Manifold Sampling for Nonconvex Optimization of Piecewise Linear Compositions

And application for robust learning of trimmed estimators

Kamil Khan, Jeffrey Larson, Matt Menickelly, Stefan M. Wild

ICML Nonconvex optimization workshop 2018

Optimization Problem

We are going to solve the problem:

$$f(x) = \psi(x) + h(F(x)) \to \min_x . \tag{1}$$

Where:

1. $\psi(\cdot) : \mathbb{R}^n \to \mathbb{R}$ is a smooth function with known derivative,
2. $F(\cdot) : \mathbb{R}^n \to \mathbb{R}^p$ is a smooth function with known derivative,
3. $h(\cdot) : \mathbb{R}^p \to \mathbb{R}$ is a continuous piecewise linear function.

Continuous Piecewise Linear Functions

$h(x)$ is a continuous picewise linear if:

- $h(x)$ is continuous,
- $\mathfrak{h} = \{h_i : i = 1, \ldots, \hat{m}\}$ is a set of affine functions,
- $h(z) \in \{\tilde{h}(z) : \tilde{h} \in \mathfrak{h}\}$.

We will use some additional notations:

- $\mathcal{S}_i = \{y : h(y) = h_i(y)\}$ and $\tilde{\mathcal{S}}_i = cl(int(\mathcal{S}_i))$,
- $I_h^e(z) = \{i : z \in \tilde{\mathcal{S}}_i\}$ is a set of active indices,
- $\mathbb{H}(z) = \{h_i : i \in I_h^e(z)\}$ is a set of active functions.

Continuous Piecewise Linear Functions

Let $x \in \mathbb{R}^3$ lets take a look at ℓ_1 norm:

$$f(x) = |x_1| + |x_2| + |x_3| = \begin{cases} x_1 + x_2 + x_3 & \text{if } x_1, x_2, x_3 \geq 0, \\ x_1 + x_2 - x_3 & \text{if } x_1, x_2 \geq 0, x_3 \leq 0, \\ x_1 - x_2 + x_3 & \text{if } x_1, x_3 \geq 0, x_2 \leq 0, \\ x_1 - x_2 - x_3 & \text{if } x_1 \geq 0, x_2, x_3 \leq 0, \\ \dots \end{cases} \tag{2}$$

- If $x \in \mathbb{R}^3_{++}$ $I_h^e(x)$ contains only 1 index.
- if $x : x_3 = 0$ then $I_h^e(x)$ contains 2 indices.
- if $x : x_3 = 0, x_2 = 0$ then $I_h^e(x)$ contains 4 indices
- if $x_1, x_2, x_3 = 0$ then $I_h^e(x)$ contains all 8 indices

Clarke Subdifferential

In general function $h(\cdot)$ is very bad:

- It is non-differentiable. Gradient doesn't exists.
- It is non-convex. Subgradient doesn't exist.

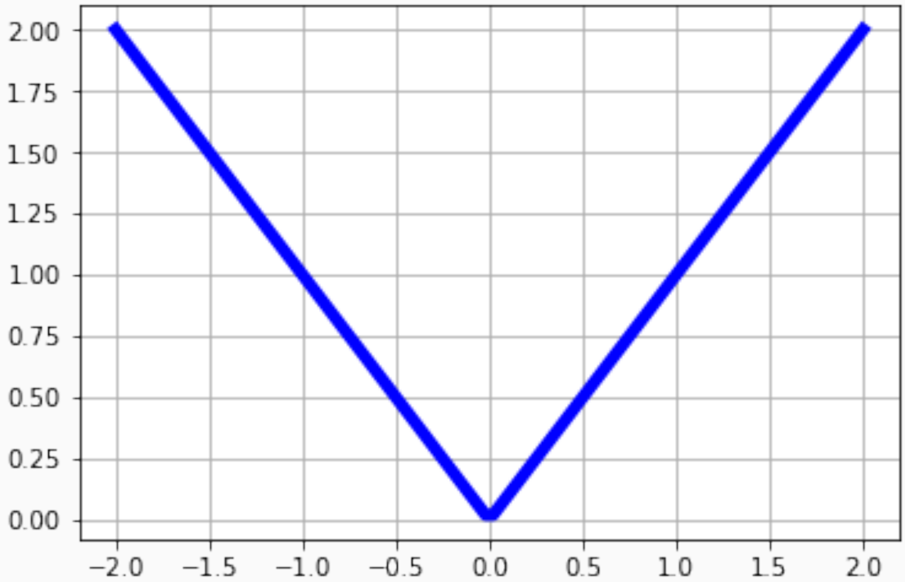
Clarke Subdifferential

We will use generalization of subdifferential:

$$\partial_B f(x) = \left\{ \lim_{y^j \to x} \nabla f(y^j) : y^j \text{ Differentiable} \right\}, \tag{3}$$

