Sparsity in Neural Networks

Berivan Isik
Electrical Engineering, Stanford University

December 30th 2022
1. What is sparsification for model compression and federated learning?

2. Model compression
   - An information-theoretic justification for model pruning

3. Compression for federated learning
   - Sparse random networks for communication-efficient federated learning
Sparsification
Federated Learning
AN INFORMATION-THEORETIC JUSTIFICATION FOR MODEL PRUNING
AISTATS'22

Joint work with
Tsachy Weissman (Stanford University)
Albert No (Hongik University)
Lossy Source Coding

\[ R(D) = \min_{p(\hat{u}|u): E[d(u,\hat{u})] \leq D} I(U; \hat{U}) \]
Distortion Metric

**Theorem:** Suppose $f(\cdot; \mathbf{w})$ is a fully-connected neural network model with $d$ layers and 1-Lipschitz activations, e.g., ReLU. Let $\hat{\mathbf{w}}$ be the reconstructed weight (after compression) where all layers are subject to compression. Then, we have the following bound on the output perturbation:

$$\sup_{|x|_1 \leq 1} \|f(x, \mathbf{w}) - f(x, \hat{\mathbf{w}})\|_1 \leq \left( \sum_{l=1}^{d} \frac{||w^{(l)} - \hat{w}^{(l)}||_1}{||w^{(l)}||_1} \right) \left( \prod_{k=1}^{d} ||w^{(k)}||_1 \right)$$

**Distortion function:**

$$d(\mathbf{u}, \hat{\mathbf{u}}) = \frac{1}{n} \sum_{i=1}^{n} |u_i - \hat{u}_i|,$$

where $u^{(l)} = \frac{w^{(l)}}{|w^{(l)}|_1}$. 
Density: Laplacian

Density vs. Norm. Weights

Stanford University
Rate-Distortion Function

For an i.i.d. Laplacian source sequence distributed according to $f(u; \lambda) = \frac{\lambda}{2} e^{-\lambda |u|}$, maximum compression for $l_1$ distortion $D$ is:

$$R(D) = \begin{cases} -\log(\lambda D), & 0 \leq D \leq \frac{1}{\lambda} \\ 0, & D > \frac{1}{\lambda} \end{cases}$$
\[ I(U; V) = H(U) - H(U|V) \]
\[ = \log\left(\frac{2e}{\lambda}\right) - H(U|V) \]
\[ = \log\left(\frac{2e}{\lambda}\right) - H(U - V|V) \]

\[ \geq \log\left(\frac{2e}{\lambda}\right) - H(U - V) \]

\[ \geq \log\left(\frac{2e}{\lambda}\right) - \log(2eE[|U - V|]) \]
\[ = \log\left(\frac{2e}{\lambda}\right) - \log(2eD) \]
\[ = -\log(\lambda D) \]

Maximum entropy theorem tells us that Laplace distribution with parameter \( \alpha \) has the maximum differential entropy \( h(f) \) over all probability densities \( f \) satisfying \( E[x] = 0 \) and \( E[|x|] = \frac{1}{\alpha} \) where \( x \sim f \).

\( U - V \) and \( V \) must be independent.

\( U - V \) must be Laplace distributed with parameter \( \frac{1}{D} \).

Stanford University
Conditions for Optimal Compression

1) $U - V$ and $V$ must be independent.
2) $U - V$ must be Laplace distributed with parameter $\frac{1}{D}$.

$$V \sim \text{LapMixer}(\lambda, D) \quad \text{+} \quad \quad Z \sim \text{Laplacean}(\frac{1}{D})$$

$$U \sim \text{Laplacean}(\lambda)$$

To achieve the maximum compression, we need a compression scheme with the following conditions:

1. Conditional probability distribution:
   $$f_{U|V}(u|v) = \frac{1}{2D} e^{-|u-v|/D}$$

2. Marginal probability distribution:
   $$f_V(v) = \lambda^2 D^2 \cdot \delta(v) + (1 - \lambda^2 D^2) \cdot \frac{\lambda}{2} e^{-\lambda|v|}$$
A NEW PRUNING ALGORITHM:
SUCCESSIVE REFINEMENT FOR PRUNING (SuRP)
Successive Refinement

$\mathbf{U}^n \xrightarrow{} m_1 \in \{0, 1\}^{nR_1} \xrightarrow{} \text{DECODER 1} \xrightarrow{} \hat{\mathbf{U}}_1^n$

$\mathbf{U}^n \xrightarrow{} m_2 \in \{0, 1\}^{nR_2} \xrightarrow{} \text{DECODER 2} \xrightarrow{} \hat{\mathbf{U}}_2^n$
First Attempt

- Consider successive refinement with $L$ decoders.

