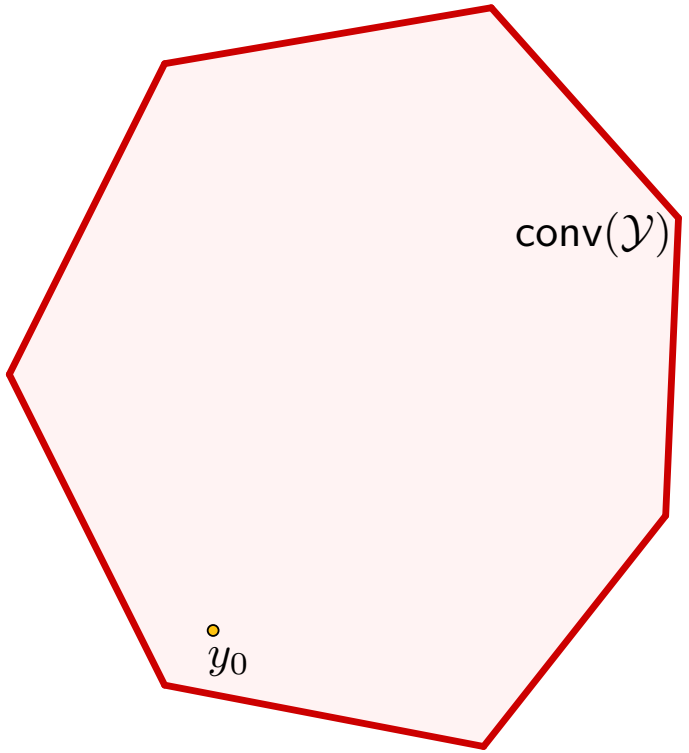
Take $\mathcal{I}_t \subseteq [\![N]\!]$ random

$\widehat{\nabla} F(y_t) \leftarrow \beta_t \mathcal{G}^{\mathcal{I}_t}(y_t) + (1 - \beta_t)\widehat{\nabla} F(y_{t-1})$

$v_t \leftarrow \operatorname{argmin}_{v \in \operatorname{conv}(\mathcal{Y})} \mathrm{LMO}\left(\widehat{\nabla} F(y_t), v\right)$

$y_{t+1} \leftarrow z_t + \gamma_t (v_t - y_t)$

end for



Stochastic Frank-Wolfe algorithm

Algorithm 1: *Momentum stochastic Frank-Wolfe* **[Braun et al., 2022]**

Input: Starting point $y_0 \in \mathcal{Y}$, step sizes γ_t and momentum terms $\beta_t \in [0, 1]$.

for $t = 0$ to $\ldots$ **do**

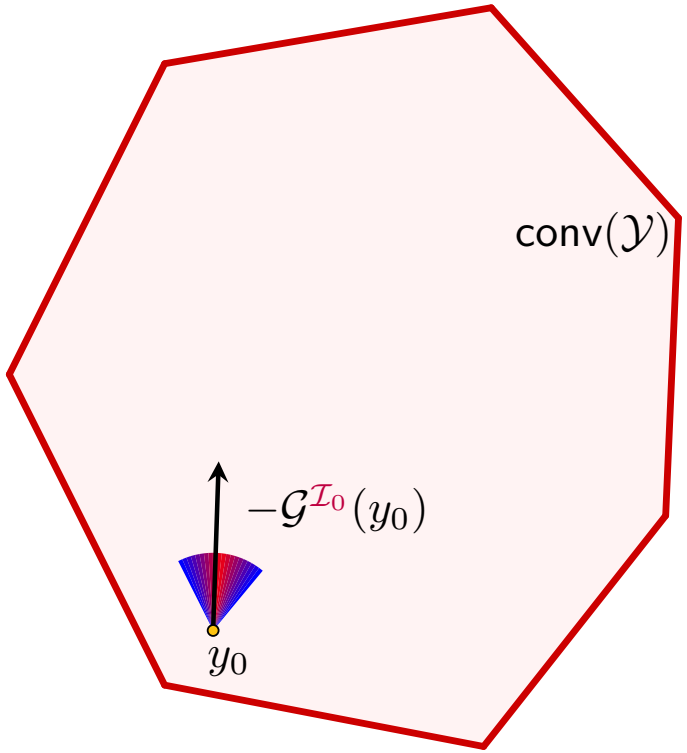
Take $\mathcal{I}_t \subseteq [\![N]\!]$ random

$\widehat{\nabla} F(y_t) \leftarrow \beta_t \mathcal{G}^{\mathcal{I}_t}(y_t) + (1 - \beta_t)\widehat{\nabla} F(y_{t-1})$

$v_t \leftarrow \operatorname{argmin}_{v \in \operatorname{conv}(\mathcal{Y})} \mathrm{LMO}\left(\widehat{\nabla} F(y_t), v\right)$

$y_{t+1} \leftarrow z_t + \gamma_t (v_t - y_t)$

end for



Stochastic Frank-Wolfe algorithm

Algorithm 1: *Momentum stochastic Frank-Wolfe* **[Braun et al., 2022]**

Input: Starting point $y_0 \in \mathcal{Y}$, step sizes γ_t and momentum terms $\beta_t \in [0, 1]$.