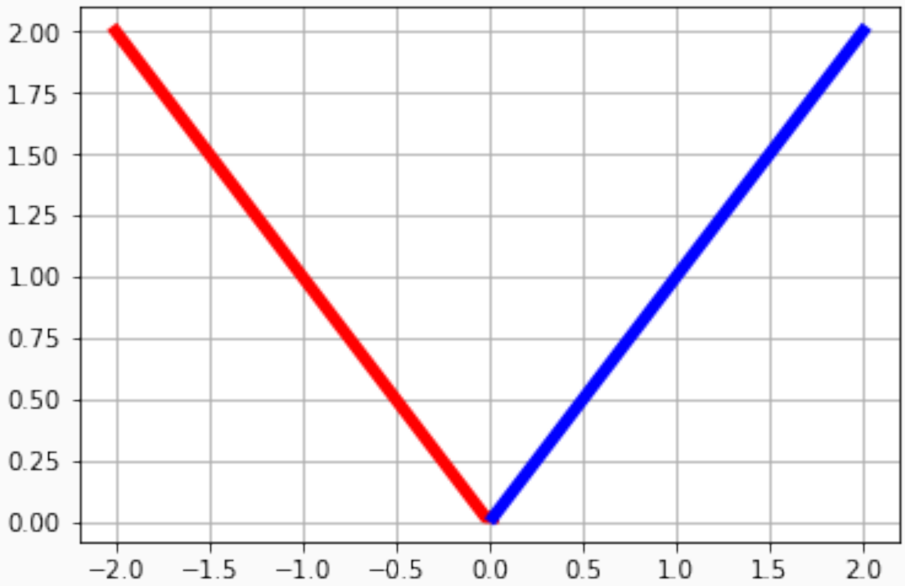
$$\partial_C f(x) = conv(\partial_B f(x)). \tag{4}$$

Clarke Subdifferential



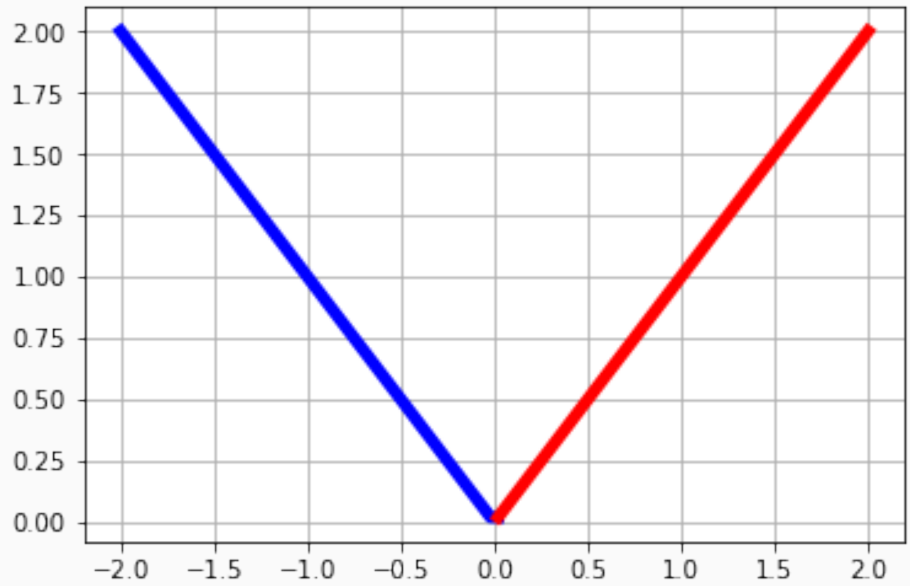
$$f(x) = |x| = \begin{cases} x & \text{If } x \geq 0, \\ -x & \text{If } x \leq 0. \end{cases} \tag{5}$$

Clarke Subdifferential



$$\lim_{y^i < 0 \to 0} \nabla f(y^i) = -1. \tag{6}$$

Clarke Subdifferential



$$\lim_{y^i > 0 \to 0} \nabla f(y^i) = 1. \tag{7}$$

Clarke Subdifferential

So we have:

- $\lim_{y^i < 0 \to 0} \nabla f(y^i) = -1$,
- $\lim_{y^i > 0 \to 0} \nabla f(y^i) = 1$,
- $\partial_B f(0) = \{-1, 1\}$,
- $\partial_C f(0) = conv(\partial_B f(0)) = conv(\{-1, 1\}) = [-1, 1]$.