- Let $\lambda = \lambda_1 < \cdots < \lambda_L$ where $D_t = 1/\lambda_{t+1}$ is the target distortion at the $t$-th decoder.

- Set $U^{(1)} = u^{(n)}$.

- At the $t$-th iteration,
  - The encoder finds $V^{(t)}$ that minimizes $d(U^{(t)}, V^{(t)})$ from a codebook $C^{(t)}$.
  - The encoder computes the residual $U^{(t+1)} = U^{(t)} - V^{(t)}$.
  - The decoder reconstructs $\hat{U}^{(t)} = \sum_{\tau=1}^{t} V^{(\tau)}$.

- Complexity is $L \cdot 2^{nR/L}$, lower than the naïve random coding strategy with complexity $2^{nR}$.

Recall the optimal marginal distribution:

$$f_{V(t)}(v) = \frac{\lambda_t^2 D^2}{\lambda_{t+1}^2} \cdot \delta(v) + (1 - \lambda_t^2 D^2) \cdot \frac{\lambda_t}{2} e^{-\lambda_t |v|}$$
### Successive Refinement for Pruning (SuRP)

#### Iteration 0:

\[ U^{(0)} \]
\[ \hat{U}^{(0)} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \]

#### Iteration 1:

\[ U^{(0)} \xrightarrow{E^{m_1}} D \xrightarrow{} \hat{U}^{(1)} = V^{(1)} = \begin{bmatrix} 0 \\ 1/\lambda_0 \\ -1/\lambda_0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \]

#### Iteration 2:

\[ U^{(1)} = U^{(0)} - V^{(1)} \xrightarrow{E^{m_2}} D \xrightarrow{} \hat{U}^{(2)} = \hat{U}^{(1)} + V^{(2)} = \begin{bmatrix} 0 \\ 1/\lambda_0 + 1/\lambda_1 \\ -1/\lambda_0 \\ -1/\lambda_1 \\ 0 \\ 0 \end{bmatrix} \]

#### Iteration 3:

\[ U^{(2)} = U^{(1)} - V^{(2)} \xrightarrow{E^{m_3}} D \xrightarrow{} \hat{U}^{(3)} = \hat{U}^{(2)} + V^{(3)} = \begin{bmatrix} 0 \\ 1/\lambda_0 + 1/\lambda_1 \\ -1/\lambda_0 \\ -1/\lambda_1 - 1/\lambda_2 \\ 0 \\ 1/\lambda_2 \end{bmatrix} \]
Successive Refinement for Pruning (SuRP)

- Consider the successive refinement problem with L decoders.
- Each decoder corresponds to one iteration of SuRP (different from a pruning iteration).
- Set $U^{(1)} = u^n$. For iteration $1 \leq t \leq L - 1$:
  1. Find index $i$ and $j$ such that $U^{(t)}_i \geq \frac{1}{\lambda_t} \log \frac{n}{2\beta}$ and $U^{(t)}_j \leq -\frac{1}{\lambda_t} \log \frac{n}{2\beta}$. If there are more than one such indices, pick an index $i$ (or $j$) randomly. Encode $(i, j)$ as $m_t$.
  2. Let $V^{(t)}$ be an $n$-dimensional all-zero vector except $V^{(t)}_i = \frac{1}{\lambda_t} \log \frac{n}{2\beta}$ and $V^{(t)}_j = -\frac{1}{\lambda_t} \log \frac{n}{2\beta}$.
  3. Let $U^{(t+1)} = U^{(t)} - V^{(t)}$.
  4. Set $\lambda_{t+1}^2 = \frac{n}{n - 2 \log \frac{n}{2\beta}} \cdot \lambda_t^2$.