for $t = 0$ to $\ldots$ **do**

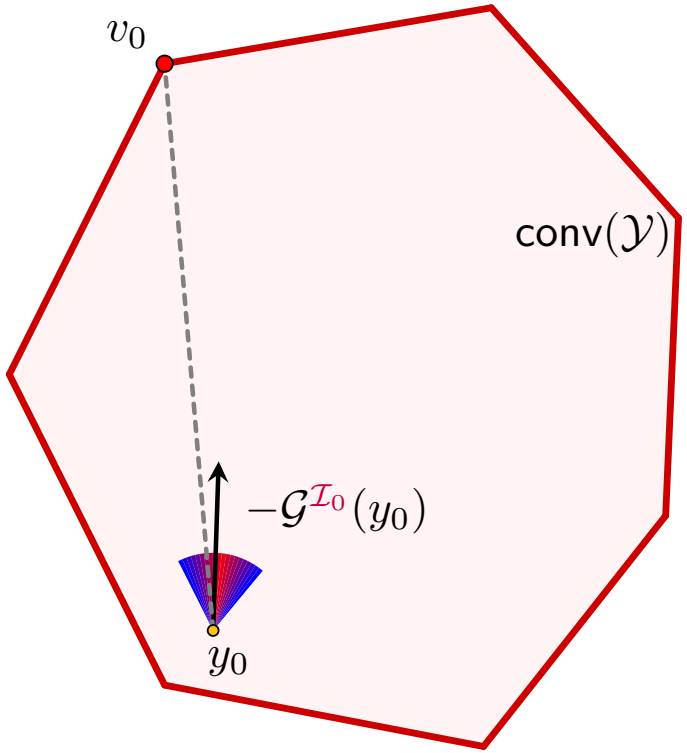
Take $\mathcal{I}_t \subseteq [\![N]\!]$ random

$\widehat{\nabla} F(y_t) \leftarrow \beta_t \mathcal{G}^{\mathcal{I}_t}(y_t) + (1 - \beta_t)\widehat{\nabla} F(y_{t-1})$

$v_t \leftarrow \operatorname{argmin}_{v \in \operatorname{conv}(\mathcal{Y})} \mathrm{LMO}\left(\widehat{\nabla} F(y_t), v\right)$

$y_{t+1} \leftarrow z_t + \gamma_t (v_t - y_t)$

end for



Stochastic Frank-Wolfe algorithm

Algorithm 1: *Momentum stochastic Frank-Wolfe* **[Braun et al., 2022]**

Input: Starting point $y_0 \in \mathcal{Y}$, step sizes γ_t and momentum terms $\beta_t \in [0, 1]$.

for $t = 0$ to $\ldots$ **do**

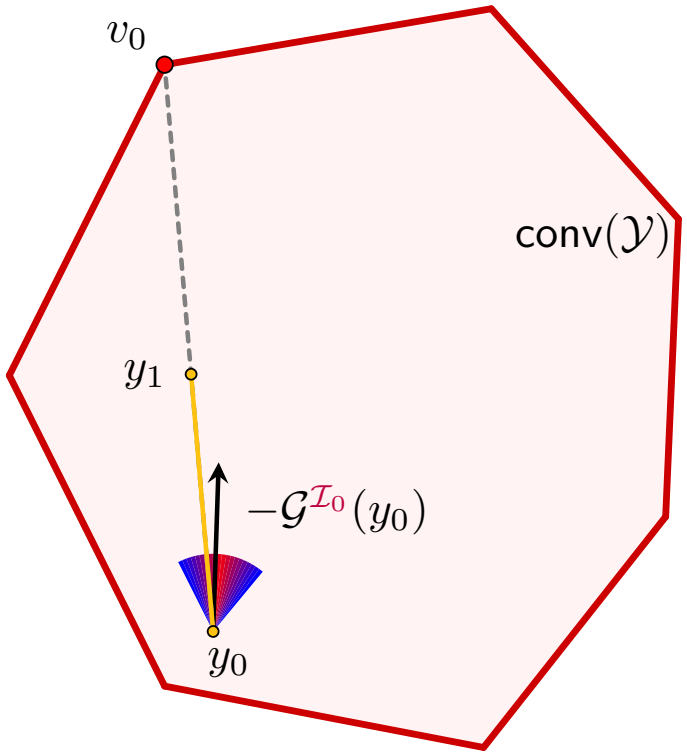
Take $\mathcal{I}_t \subseteq [\![N]\!]$ random

$\widehat{\nabla} F(y_t) \leftarrow \beta_t \mathcal{G}^{\mathcal{I}_t}(y_t) + (1 - \beta_t)\widehat{\nabla} F(y_{t-1})$

$v_t \leftarrow \operatorname{argmin}_{v \in \operatorname{conv}(\mathcal{Y})} \mathrm{LMO}\left(\widehat{\nabla} F(y_t), v\right)$

$y_{t+1} \leftarrow z_t + \gamma_t (v_t - y_t)$

end for



Stochastic Frank-Wolfe algorithm

Algorithm 1: *Momentum stochastic Frank-Wolfe* **[Braun et al., 2022]**

Input: Starting point $y_0 \in \mathcal{Y}$, step sizes γ_t and momentum terms $\beta_t \in [0, 1]$.

