Main Idea

- Introduce a new method for stochastic optimization, Adam, which uses adaptive estimates of lower-order moments
- By Diederik P. Kingma and Jimmy Lei Ba
- The motivation of this paper is to introduce a new optimization algorithm for neural networks that only requires first-order gradients and little memory, and outperforms other algorithms in terms of simplicity, efficiency, and accuracy.

Summary

- The paper explains how the Adam algorithm works. There are multiple parts to the algorithm. The algorithm uses the first and second moments of the gradient as well as bias-corrected estimates for those moments to update the weight parameters, which is essentially a combination of RMSprop and momentum. The paper also presents proof as to the convergence of the optimization algorithm.

Approach and Contributions

- Arguments
    - Stochastic gradient-based optimization algorithms are the backbone of many science and engineering fields. Thus, having an efficient method is key to improve speed and accuracy of many methods.
- Approach and Findings
    - Approach
        - Update rule
            - Obtain gradients at timestep t
            - Update biased first moment estimate (weighted moving average of gradient)
            - Update biased second moment estimate (weighted moving average of squared gradient)
            - Correct bias for the first and second moments
            - Update parameters ($\theta t \leftarrow \theta t-1 - \alpha \cdot \hat{} mt/(\sqrt{} vt + e)$)
        - Convergence analysis
            - The convergence of the algorithm is evaluated using regret, and the Adam algorithm has O(sqrt(T)) regret bound, where T is the total number of timesteps. This is comparable to the best known bound for this general convex online learning problem.
    - Findings
        - The researchers empirically evaluated the performance of Adam compared to other algorithms.
            - Logistic regression

        - MNIST images: logistic regression performed much better when using the Adam optimization algorithm compared to AdaGrad. It was slightly better than SGD.
        - IMDB movie rating: Tested the algorithms' ability to handle sparse datasets. Adam performed much better than SGD, and slightly better than RMSprop and Adagrad with less noise.
    - Multi-layer neural network
        - Evaluated Adam vs SFO (sum of functions method) using a neural network with 2 hidden layers and 1000 neurons per layer with ReLU as the activation function.
        - SFO is 5-10x slower than Adam, and Adam shows better convergence and others.
    - CNN
        - Evaluated on a CNN architecture with 3 alternating stages of 5x5 kernels and 3x3 max pooling with stride of 2 followed by a fully connected layer with 1000 neurons and ReLU as the activation function.
        - Overall, Adam outperformed all other optimization methods, and converged much faster.

Improvements

- Although the paper goes into great detail about the derivation of Adam and proof of convergence, it would be nice for the researchers to double down on the mathematical rigor by presenting proof as to why Adam converged faster and is more efficient than other methods. Even though the empirical findings are convincing, mathematical rigor would be even more solid proof of the merits of Adam.