Review on “The Reversible Residual Network: Backpropagation Without Storing Activations”

Authors: A. N. Gomez, M. Ren, R. Urtasun, R. B. Grosse

# Short Summary

This paper presents RevNet, a variant of ResNet where activations **for most layers** are calculated from one layer to the next instead of stored. The authors show that this network can achieve similar classification accuracy to equally-sized ResNets but storage requirements are independent of network depth.

ResNet introduced a ‘residual’ block which allows input into the layer to be directly passed through to the output. As a result, back propagation can be performed with mitigated risk of vanishing/exploding gradients. This allowed for bigger networks which could achieve greater results and introduced the need to deal with the memory requirement of storing activations (which is what this paper addresses).

RevNet partitions the units in each layer into two groups (partitioning channels) with two residual functions that relate the groups. Any layer that is intended to be reversible must possess a stride of 1 to avoid discarding information. During backpropagation, a Reversible Residual Block Backprop algorithm is able to compute the necessary signal and gradient parameters to update weights with a realistic computational overhead of about 50%. This yields a storage cost and a computation cost where L is the number of layers with a unit forward or backpropagation cost. In the experiments comparing ResNet and RevNet, there was less than 0.5% classification performance degradation.

# Main Contributions

* Introduced RevNet and an associated network architecture
* Introduced an algorithm that enables backpropagation in a reversible layer
* Demonstrated that classification performance remains comparable to ResNet
* Provided sample implementation of network to recreate experiment

# High-Level Evaluation of Paper

Compared to other papers I’ve read recently, I found this one particularly easy to read and engaging. The structure and writing style were clear and concise in most cases. I did struggle a bit to understand their interpretation of backpropagation as a computation graph but that is likely due to unfamiliarity with graph theory. While the figures, equations and provided algorithm were generally easy to follow, I did not understand why the derivatives and are different when and this seems to be an important aspect of the algorithm.

In other regards, the complexity analysis of storage and memory costs (as well as comparisons to other methods) were welcome additions in terms of understanding the trade-offs of the proposed network architecture. On that note, I appreciate that the authors touched on the limitations of some of their approaches, investigated them and offered potential alternatives.

# Discussion on Evaluation Methodology

The major evaluation in this paper was the comparison of ResNet and RevNet. The experimental setup and example code is provided in the appendix allowing a reader to potentially recreate the results. To ensure that the testing was fair, the authors tried to match the computational depth and parameters of both architectures as closely as possible. It seems, however, that this was performed qualitatively rather than quantitatively. Comparisons between the two networks encompass three different sizes on CIFAR-10, CIFAR-100 and ImageNet demonstrating that RevNet and ResNet perform similarly. The training curves are helpful in gaining insight into the classification performance of the two approaches.

I think it would have been beneficial to also include a curve comparing the training speed of both methods to show whether the expected computational overhead of RevNet is a significant factor.

# Possible Directions for Future Work

As mentioned in the paper, this network is being investigated in areas where memory limitations are constraining performance gains such as semantic segmentation and recurrent neural networks. Theoretically, RevNets should displace ResNets if the computational overhead can be mitigated. Perhaps, one should investigate the trade-off between reversible and non-reversible layers in a network to get an idea of what type of structure would yield the optimal balance between memory and computation.