for $t = 0$ to $\ldots$ **do**

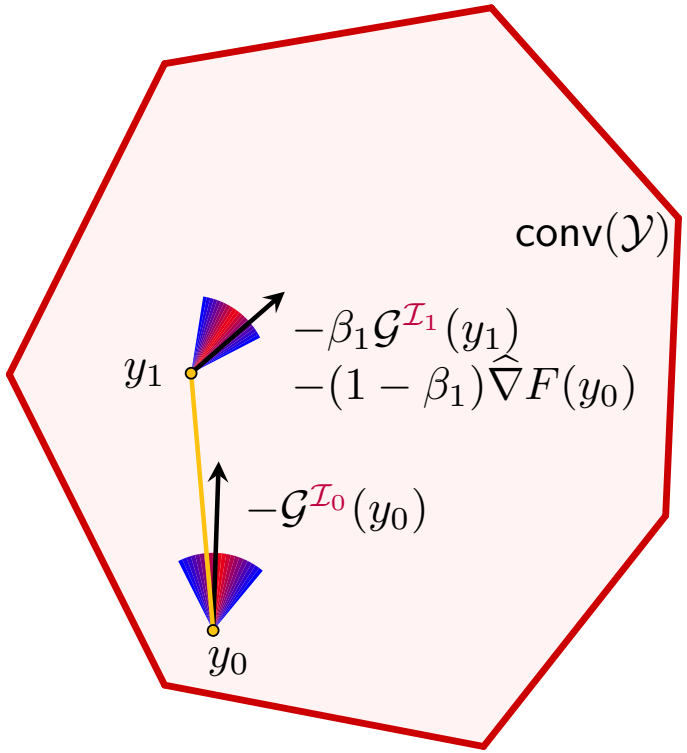
Take $\mathcal{I}_t \subseteq [\![N]\!]$ random

$\widehat{\nabla} F(y_t) \leftarrow \beta_t \mathcal{G}^{\mathcal{I}_t}(y_t) + (1 - \beta_t)\widehat{\nabla} F(y_{t-1})$

$v_t \leftarrow \operatorname{argmin}_{v \in \operatorname{conv}(\mathcal{Y})} \mathrm{LMO}\left(\widehat{\nabla} F(y_t), v\right)$

$y_{t+1} \leftarrow z_t + \gamma_t (v_t - y_t)$

end for



Stochastic Frank-Wolfe algorithm

Algorithm 1: *Momentum stochastic Frank-Wolfe* **[*Braun et al., 2022*]**

Input: Starting point $y_0 \in \mathcal{Y}$, step sizes γ_t and momentum terms $\beta_t \in [0, 1]$.

for $t = 0$ to $\ldots$ **do**

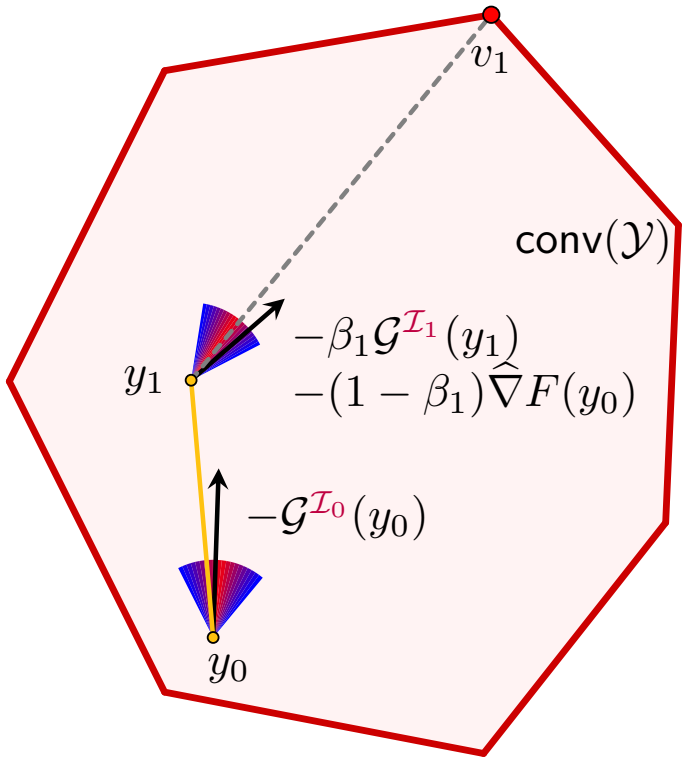
Take $\mathcal{I}_t \subseteq [\![N]\!]$ random

$\widehat{\nabla} F(y_t) \leftarrow \beta_t \mathcal{G}^{\mathcal{I}_t}(y_t) + (1 - \beta_t)\widehat{\nabla} F(y_{t-1})$

$v_t \leftarrow \operatorname{argmin}_{v \in \operatorname{conv}(\mathcal{Y})} \mathrm{LMO}\left(\widehat{\nabla} F(y_t), v\right)$

$y_{t+1} \leftarrow z_t + \gamma_t (v_t - y_t)$

end for



Stochastic Frank-Wolfe algorithm

Algorithm 1: *Momentum stochastic Frank-Wolfe* **[*Braun et al., 2022*]**

Input: Starting point $y_0 \in \mathcal{Y}$, step sizes γ_t and momentum terms $\beta_t \in [0, 1]$.