Recall the optimal marginal distribution:

$$f_V(v) = \lambda^2 D^2 \cdot \delta(v) + (1 - \lambda^2 D^2) \cdot \frac{\lambda}{2} e^{-\lambda|v|}$$
Results

![Graph 1: Sparsity vs. Iterations](image1)

- **Sparsity (%)** vs. **Iterations (x10^6)**

![Graph 2: Test Accuracy vs. Iterations](image2)

- **Test Accuracy (%)** vs. **Iterations (x10^6)**

- **Baseline** (blue line)
- **SuRP** (orange line)
## Results

### CIFAR-10

<table>
<thead>
<tr>
<th>Pruning Ratio</th>
<th>95.60%</th>
<th>98.20%</th>
<th>98.85%</th>
<th>99.26%</th>
<th>99.53%</th>
<th>99.81%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global [3]</td>
<td>90.80</td>
<td>85.55</td>
<td>81.56</td>
<td>54.58</td>
<td>41.91</td>
<td>21.87</td>
</tr>
<tr>
<td>Uniform [4]</td>
<td>90.78</td>
<td>84.17</td>
<td>55.68</td>
<td>38.51</td>
<td>26.41</td>
<td>11.58</td>
</tr>
<tr>
<td>VGG-16 Adaptive [1]</td>
<td>91.20</td>
<td>89.44</td>
<td>87.85</td>
<td>86.53</td>
<td>84.84</td>
<td>74.54</td>
</tr>
<tr>
<td>LAMP [2]</td>
<td>92.06</td>
<td>91.66</td>
<td>91.07</td>
<td>90.49</td>
<td>89.64</td>
<td>87.07</td>
</tr>
<tr>
<td>SuRP (ours)</td>
<td>92.13</td>
<td>91.72</td>
<td>91.21</td>
<td>90.73</td>
<td>90.65</td>
<td>87.28</td>
</tr>
</tbody>
</table>

### ResNet-50 on ImageNet

<table>
<thead>
<tr>
<th>Pruning Ratio</th>
<th>86.58%</th>
<th>94.50%</th>
<th>96.48%</th>
<th>97.75%</th>
<th>98.56%</th>
<th>99.41%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global [3]</td>
<td>86.97</td>
<td>85.02</td>
<td>83.15</td>
<td>80.52</td>
<td>76.28</td>
<td>47.47</td>
</tr>
<tr>
<td>Uniform [4]</td>
<td>86.70</td>
<td>84.53</td>
<td>82.05</td>
<td>77.19</td>
<td>64.24</td>
<td>20.45</td>
</tr>
<tr>
<td>ResNet-20 Adaptive [1]</td>
<td>87.00</td>
<td>85.00</td>
<td>83.23</td>
<td>80.40</td>
<td>76.40</td>
<td>52.06</td>
</tr>
<tr>
<td>LAMP [2]</td>
<td>87.12</td>
<td>85.64</td>
<td>84.18</td>
<td>81.56</td>
<td>78.63</td>
<td>67.01</td>
</tr>
<tr>
<td>SuRP (ours)</td>
<td>90.44</td>
<td>88.87</td>
<td>87.05</td>
<td>83.98</td>
<td>79.00</td>
<td>70.64</td>
</tr>
</tbody>
</table>

### Pruning Ratio: 80% and 90%

<table>
<thead>
<tr>
<th>Pruning Ratio</th>
<th>80%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adaptive [27]</td>
<td>75.60</td>
<td>73.90</td>
</tr>
<tr>
<td>SNIP [59]</td>
<td>72.00</td>
<td>67.20</td>
</tr>
<tr>
<td>DSR [74]</td>
<td>73.30</td>
<td>71.60</td>
</tr>
<tr>
<td>SNFS [18]</td>
<td>74.90</td>
<td>72.90</td>
</tr>
<tr>
<td>RiGL [22]</td>
<td>74.60</td>
<td>72.00</td>
</tr>
<tr>
<td>SuRP (ours)</td>
<td>75.54</td>
<td>73.93</td>
</tr>
</tbody>
</table>
SPARSE RANDOM NETWORKS FOR COMMUNICATION-EFFICIENT FEDERATED LEARNING

Joint work with
Francesco Pase (University of Padova)
Deniz Gunduz (Imperial College London)
Tsachy Weissman (Stanford University)
Michele Zorzi (University of Padova)
Contributions