Clarke Subdifferential

Take a look at our target function at point x

$$f(x) = \psi(x) + h(F(x)) \tag{8}$$

We have:

- an active set $I_h^e(x)$ of size m,
- set of active functions $\mathbb{H}(x) = \{a_i^T x + b_i, i = 1, \dots, m\}$,
- m different gradients in form $\nabla f(x) = \nabla\psi(x) + \nabla F(x) a_i$,
- $\partial_C f(x) = conv(\nabla\psi(x) + \nabla F(x) a_i | i \in I_h^e)$.

$F(x)$ Approximation

Suppose we don't have an access to exact function $F(\cdot)$. But instead we have an element-wise approximations m^{F_i}:

- $|F_i(x+s) - m^{F_i}(x+s)| \leq \kappa_{i,ef}\Delta^2 \quad \forall s \in \mathcal{B}(0,\Delta)$,
- $\|\nabla F_i(x+s) - \nabla m^{F_i}(x+s)\| \leq \kappa_{i,eg}\Delta \quad \forall s \in \mathcal{B}(0,\Delta)$,
- $\|\nabla^2 m^{F_i}(x)\| \leq \kappa_{i,mH}$.

Master Model and Generators $\mathfrak{G}$

Suppose we have a point x^k. For this point, we form a generator $\mathfrak{G}^k$ based on active and potentially active indices.

In order to build a generator $\mathfrak{G}^k$ we will use approximations m^{F_i}.

Generator $\mathfrak{G}^k$ contains elements in form $\nabla\psi(x^k) + \nabla M(x^k)a_i$.

Elements of generator $\mathfrak{G}^k$ form a matrix G^k.

Master Model and Generators $\mathfrak{G}$

At each step k we want:

$$\{\nabla\psi(x^k) + \nabla M(x^k)a_i : i \in I_h^e(F(x^k))\} \subseteq \mathfrak{G}^k, \tag{9}$$

$$\mathfrak{G}^k \subseteq \{\nabla\psi(x^k) + \nabla M(x^k)a_i | y \in \mathcal{B}(x^k, \Delta_k), i \in I_h^e(F(y))\}. \tag{10}$$

It is important for a good approximation of $\partial_C f(x)$.

In practive all we can do is build this sets using sampling:

- $\{\nabla\psi(x^k) + \nabla M(x^k)a_i : i \in I_h^e(F(x^k))\} \subseteq \mathfrak{G}^k$,
- $\mathfrak{G}^k \subseteq \{\nabla\psi(x^k) + \nabla M(x^k)a_i | y \in Y, i \in I_h^e(F(y))\}$ for some $Y \subset \mathcal{B}(x^k, \Delta_k)$

Master Model and Generators $\mathfrak{G}$

We want to find:

$$g^k = proj(0, conv(\mathfrak{G}^k)). \tag{11}$$

To find this projection we will solve a problem:

$$\begin{cases} \lambda^T(G^k)^T G^k \lambda \to \min_\lambda \\ , e^T\lambda = 1, \lambda \geq 0. \end{cases} \tag{12}$$

Finally we have:

$$g_k = G^k \lambda^*. \tag{13}$$

Master Model and Generators $\mathfrak{G}$

We want to build a master model m_k^f such that $\nabla m_k^f = g_k$:

$$m_k^f = \psi(x^k) + \sum_{i=1}^{p} w_i^k m^{F_i}(x) + \sum_{i=1}^{p} \lambda_i^* b_{j_i}. \tag{14}$$

Where $w^k = A^k \lambda^*$ and A is matrix formed from components a_i.

Sufficient Decrease Condition

On each step k we will use master model in trust region subproblem:

$$\begin{cases} m_k^f(x^k + s^k) \to \min\limits_{s^k}, \\ s \in \mathcal{B}(0, \Delta_k). \end{cases} \tag{15}$$

We don't need an exact solution. We want s^k satisfy:

$$\psi(x^k) - \psi(x^k + s^k) + \left(M(x^k) - M(x^k + s^k), a^{(k)} \right) \geq \frac{\kappa_d}{2} \min\{\Delta_k, \frac{\|g^k\|}{\kappa_{mH}}\}. \tag{16}$$

Where $a^{(k)}$ corresponds to function $h^{(k)}$:

$$h^{(k)}(F(x^k)) \leq h(F(x^k)) h^{(k)}(F(x^k + s^k)) \geq h(F(x^k + s^k)). \tag{17}$$