for $t = 0$ to $\ldots$ **do**

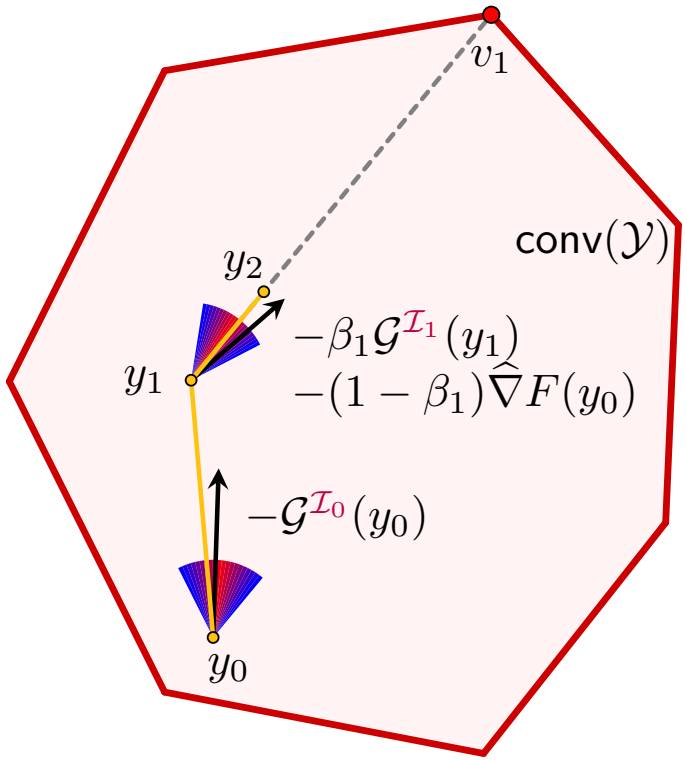
Take $\mathcal{I}_t \subseteq [\![N]\!]$ random

$\widehat{\nabla} F(y_t) \leftarrow \beta_t \mathcal{G}^{\mathcal{I}_t}(y_t) + (1 - \beta_t)\widehat{\nabla} F(y_{t-1})$

$v_t \leftarrow \operatorname{argmin}_{v \in \mathsf{conv}(\mathcal{Y})} \mathsf{LMO}\left(\widehat{\nabla} F(y_t), v\right)$

$y_{t+1} \leftarrow z_t + \gamma_t (v_t - y_t)$

end for



Stochastic Frank-Wolfe algorithm

Algorithm 1: *Momentum stochastic Frank-Wolfe* [***Braun et al., 2022***]

Input: Starting point $y_0 \in \mathcal{Y}$, step sizes γ_t and momentum terms $\beta_t \in [0, 1]$.

for $t = 0$ to $\ldots$ **do**

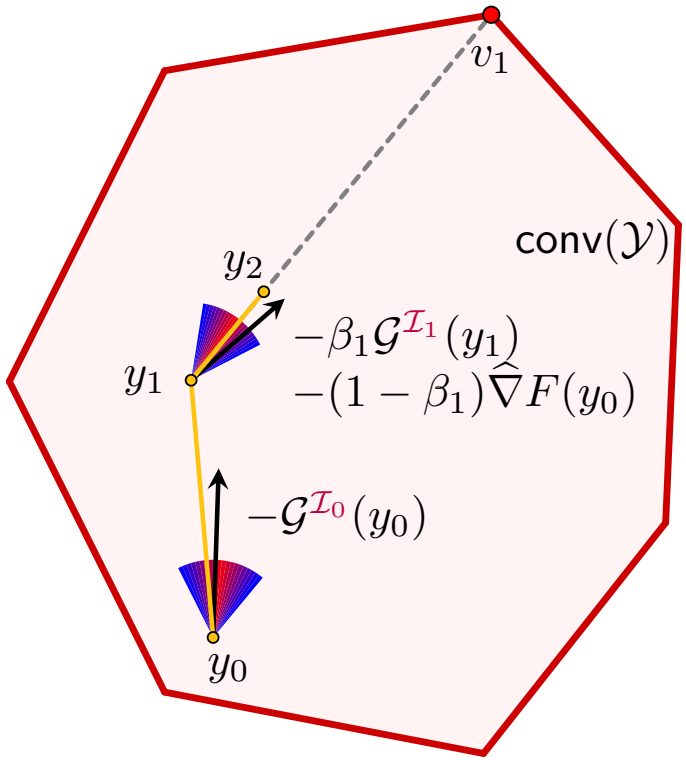
Take $\mathcal{I}_t \subseteq [\![N]\!]$ random

$\widehat{\nabla} F(y_t) \leftarrow \beta_t \mathcal{G}^{\mathcal{I}_t}(y_t) + (1 - \beta_t)\widehat{\nabla} F(y_{t-1})$

$v_t \leftarrow \operatorname{argmin}_{v \in \mathrm{conv}(\mathcal{Y})} \mathrm{LMO}\left(\widehat{\nabla} F(y_t), v\right)$

