Unified Gradient Reweighting for Model Biasing with Applications to Source Separation

Efthymios Tzinis[1,*]
Dimitrios Bralios[1,2,*]
Paris Smaragdis[1,3]

1 2 3 Adobe
* Equal Contribution

UNIVERSITY OF ILLINOIS URBANA-CHAMPAIGN

Motivation

- Can we **take advantage of the bias** in neural networks?
- Can we control the **importance** of each training example and **shift the operating point** of our model towards a specified behavior?
- How can we use bias in order to make our estimation models more **robust**, **converge faster** and more **accurate for classes of interest**?

Conventional Gradient Updates

- Compute the gradient wrt the loss function and update the parameters using an **unbiased** estimator of the true gradient.

$$\boldsymbol{\theta}_{k+1} = \boldsymbol{\theta}_k - \eta \sum_{i=1}^{B} \frac{\mathbf{g}_k^{(i)}}{B}, \quad \mathbf{g}_k^{(i)} = \nabla_{\boldsymbol{\theta}_k} \mathcal{L}\left(f_{\boldsymbol{\theta}}\left(\mathbf{o}^{(i)}\right), \mathbf{s}^{(i)}\right)$$

- All the examples in each batch **contribute equally**.

$$\boldsymbol{\delta}_k = \underset{\mathcal{U}\{1,B\}}{\mathbb{E}}\left[\mathbf{g}_k^{(i)}\right] = \sum_{i=1}^{B} \frac{1}{B} \mathbf{g}_k^{(i)}$$

Unified Gradient Reweighting

We **generalize the updates using a user defined pmf** in order to weight the importance of the training examples non-uniformly, based on the **operation point** that we want to **shift our model towards**.

$$\boxed{\widetilde{\boldsymbol{\delta}}_k = \underset{p_k}{\mathbb{E}}\left[\mathbf{g}_k^{(i)}\right] = \sum_{i=1}^{B} p_k\left(\mathbf{o}^{(i)}, \mathbf{s}^{(i)}\right) \mathbf{g}_k^{(i)}}$$

Softmax Gradient Reweighting

Although we could define any valid pmf we propose the following simple and flexible parameterized family of distributions:

- Given an observed signal **o** and the corresponding target signals **s** for each example in the batch, we can define a weighting function **F** which can also be dynamically evolving across optimization iterations. **k** denotes the iteration index and **i, j** are batch indices.

$$\boxed{p_k\left(\mathbf{o}^{(i)}, \mathbf{s}^{(i)}\right) = \frac{\exp(\mathrm{F}_k(\mathbf{o}^{(i)}, \mathbf{s}^{(i)}))}{\sum_{j=1}^{B} \exp(\mathrm{F}_k(\mathbf{o}^{(j)}, \mathbf{s}^{(j)}))}, \quad \forall\ i, k}$$

[1] Tzinis et al., "Sudo RM -RF: Efficient Networks for Universal Audio Source Separation," MLSP 2020.
[2] J. Le Roux, et al., "Sdr–half-baked or well done?," ICASSP 2019.

Experimental Setup

- We perform experiments on speech (utterances from WSJ) and environmental sound (drawn from ESC 50) separation as well as their cross-product combinations.
- We utilize the **Sudo -rm rf** [1] model which provides a good trade-off between separation performance and computational requirements.
- We **configure the weighting function F** in order to show how we can tackle real-world problems using our gradient reweighting method.
- We use the as signal level loss function the negative permutation invariant scale-invariant signal to distortion ratio (SI-SDR) [2].

$$\text{SI-SDR}(\hat{\mathbf{s}}, \mathbf{s}^*) = -10 \log_{10}\left(\|\rho \mathbf{s}^*\|^2 / \|\rho \mathbf{s}^* - \hat{\mathbf{s}}\|^2\right) \qquad \rho = \hat{\mathbf{s}}^\top \mathbf{s}^* / \|\mathbf{s}\|^2$$

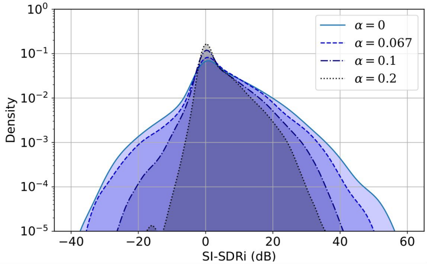
- We evaluate all our models using SI-SDR improvement (SI-SDRi) over the input mixture.