1. Existence of subnetworks inside larger networks with random weights that perform well on clients’ non-iid dataset.

2. Finding these subnetworks in a communication-efficient way. (less than 1 bpp)

3. Fast convergence.

4. Efficient representation of the final model. (less than 1 bpp)

5. Privacy amplification in the presence of LDP mechanisms.
FedPM

Trainable probability mask

$\theta^t \in [0, 1]^d$

Stochastic binary mask

$m^t \sim \text{Bern}(\theta^t) \in \{0, 1\}^d$

Randomly weighted dense network

$w^{\text{init}} \in \mathbb{R}^d$

Randomly weighted sparse network

$\hat{w}^t = m^t \odot w^{\text{init}}$
Communication Strategy

True Mean: \( \bar{\theta}_{g,t} = \frac{1}{K} \sum_{k=1}^{K} \theta^k \)

Estimated Mean: \( \hat{\theta}_{g,t} = \frac{1}{K} \sum_{k=1}^{K} m_{k,t} \)

- Unbiased Estimate:
  \[ \mathbb{E}_{M^{k,t} \sim \text{Bern}(\theta^k,t) \ \forall k \in K_t} [\hat{\theta}_{g,t}] = \bar{\theta}_{g,t} \]

- Error:
  \[ \mathbb{E}_{M^{k,t} \sim \text{Bern}(\theta^k,t) \ \forall k \in K_t} \left[ \|\hat{\theta}_{g,t} - \bar{\theta}_{g,t}\|_2^2 \right] \leq \frac{d}{4K} \]
Results (CIFAR-10)

Server Test Accuracy (conv6-cifar10)

Average Bitrate (conv6-cifar10)

- FedPM (ours)
- SignSGD
- TernGrad
- QSGD
- DRIVE
- EDEN
- FedMask
Bayesian Aggregation

We can model the probability mask with a Beta distribution.

\[ \alpha^{g,t} = \alpha^{g,t-1} + M^{\text{agg},t} \]

\[ \beta^{g,t} = \beta^{g,t-1} + K \cdot 1 - M^{\text{agg},t} \]

where

\[ M^{\text{agg},t} = \sum_{k \in \mathcal{K}_t} M^{k,t} \]

\[ \theta^{g,t} = \frac{\alpha^{g,t} - 1}{\alpha^{g,t} + \beta^{g,t} - 2} \]
## Results

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>$\rho = 1$</th>
<th>$\rho = 0.5$</th>
<th>$\rho = 0.1$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>$c_{\text{max}} = 4$</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DRIVE (Vargaftik et al., 2021)</td>
<td>0.739 ± 0.005</td>
<td>0.632 ± 0.010</td>
<td>0.405 ± 0.018</td>
</tr>
<tr>
<td>EDEN (Vargaftik et al., 2022)</td>
<td>0.717 ± 0.006</td>
<td>0.665 ± 0.012</td>
<td>0.360 ± 0.016</td>
</tr>
<tr>
<td>QSGD (Alistarh et al., 2017)</td>
<td>0.709 ± 0.006</td>
<td>0.644 ± 0.014</td>
<td>0.399 ± 0.020</td>
</tr>
<tr>
<td>FedMask (Li et al., 2021)</td>
<td>0.531 ± 0.044</td>
<td>0.435 ± 0.057</td>
<td>0.362 ± 0.024</td>
</tr>
<tr>
<td>FedPM (Ours)</td>
<td><strong>0.748 ± 0.003</strong></td>
<td><strong>0.720 ± 0.007</strong></td>
<td><strong>0.496 ± 0.007</strong></td>
</tr>
<tr>
<td><strong>$c_{\text{max}} = 2$</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DRIVE (Vargaftik et al., 2021)</td>
<td>0.434 ± 0.025</td>
<td>0.376 ± 0.014</td>
<td>0.221 ± 0.003</td>
</tr>
<tr>
<td>EDEN (Vargaftik et al., 2022)</td>
<td>0.535 ± 0.050</td>
<td>0.461 ± 0.016</td>
<td>0.219 ± 0.005</td>
</tr>
<tr>
<td>QSGD (Alistarh et al., 2017)</td>
<td>0.476 ± 0.033</td>
<td>0.464 ± 0.002</td>
<td>0.243 ± 0.014</td>
</tr>
<tr>
<td>FedMask (Li et al., 2021)</td>
<td>0.420 ± 0.028</td>
<td>0.387 ± 0.062</td>
<td>0.197 ± 0.030</td>
</tr>
<tr>
<td>FedPM (Ours)</td>
<td><strong>0.643 ± 0.016</strong></td>
<td><strong>0.556 ± 0.031</strong></td>
<td><strong>0.277 ± 0.003</strong></td>
</tr>
</tbody>
</table>

Table 1: Average final accuracy $\pm \sigma$ in non-IID data split with $c_{\text{max}} = 4$ and $c_{\text{max}} = 2$, and partial participation with ratios $\rho = \{0.1, 0.5, 1\}$, for FedPM, FedMask, and the strongest baselines in the IID experiments: EDEN, DRIVE, and QSGD. The training duration was set to $t_{\text{max}} = 200$ rounds.
Thank You!