$y_{t+1} \leftarrow z_t + \gamma_t (v_t - y_t)$

end for



Remarks

- LMO offers flexible modelling: add constraints, cuts, ... [***Besançon et al., 2022***]

Stochastic Frank-Wolfe algorithm

Algorithm 1: *Momentum stochastic Frank-Wolfe* **[*Braun et al., 2022*]**

Input: Starting point $y_0 \in \mathcal{Y}$, step sizes γ_t and momentum terms $\beta_t \in [0, 1]$.

for $t = 0$ to $\ldots$ **do**

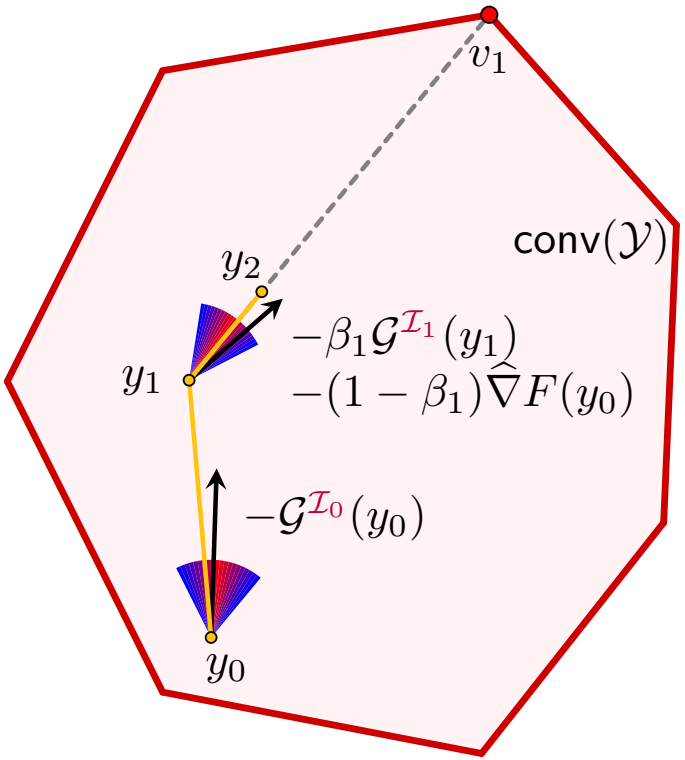
Take $\mathcal{I}_t \subseteq [\![N]\!]$ random

$\widehat{\nabla} F(y_t) \leftarrow \beta_t \mathcal{G}^{\mathcal{I}_t}(y_t) + (1 - \beta_t)\widehat{\nabla} F(y_{t-1})$

$v_t \leftarrow \operatorname{argmin}_{v \in \operatorname{conv}(\mathcal{Y})} \mathrm{LMO}\left(\widehat{\nabla} F(y_t), v\right)$

$y_{t+1} \leftarrow z_t + \gamma_t (v_t - y_t)$

end for



Remarks

- LMO offers flexible modelling: add constraints, cuts, ... **[*Besançon et al., 2022*]**
- Quadratic spanning tree: LMO = Kruskal$\rightarrow$ solved efficiently!

Experimental setup

	ERM	Robust
Objective	$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \frac{1}{N}\sum_{k=1}^{N} z^\top \widehat{\xi}_k z$	$\underset{z\in\mathcal{Z},\lambda\geq 0}{\text{minimize}} \quad F(z,\lambda)$

Experimental setup

	ERM	Robust
Objective	$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \frac{1}{N}\sum_{k=1}^{N} z^\top \widehat{\xi}_k z$	$\underset{z\in\mathcal{Z},\lambda\geq 0}{\text{minimize}} \quad F(z,\lambda)$
Gradient	$\frac{1}{N}\sum_{k=1}^{N}\left(\widehat{\xi}_k + \widehat{\xi}_k{}^\top\right) z$	$\mathcal{G}^{\mathcal{I}}(z,\lambda)$

Experimental setup

	ERM	Robust
Objective	$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \frac{1}{N}\sum_{k=1}^{N} z^\top \widehat{\xi}_k z$	$\underset{z\in\mathcal{Z},\lambda\geq 0}{\text{minimize}} \quad F(z,\lambda)$
Gradient	$\frac{1}{N}\sum_{k=1}^{N}\left(\widehat{\xi}_k + \widehat{\xi}_k{}^\top\right) z$	$\mathcal{G}^{\mathcal{I}}(z,\lambda)$
Solution	z_{erm}	z_{robust}

Experimental setup

	ERM	Robust
Objective	$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \frac{1}{N}\sum_{k=1}^{N} z^\top \widehat{\xi}_k z$	$\underset{z\in\mathcal{Z},\lambda\geq 0}{\text{minimize}} \quad F(z,\lambda)$
Gradient	$\frac{1}{N}\sum_{k=1}^{N}\left(\widehat{\xi}_k + \widehat{\xi}_k^{\top}\right) z$	$\mathcal{G}^{\mathcal{I}}(z,\lambda)$
Solution	z_{erm}	z_{robust}
Train Loss	Identical (and easy!) LMO	Identical (and easy!) LMO