Results and Discussion

Robust Separation (on environmental sound separation)

$$\boxed{\widehat{\mathrm{F}}_k(\mathbf{o}^{(i)}, \mathbf{s}^{(i)}) = \alpha \mathcal{L}(\hat{\mathbf{s}}^{(i)}, \mathbf{s}^{(i)}), \quad \alpha > 0}$$

- We can control the **trade-off** between the **mean estimation accuracy and robustness**
- In many real-world applications, **a more robust model might be preferred over a more accurate (on average) model with higher variance.**
- Increasing alpha leads to put more weight on "difficult" examples.



α	Statistics: mean	std	Quantiles: 1	5	10	25	50	75	90	95	99
0	**5.0**	9.0	-17.5	-6.9	-2.5	0.0	**2.7**	**9.5**	**17.1**	**22.0**	**31.5**
1/15	4.6	7.8	-14.7	-4.4	-1.4	**0.1**	2.4	8.4	15.2	19.4	29.0
1/10	3.6	6.0	-6.5	-2.2	-1.0	-0.1	1.1	6.1	12.1	16.1	23.5
1/5	3.0	**4.8**	**-2.8**	**-0.9**	**-0.3**	0.0	0.6	4.9	9.8	13.2	19.8

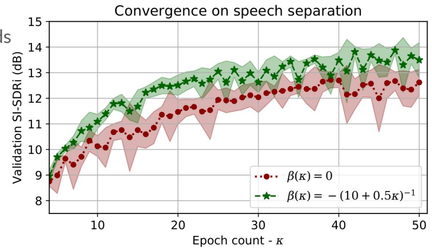
Test SI-SDRi (dB)

Faster Convergence (Curriculum Learning)

We make the model be more biased towards learning the "**easy**" examples (with lower value of loss) first, and gradually converging to a uniform distribution .

$$\boxed{\widetilde{\mathrm{F}}_k(\mathbf{o}^{(i)}, \mathbf{s}^{(i)}) = \beta(k) \mathcal{L}(\hat{\mathbf{s}}^{(i)}, \mathbf{s}^{(i)})}$$

- The gradient reweighted configuration yields a **much faster convergence** in terms of mean SI-SDRi for the same number of training epochs compared to the baseline with unbiased updates.



Biasing the model towards specific classes

We train the model using mixtures with sources from both *speech* and *environmental* (*Env.*) sounds. We use higher values of gamma for the corresponding class that we are mostly interested in.

$$\boxed{\check{\mathrm{F}}_k(\mathbf{o}^{(i)}, \mathbf{s}^{(i)}) = \gamma(c^{(i)})}$$

γ Speech	γ Env.	Mean test SI-SDRi (dB) Speech	Env.	Combined
0	0	12.2 ± 0.1	13.1 ± 0.1	12.7 ± 0.1
0	3	11.8 ± 0.2	$\mathbf{13.5 \pm 0.1}$	12.6 ± 0.1
3	0	$\mathbf{12.7 \pm 0.1}$	13.1 ± 0.1	$\mathbf{12.9 \pm 0.1}$

We can get a **significant boost** in the reconstruction quality for the class that we choose the higher weight over the baseline (same weights).

Conclusions

- We have presented a **simple** and **easily extendable** unified gradient reweighting scheme with **negligible computational cost**.
- We showed that we can use it towards solving multiple real-world problems appearing in the process of training separation networks, such as: **robustness**, **convergence** and **adaptation to specific classes**.