Universal Adversarial Perturbation starring Frank-Wolfe

Chiara Bigarella Student nr. 2004248

Silvia Poletti Student nr. 1239133

Gurjeet Singh Student nr. 2004251

Francesca Zen Student nr. 2010640

1. Introduction

2. Decentralized Stochastic Gradient Free Frank Wolfe

In this section we discuss the Stochastic Gradient Free FW in a decentralized setup. In this setting we have some devices, that we call workers, connected to a master node to read, write and exchange information.
In our case we have M workers to which is spread the data, they compute their local gradient, using the loss function and the parameters given from I-RDSA, and send them to the master node who uses the gradients collected to return the new perturbated data.
Given an input image $\boldsymbol{\delta}$, the algorithm initializes both the starting point and the gradient to a zero $\boldsymbol{d}$-vector and $\boldsymbol{M \times d}$ matrix, respectively. Then each worker computes its own gradient estimation based on the parameters given from I-RDSA in the following way:

$$\mathbf{g}_i = (1-\rho_t)\mathbf{g}_{i,t-1} + \rho_t \mathbf{g}_i(\delta_t, \mathbf{y}).$$

When all of the workers has come to the end, they send the results to the master node, which computes the average of the ones collected and get back the obtained value as new initial point $\boldsymbol{\delta_{t+1}}$:

$$\delta_{t+1} = (1-\gamma_t)\delta_t + \gamma_t \mathbf{v}_t$$

When all of the iterations are done, the algorithm returns the average of the gradient computation done by the master node.

3. Decentralized Variance-Reduced Stochastic Gradient Free Frank Wolfe

In this section we employ the SPIDER variance reduction technique, which is a built for dynamic tracking, while avoiding excessive querying to oracles and ultimately reducing query complexity.

Algorithm 1 Decentralized Stochastic Gradient Free FW

Require: Input image $\boldsymbol{\delta}$, labels $\boldsymbol{y}$, Loss Function $\boldsymbol{F(\delta; y)}$, number of queries $\boldsymbol{T}$, number of workers $\boldsymbol{M}$, image dimension $\boldsymbol{d}$, tolerance $\boldsymbol{\varepsilon}$, number of directions $\boldsymbol{m}$.

Ensure: $\boldsymbol{\delta^T}$

1: Initialize $\boldsymbol{\delta_0 = 0}$.
2: **for** $\boldsymbol{t = 0, \dots, T-1}$ **do**
3: Master node computes parameters required for the computation of the I-RDSA scheme:

$$(\rho_t, c_t)_{I-RDSA} = \Big(\frac{4}{(1+\frac{d}{m})^{1/3}(t+8)^{2/3}}, \frac{2\sqrt{m}}{d^{3/2}(t+8)^{1/3}} \Big)$$

4: For each worker $\boldsymbol{i}$ compute I-RDSA:
Sample $\{\mathbf{z}_{n,t}\}_{n=1}^m \sim \boldsymbol{N(0, 1_d)}$
$\mathbf{g}_i(\delta_t; \mathbf{y}) = \frac{1}{m}\sum_{n=1}^m \frac{F(\delta_t + c_t \mathbf{z}_{n,t}; \mathbf{y}) - F(\delta_t; \mathbf{y})}{c_t} \mathbf{z}_{n,t}$
5: Workers compute

$$\mathbf{g}_i = (1-\rho_t)\mathbf{g}_{i,t-1} + \rho_t \mathbf{g}_i(\delta_t, \mathbf{y}).$$

6: Push $\mathbf{g}_{i,t}$ to the master node.
7: Master node computes

$$\mathbf{g}_t = \frac{1}{M}\sum_{i=1}^M \mathbf{g}_{i,t}.$$

8: Master node computes $\mathbf{v}_t = -\boldsymbol{\varepsilon sign}(\mathbf{g}_t)$.
9: Master node computes $\boldsymbol{\delta_{t+1} = (1-\gamma_t)\delta_t + \gamma_t \mathbf{v}_t}$ and sends it to all nodes.
10: **end for**

4. Distributed Stochastic Gradient Free Frank Wolfe

In this section we discuss the Distributed Stochastic Grandient Free FW in a distributed setup. In this setting we have that the $\boldsymbol{M}$ workers do not have a central coordinator, instead they exchange information in a peer-to-peer manner. The internode comunication network used by the workers is modeled as an undirected simple connective graph $\boldsymbol{G = (V, E)}$, with $\boldsymbol{V = \{1, \dots, M\}}$ the set of nodes and $\boldsymbol{E}$

Algorithm 2 Decentralized Variance-Reduced Stochastic Gradient Free FW

Require: Input image δ, labels y, Loss Function $F(\delta; y)$, number of queries T, number of workers M, image dimension d, tolerance ε, number of images S_1, number of function components S_2, total number of function component n, period q, duality gap tolerance tol.

Ensure: Universal prturbation's history.

Initialize $\delta_0 = 0$.

for $t = 0, \dots, T-1$ **do**

  Each worker i computes:

  if $mod(t, q) = 0$ **then**

    Draw $S_1' = \frac{S_1 d}{M}$ samples for each dimension at each worker i and compute its local gradient $\mathbf{e}_k^T \mathbf{g}_i(\delta_t) = \frac{1}{n}\sum_{j=1}^{n} \frac{F_{i,j}(\mathbf{x}_t + \eta \mathbf{e}_k) - F_{i,j}(\mathbf{x}_t)}{\eta}$ along each canonical basis vector $\mathbf{e}_k$.