An Information-Theoretic Justification for Model Pruning (AISTATS’22)

Sparse Random Networks for Communication-Efficient Federated Learning
Results (CIFAR-100)

Server Test Accuracy (conv10-cifar100)

Average Bitrate (conv10-cifar100)
Results (MNIST)

Server Test Accuracy (conv4-mnist)

Average Bitrate (conv4-mnist)
Results (EMNIST)
## Results

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>$\rho = 1$</th>
<th>$\rho = 0.5$</th>
<th>$\rho = 0.1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRIVE (Vargaftik et al., 2021)</td>
<td>$0.885 \pm 9 \cdot 10^{-5}$</td>
<td>$0.885 \pm 1 \cdot 10^{-4}$</td>
<td>$0.885 \pm 1 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>EDEN (Vargaftik et al., 2022)</td>
<td>$0.885 \pm 1 \cdot 10^{-4}$</td>
<td>$0.885 \pm 1 \cdot 10^{-4}$</td>
<td>$0.885 \pm 1 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>QSGD (Alistarh et al., 2017)</td>
<td>$0.982 \pm 0.027$</td>
<td>$0.923 \pm 0.029$</td>
<td>$0.91 \pm 0.05$</td>
</tr>
<tr>
<td>FedMask (Li et al., 2021)</td>
<td>$1 \pm 3 \cdot 10^{-6}$</td>
<td>$1 \pm 8 \cdot 10^{-8}$</td>
<td>$1 \pm 6 \cdot 10^{-7}$</td>
</tr>
<tr>
<td>FedPM (Ours)</td>
<td>$0.863 \pm 0.077$</td>
<td>$0.912 \pm 0.056$</td>
<td>$0.996 \pm 0.003$</td>
</tr>
<tr>
<td>$c_{\text{max}} = 4$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>$\rho = 1$</th>
<th>$\rho = 0.5$</th>
<th>$\rho = 0.1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRIVE (Vargaftik et al., 2021)</td>
<td>$0.885 \pm 7 \cdot 10^{-5}$</td>
<td>$0.885 \pm 2 \cdot 10^{-4}$</td>
<td>$0.885 \pm 2 \cdot 10^{-4}$</td>
</tr>
<tr>
<td>EDEN (Vargaftik et al., 2022)</td>
<td>$0.885 \pm 1 \cdot 10^{-4}$</td>
<td>$0.885 \pm 7 \cdot 10^{-5}$</td>
<td>$0.885 \pm 7 \cdot 10^{-5}$</td>
</tr>
<tr>
<td>QSGD (Alistarh et al., 2017)</td>
<td>$1.230 \pm 0.043$</td>
<td>$1.234 \pm 0.038$</td>
<td>$1.082 \pm 0.01$</td>
</tr>
<tr>
<td>FedMask (Li et al., 2021)</td>
<td>$1 \pm 2 \cdot 10^{-6}$</td>
<td>$1 \pm 2 \cdot 10^{-6}$</td>
<td>$1 \pm 2 \cdot 10^{-7}$</td>
</tr>
<tr>
<td>FedPM (Ours)</td>
<td>$0.868 \pm 0.076$</td>
<td>$0.904 \pm 0.063$</td>
<td>$0.997 \pm 0.01$</td>
</tr>
<tr>
<td>$c_{\text{max}} = 2$</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Average bitrate $\pm \sigma$ over the whole training process in non-IID data split with $c_{\text{max}} = 4$ and $c_{\text{max}} = 2$, and partial participation with ratios $\rho = \{0.1, 0.5, 1\}$, for FedPM, FedMask, and the strongest baselines in the IID experiments: EDEN, DRIVE, and QSGD. The training duration was set to $t_{\text{max}} = 200$ rounds.