Sufficient Decrease Condition

For any a_q such that $\{\nabla\psi(x^k) + \nabla M(x^k)a_q) \in \mathfrak{G}^k$:

$$j^* = \max\left\{0, \left\lceil \log_{\kappa_d}\left(\frac{\|\nabla\psi(x^k) + \nabla M(x^k)a_q\|}{\kappa_{fH}\Delta_k}\right)\right\rceil\right\}. \tag{18}$$

And potential s^k:

$$\hat{s}^k = -\kappa_d^{j^*}\Delta_k \frac{\nabla\psi(x^k) + \nabla M(x^k)a_q}{\|\nabla\psi(x^k) + \nabla M(x^k)a_q\|.} \tag{19}$$

ρ_k Test

To control the quallity of approximation in trust region optimization we will use coefficient:

$$\rho_k = \frac{\psi(x^k) - \psi(x^k + s^k) + h^{(k)}(F(x^k)) - h^{(k)}(F(x^k + s^k))}{\psi(x^k) - \psi(x^k + s^k) + (M(x^k) - M(x^k + s^k), a^{(k)}).} \qquad (20)$$

If ρ_k is sufficiently large we accept point $x^k + s^k$.

Trimmed Estimation

Suppose we have datset $S = \{(x_1, y_1), \dots, (x_N, y_N)\}$, loss function $l(\hat{y}, y)$. And prediction function $F(x, w)$, where w is a parameters vector. We want to solve a problem:

$$\frac{1}{q} \sum_{i=1}^{q} l_{(i)} \left(F(x^{(i)}, w), y^{(i)} \right) \to \min_{w} . \tag{21}$$

In order to calculate this function we have to calculate loss for every element of S and select smallest q values.

Trimmed Estimation

Let's present it in the form of:

$$f(x) = \psi(x) + h(F(x)). \tag{22}$$

- $\psi(w) = 0$,
- $F(w) = [l(F(x_1, w), y_1, \dots, l(F(x_N, w), y_N)]$,
- $h(\cdot) = \{g^a(l(w)) : a \in I^{q,N}(l(w))\}$,
- $I^{q,N}(l(w)) = \{(i_1, \dots, i_q) : l_{i_j}(F(x^{i_j}, w), y^{i_j}) \leq l_{(q)}(F(x^{(q)}, w), y^{(q)})\}$,
- $g_i^a(l(w)) = \begin{cases} 1/q & i \in a, \\ 0 & \text{otherwise.} \end{cases}$

Trimmed Estimation

So we have:

$$f(w) = \frac{1}{q} \sum_{i=1}^{q} l_{(i)} \left(F(x^{(i)}, w), y^{(i)} \right) = h(F(w)). \tag{23}$$

We already can use manifold sampling algorithm to solve this problem. However authors proposed some modifications.

Direction Search

In order to make an optimization step we will solve the problem $\mathcal{M}$:

$$\begin{cases} \tau \to \min\limits_{\tau \in \mathbb{R}, s \in \mathbb{R}^n}, \\ \|s\|_2 \leq \Delta_k, \\ g^a(l(w^k)) - h(l(w^k)) + \nabla(g^a \circ l)(w^k)^T s \leq \tau \quad \forall a \in \mathfrak{G}^k. \end{cases} \tag{24}$$

Sampling

We can replace function $F(x)$ with approximation. This approximation could be stochastic.

At each step we consider a subsample $S \subseteq \{1, \ldots, N\}$. We replace function $l(w)$ projection of function which used only subsampled elements.

Also, the authors modified the acceptance criteria:

$$\rho_k = \frac{h(l_{S^k}(w^k)) - h(l_{S^k}(w^k + s^k))}{-\tau_k.} \tag{25}$$

Final Algorithm

Input: parameters $\gamma_{\text{dec}} \in (0, 1)$, $\gamma_{\text{inc}} > 1$, $\theta \in (0, 1)$, initial point $w^0 \in \mathbb{R}^n$, trust-region radius $\Delta_0 > 0$

repeat

  Sample $S^k \subset \{1, \ldots, N\}$

  $(\tau_k, s^k) \leftarrow \mathcal{M}(S^k, h, w^k, \Delta_k)$

  Sample $S^{k'} \subset \{1, \ldots, N\}$

  if $\rho_k \geq \theta$

    $w^{k+1} \leftarrow w^k + s^k$

    $\Delta_{k+1} \leftarrow \gamma_{\text{inc}} \Delta_k$

  else

    $w^{k+1} \leftarrow w^k$

    $\Delta_{k+1} \leftarrow \gamma_{\text{dec}} \Delta_k$

  end if

until budget exhausted