Experimental setup

	ERM	Robust
Objective	$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \frac{1}{N}\sum_{k=1}^{N} z^\top \widehat{\xi}_k z$	$\underset{z\in\mathcal{Z},\lambda\geq 0}{\text{minimize}} \quad F(z,\lambda)$
Gradient	$\frac{1}{N}\sum_{k=1}^{N}\left(\widehat{\xi}_k + \widehat{\xi}_k^\top\right) z$	$\mathcal{G}^{\mathcal{I}}(z,\lambda)$
Solution	z_{erm}	z_{robust}
Train Loss	Identical (and easy!) LMO	Identical (and easy!) LMO

Build $\tilde{\mathbf{P}}$ a shifted distribution, *e.g.* $\tilde{\mathbf{P}} = \mathbf{P} + \text{noise}$

Sample Test Set $= \left\{\tilde{\xi} \sim \tilde{\mathbf{P}}\right\}$

Experimental setup

	ERM	Robust
Objective	$\underset{z\in\mathcal{Z}}{\text{minimize}} \quad \frac{1}{N}\sum_{k=1}^{N} z^\top \widehat{\xi}_k z$	$\underset{z\in\mathcal{Z},\lambda\geq 0}{\text{minimize}} \quad F(z,\lambda)$
Gradient	$\frac{1}{N}\sum_{k=1}^{N}\left(\widehat{\xi}_k + \widehat{\xi}_k^\top\right) z$	$\mathcal{G}^{\mathcal{I}}(z,\lambda)$
Solution	z_{erm}	z_{robust}
Train Loss	Identical (and easy!) LMO	Identical (and easy!) LMO

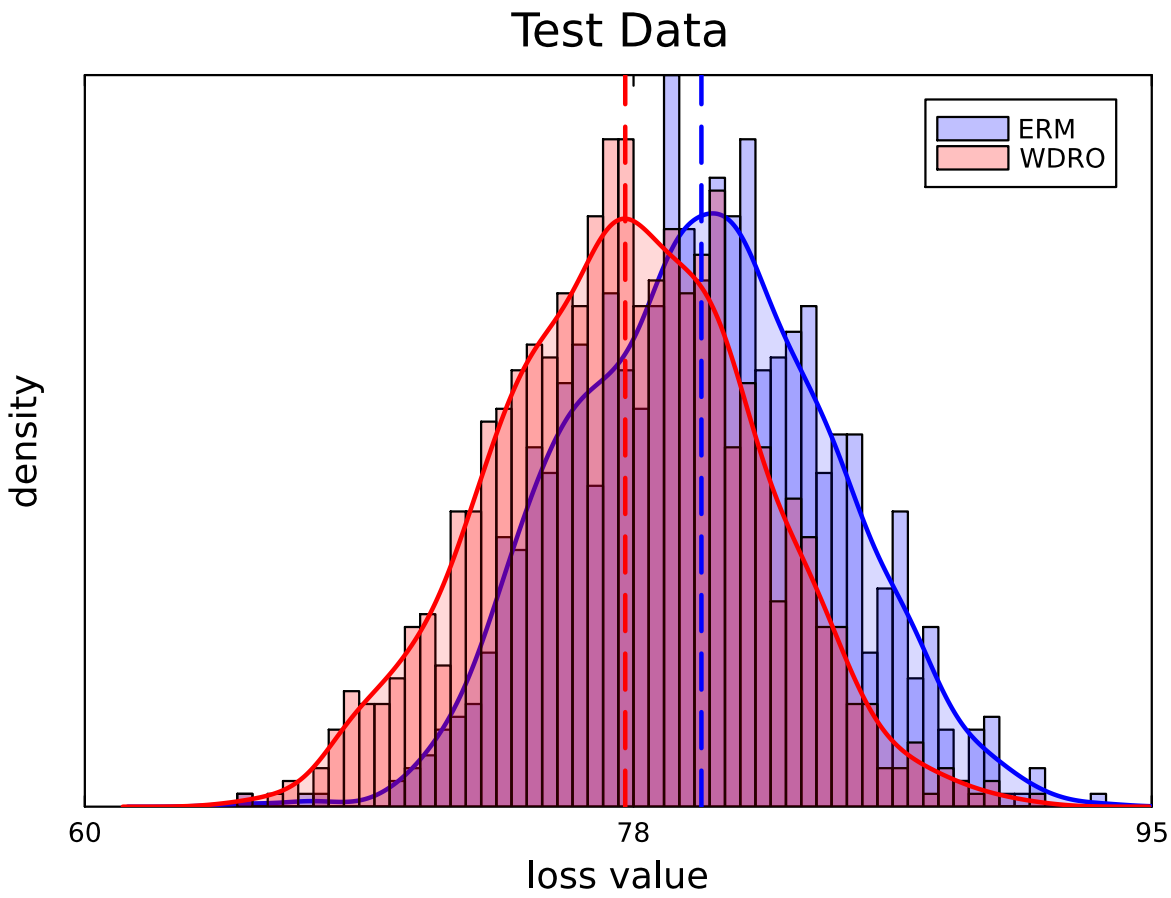
Build $\tilde{\mathbf{P}}$ a shifted distribution, *e.g.* $\tilde{\mathbf{P}} = \mathbf{P} + \text{noise}$

Sample Test Set $= \left\{\tilde{\xi} \sim \tilde{\mathbf{P}}\right\}$

Test Loss	$\left\{ z_{\text{erm}}{}^\top \tilde{\xi} z_{\text{erm}} \;\middle|\; \tilde{\xi} \in \text{Test Set} \right\}$	$\left\{ z_{\text{robust}}{}^\top \tilde{\xi} z_{\text{robust}} \;\middle|\; \tilde{\xi} \in \text{Test Set} \right\}$

Experiments on Quadratic Minimum Spanning Tree

Evaluating losses on $^{\star}{}_{\star}{}^{\star}{}_{\star}NEW^{\star}{}_{\star}{}^{\star}{}_{\star}$ data!