    Each worker updates $\mathbf{g}_{i,t} = \mathbf{g}_i(x)_t$.

  else

    Draw S_2 pairs of component functions and Gaussian random vectors $\{\mathbf{z}\}$ at each worker i and update

$$\mathbf{g}_i(x)_t = \frac{1}{|S_2|}\sum_{j \in S_2} \frac{F_{i,j}(\mathbf{x}_t + \eta \mathbf{e}_k) - F_{i,j}(\mathbf{x}_t)}{\eta}\mathbf{z} - \frac{F_{i,j}(\mathbf{x}_{t-1} + \eta \mathbf{e}_k) - F_{i,j}(\mathbf{x}_{t-1})}{\eta}\mathbf{z}$$

    Each worker updates

$$\mathbf{g}_{i,t} = \mathbf{g}_i(\mathbf{x}_t)\mathbf{g}_{i,t-1}.$$

  end if

  Each worker pushes $\mathbf{g}_{i,t}$ to the master node.

  Master node computes

$$\mathbf{g}_t = \frac{1}{M}\sum_{i=1}^{M} \mathbf{g}_{i,t}.$$

  Master node computes $\mathbf{v}_t = -\varepsilon sign(\mathbf{g}_t)$.

  Master node computes $\delta_{t+1} = (1 - \gamma_t)\delta_t + \gamma_t \mathbf{v}_t$ and sends it to all nodes.

end for

the set of comunication links. Each node comunicates and exchange information with its own neighbors, and given a node n we indicate with $\Omega_n = \{l \in V | (n, l) \in E\}$ its neighborhood. Node n has degree $d_n = |\Omega_n|$. Also, we use the $M \times M$ adjacency matrix $A = [A_{ij}]$ to describe the edges of the graph G: we have $A_{ij} = 1$ if $(i, j) \in E$, otherwise we get $A_{ij} = 0$. We define the diagonal matrix $D = (d_1 \dots d_M)$ in order to compute the graph Laplacian $L = D - A$. The Laplacian of L is define to be the matrix

$$\mathcal{L}(u, v) = \begin{cases} 1 & \text{if } u = v \text{ and } d_v \neq 0 \\ -\frac{1}{\sqrt{d_u d_v}} & \text{if } u \text{ and } v \text{ are adjacent} \\ 0 & \text{otherwise} \end{cases}$$

We can write $\mathcal{L} = D^{-1/2} L D^{-1/2}$, with the convention that $D^{-1}(v, v) = 0$ if $d_v = 0$. The Laplacian $\mathcal{L}$ is used to compute the weghted matrix $W = 1 - \mathcal{L}$.

At every time instat t in Algorithm 3, an agent i exchange its current iterate with its neighbors and average the iterates as follows:

$$\bar{\delta}_t^i \leftarrow \sum_{j=1}^{M} W_{ij} \delta_t^i.$$

After this, each worker i use I-RDSA to compute its local gradient estimation g_t^i, for $i = 1, \dots, M$, using $c_t = \frac{2\sqrt{m}}{d^{3/2}(t+8)^{1/3}}$. When all of them are calculated, they are collected as follow:

$$\bar{\mathbf{g}}_t^i \leftarrow \sum_{j=1}^{M} W_{ij} \mathbf{G}_t^j.$$

After the update, the algorithm computes the Frank-Wolfe step, calculating the new iterate δ_{t+1}^i using $\gamma_t = t^{-1/2}$.

A distributed FW algorithm requires two round of communication: one for the iterate and one for the gradient estimate.

The performance of Algorithm 3 depends on how well the averaged iterates and the averaged gradient estimates are tracked across the network.

5. Experiments

6. Conclusions

Algorithm 3 Distributed Stochastic Gradient Free FW

Require: Input image $\boldsymbol{\delta_0}$, labels $\boldsymbol{y}$, Loss Function $\boldsymbol{F(x; y)}$, number of queries $\boldsymbol{T}$, number of workers $\boldsymbol{M}$, image dimension $\boldsymbol{d}$, tolerance $\boldsymbol{\varepsilon}$, number of directions $\boldsymbol{m}$, adjacency matrix $\boldsymbol{A}$.

Ensure: Average perturbated images.

Initialize $\boldsymbol{\delta_0^i} = 0$ for $\boldsymbol{i = 1, \ldots, M}$.

for $\boldsymbol{t = 1, \ldots, T}$ **do**

Approximate the average iterate:

$$\boldsymbol{\bar{\delta}_t^i \leftarrow \sum_{j=1}^{M} W_{ij} \delta_t^i}$$

At each node $\boldsymbol{i}$, employ I-RDSA to estimate the gradient $\mathbf{g}_t^i = \sum_{j=1}^{d} \frac{F(\bar{\delta}_t^i + c_t \mathbf{e}_j) - F(\bar{\delta}_t^i)}{c_t} \mathbf{e}_j$.

Approximate the average gradient:

$$\mathbf{G}_t^i = \bar{\mathbf{g}}_{t-1}^i + \mathbf{g}_t^i - \mathbf{g}_{t-1}^i$$

$$\bar{\mathbf{g}}_t^i \leftarrow \sum_{j=1}^{M} W_{ij} \mathbf{G}_t^j$$

Each worker computes $\mathbf{v_t} = -\boldsymbol{\varepsilon sign}(\mathbf{g_t})$.

Frank-Wolfe Step: update

$$\boldsymbol{\delta_{t+1}^i \leftarrow (1 - \gamma_t) \bar{x}_t^i + \gamma_t v_t^i}$$

for all worker $\boldsymbol{i = 1, \ldots, M}$.

end for