**Abstract**

Residual partitioning is a second-order optimization algorithm for training neural nets.

- In each iteration, residual partitioning uses Jensen’s inequality to bound the objective function.
- The bound has a diagonal Hessian, and so is easy to optimize.
- We compare with Adam and SGD by training an autoencoder and show residual partitioning quickly converges to a better or comparable solution.

**Introduction**

Training a neural net is a hard optimization problem. The curvature of the objective function varies greatly in different directions, causing SGD to zig-zag in directions of high curvature and converge slowly in directions of low curvature. Fast optimization algorithms estimate the curvature and adjust the learning rate for each parameter accordingly.

- Adaptive learning rate methods like Adam estimate the curvature from past gradients.
- Newton’s method uses the full Hessian matrix, but requires solving a dense linear system in each iteration.
- Approximating the Hessian by its diagonal is fast, but struggles with the non-convexity of the objective function.

**Setup**

We present a simplified version of residual partitioning here: if the score function is $L$, the network parameters are $w_i$, the output of the network is $y_i$, and the change in network parameters and output are $\Delta w_i$ and $\Delta y_i$, then the objective function is approximately

$$\max_{\{\Delta w_i\}} \frac{1}{2} \sum_{i=1}^n \left( \sum_{s=1}^n \frac{\partial L}{\partial y_s} (\Delta y_s) + \frac{1}{2} \frac{\partial^2 L}{\partial y_s^2} (\Delta y_s)^2 \right) \quad (1)$$

$$\Delta y_i \equiv \sum_{s=1}^n \frac{\partial y_i}{\partial w_s} \Delta w_s \quad (2)$$

**The Bound**

This objective function has a dense Hessian matrix: the $(i, j)$ component of the Hessian is $\frac{\partial^2 L}{\partial y_i \partial y_j}$. However, we can lower bound the objective by bounding $(\Delta y_i)^2$ with Jensen’s inequality: first, introduce a set of partitioning variables $\{\varepsilon_{si}\}$ that add to one. Then equation (3) holds due to Jensen’s inequality.

$$\sum_{s=1}^n \frac{\partial y_i}{\partial w_s} \sum_{i=1}^n \varepsilon_{si} = 1$$

$$(\Delta y_i)^2 = \left( \sum_{s=1}^n \frac{\partial y_i}{\partial w_s} \Delta w_s \right)^2 \leq \sum_{s=1}^n \left( \frac{\partial y_i}{\partial w_s} \right)^2 \sum_{i=1}^n \varepsilon_{si} (\Delta w_i)^2 \quad (3)$$

**Residual Partitioning Bound**

$$\sum_{s=1}^n \frac{\partial^2 L}{\partial y_s \partial y_i} (\Delta y_s)(\Delta w_i) \quad (4)$$

The solution is

$$\varepsilon_{si} = \frac{|\partial y_i / \partial w_s|}{\sum_{i=1}^n |\partial y_i / \partial w_i|} \quad (5)$$

This says we should turn down the learning rate for parameters with small gradients, and turn up the learning rate for parameters with big gradients. Residual partitioning optimizes a bound on the objective, which overestimates the curvature and yields smaller than optimal learning rates. This can be fixed by scaling $\Delta w_i$ by a global learning rate $\lambda > 1$.

**Choosing Partitioning Variables**

The learning rates returned by residual partitioning depend on the choice of partitioning variables. Note that the learning rate for $w_i$ will be large if $\varepsilon_{si}$ is large, but the $\varepsilon_{si}$ add to one, so the learning rates cannot all be large. We can choose the partitioning variables so the learning rates will be large on average by minimizing the sum of the inverse learning rates:

$$\min \sum_{s=1}^n \sum_{i=1}^n \varepsilon_{si} \frac{\partial^2 L}{\partial y_s \partial y_i} (\Delta y_s)(\Delta w_i) \quad (6)$$

The solution is

$$\varepsilon_{si} = \frac{|\partial y_i / \partial w_s|}{\sum_{i=1}^n |\partial y_i / \partial w_i|} \quad (5)$$

We compare residual partitioning with Adam and SGD by training an autoencoder with 5 layers on MNIST, Fashion-MNIST, and CIFAR-10. We used mini-batch sizes of 100, and tuned the global learning rate for each algorithm and dataset. The results are in Figures (b)-(d). We observe that residual partitioning quickly converges to a better or comparable solution and overfits less than Adam and SGD.