Conclusion and perspectives

- Entropic regularization of WDRO:
- Stochastic estimator of the gradient:
- Momentum stochastic Frank-Wolfe algorithm:

Conclusion and perspectives

- Entropic regularization of WDRO:
- Stochastic estimator of the gradient:
- Momentum stochastic Frank-Wolfe algorithm:

} for generic convex differentiable loss function f

Conclusion and perspectives

- Entropic regularization of WDRO:
- Stochastic estimator of the gradient:
- Momentum stochastic Frank-Wolfe algorithm:

} for generic convex differentiable loss function f

- Robust solutions to distributional shift
  - ➞ Illustration on the Quadratic Minimum Spanning Tree Problem
  - ➞ Illustration on the Traffic Assignment Problem (in the paper)

Conclusion and perspectives

- Entropic regularization of WDRO:
- Stochastic estimator of the gradient:
- Momentum stochastic Frank-Wolfe algorithm:

} for generic convex differentiable loss function f

- Robust solutions to distributional shift
  - ➞ Illustration on the Quadratic Minimum Spanning Tree Problem
  - ➞ Illustration on the Traffic Assignment Problem (in the paper)



- Adaptation to other problems:
  - ➞ Robustify your owns!

References I


Assad, A. and Xu, W. (1992).
The quadratic minimum spanning tree problem.
Naval Research Logistics.


Azizian, W., Iutzeler, F., and Malick, J. (2023).
Regularization for Wasserstein distributionally robust optimization.
ESAIM: Control, Optimisation and Calculus of Variations.


Besançon, M., Carderera, A., and Pokutta, S. (2022).
`FrankWolfe.jl`: A high-performance and flexible toolbox for Frank–Wolfe algorithms and conditional gradients.
INFORMS Journal on Computing.


Braun, G., Carderera, A., Combettes, C., Hassani, H., Karbasi, A., Mokhtari, A., and Pokutta, S. (2022).
Conditional gradient methods.
arXiv preprint arXiv:2211.14103.


Chow, V. T. F., Cui, Z., and Long, D. Z. (2022).
Target-oriented distributionally robust optimization and its applications to surgery allocation.
INFORMS Journal on Computing.

References II


Esfahani, P. and Kuhn, D. (2018).
Data-driven distributionally robust optimization using the Wasserstein metric: Performance guarantees and tractable reformulations.
Mathematical Programming.


Ghosal, S. and Wiesemann, W. (2020).
The distributionally robust chance-constrained vehicle routing problem.
Operations Research.


Schrijver, A. (2002).
Combinatorial optimization.

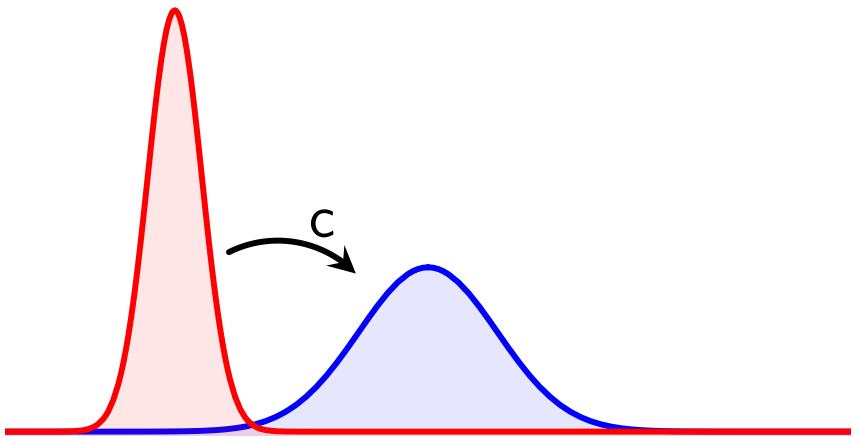

Sun, L., Xie, W., and Witten, T. (2023).
Distributionally robust fair transit resource allocation during a pandemic.
Transportation science.


Vincent, F., Azizian, W., Iutzeler, F., and Malick, J. (2024).
`skwdro`: A library for Wasserstein distributionally robust machine learning.

Wasserstein/Optimal Transport

Wasserstein distance:

$$W(Q_1, Q_2) = \inf_{\substack{\pi \in \mathbf{Proba}(\Xi \times \Xi) \\ [\pi]_1 = Q_1, [\pi]_2 = Q_2}} \int_{\Xi \times \Xi} c\left(\xi, \xi'\right) \mathrm{d}\pi\left(\xi, \xi'\right)$$



"How much effort needed to transport Q_1 to Q_2?

WDRO parameters: $\lambda_{\max}$

$$\exp\left(f(z,\zeta) - \lambda c(\widehat{\xi}_k,\zeta)/\varepsilon\right)$$

Heuristic: $\lambda_{\max}$ such that $\lambda_{\max} = \mathcal{O}\left(\mathbf{E}_z\left[\operatorname{diam}\left(f(z,\Xi)\right)\right]/\mathbf{E}_{k,\zeta}\left[c(\widehat{\xi}_k,\zeta)\right]\right)$

Require: training samples $\left\{\widehat{\xi}_1,\ldots,\widehat{\xi}_N\right\}$ of the learning problem

for $k \in [\![N]\!]$ **do**

Sample $\zeta_1^{(k)},\ldots,\zeta_S^{(k)} \sim \mathcal{N}\left(\widehat{\xi}_k,\sigma^2\mathsf{I}\right)$

$z_k \leftarrow \mathrm{LMO}\left(\zeta_k\right)$ with $\zeta_k \sim \mathcal{N}\left(\widehat{\xi}_k,\sigma^2\mathsf{I}\right)$

end for

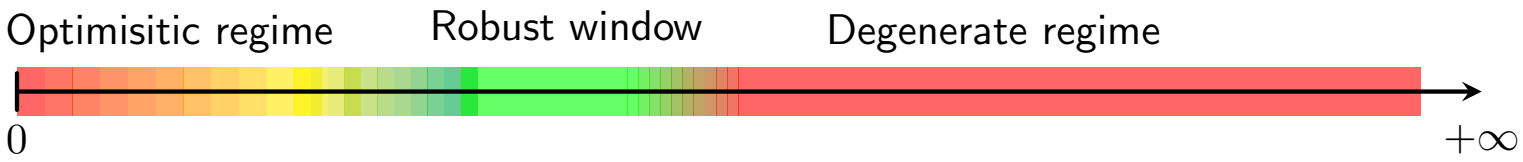
$$\widetilde{c} \leftarrow \frac{1}{NS}\sum_{k=1}^{N}\sum_{s=1}^{S} c(\widehat{\xi}_k,\zeta_s^{(k)})$$

$$\widetilde{\Delta_f} \leftarrow \frac{1}{N}\sum_{k=1}^{N}\left(\max_{s\in[\![S]\!]}\left\{f(z_k,\zeta_s^{(k)})\right\} - \min_{s'\in[\![S]\!]}\left\{f(z_k,\zeta_s^{(k)})\right\}\right)$$

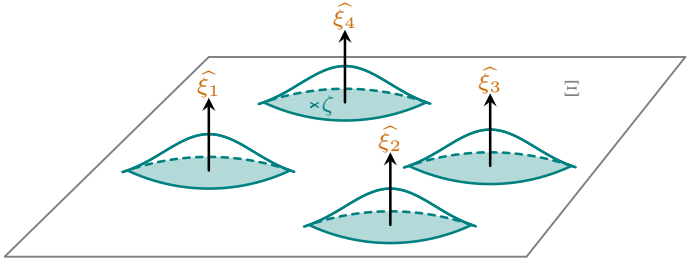
return $\lambda_{\max} \overset{\text{def}}{=} \widetilde{\Delta_f}/2\widetilde{c}$

WDRO parameters: $\rho, \sigma, \varepsilon$

- ρ changes behavior of the model:



- σ controls scenario sampling:



  - Small σ: small variety in the sampling process $\rightarrow$ closer to available data
  - Big σ: higher chances that

$$\min_{z \in \mathcal{Z}} f(z, \zeta)$$

    becomes infeasible.

- ε controls the approximation
  - big ε: "blind model": converges fast **but** untrustworthy solutions
  - small ε: "sharp model": reliable solutions **but** high computational cost

Algorithm convergence

Theorem

With step-sizes $\alpha_t = 2/(t+7)$, momentum terms $\beta_t = 4/(t+8)^{2/3}$, mini-batch size b, our Stochastic Frank-Wolfe algorithm gives iterates $(z_t, \lambda_t)_{t\geq 0} \in \text{conv}\left(\mathcal{Z} \times [0, \lambda_{\max}]\right)$ that verify, for all $t \geq 0$:

$$\mathbf{E}\left[F(z_t, \lambda_t)\right] - F(z^\star, \lambda^\star) \leq \frac{Q}{\sqrt[3]{t+9}}$$

with

$$Q \leq \frac{LD}{2} + 2D \max\left\{3\|\nabla F(z_0, \lambda_0)\|, \sqrt{\frac{64C^2}{bS} \cdot \exp\left(\frac{4B}{\varepsilon}\right)\left(1 + \frac{8B\text{Lip}(c)\sigma^2}{\varepsilon(\rho - \text{Lip}(c)\sigma^2 d)}\right)^{d/2} + 2L^2D^2}\right\}$$

$B \stackrel{\text{def}}{=} \|f(z,\cdot)\|_\infty$, $C \stackrel{\text{def}}{=} \|\nabla_z f(z,\cdot)\|_\infty$, $D \stackrel{\text{def}}{=} \text{Diam}(\mathcal{Z} \times [0, \lambda_{\max}])$ and $L > 0$ such that F is L-smooth.