The Marginal Value of Adaptive Gradient Methods in Machine Learning

Presented by:
Alokendu Mazumder (201916003)
Pious Pradhan (201916010)

Conference Details

Published at:

Advances in Neural Information Processing Systems 30 (NIPS 2017) – Long Beach, California

Authored by:

Ashia C. Wilson (UC Berkeley),
Rebecca Roelofs (UC Berkeley),
Mitchell Stern (UC Berkeley),
Nathan Srebro (Toyota technological Institute, Chicago) &
Benjamin Recht (UC Berkeley)

Problem Statement

- COMPARING VARIOUS OPTIMIZATION TECHNIQUES.

- STUDYING GENERALIZING CAPABILITIES OF ADAPTIVE OPTIMIZATION TECHNIQUES AND FAMILY OF GRADIENT DESCENT ON OVERPARAMETERIZED PROBLEMS.

Family of Non-Adaptive Methods

1. Stochastic Gradient Descent

$$w_{k+1} = w_k - \alpha_k \tilde{\nabla} f(w_k)$$

where $\tilde{\nabla} f(w_k) := \nabla f(w_k; x_{i_k})$ is the gradient of some loss function f computed on a batch of data x_{i_k}.

2. Stochastic Momentum Methods

$$w_{k+1} = w_k - \alpha_k \tilde{\nabla} f(w_k + \gamma_k(w_k - w_{k-1})) + \beta_k(w_k - w_{k-1})$$

Polyak's heavy-ball method (HB) with $\gamma_k = 0$,

Nesterov's Accelerated Gradient method (NAG) with $\gamma_k = \beta_k$

Family of Adaptive Methods

1. **AdaGrad**

$$v_t^w = v_{t-1}^w + (\nabla w_t)^2$$

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{v_t^w + \epsilon}} * \nabla w_t$$

2. **RMSProp**

$$v_t^w = \beta * v_{t-1}^w + (1 - \beta)(\nabla w_t)^2$$

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{v_t^w + \epsilon}} * \nabla w_t$$

3. **Adam**

$$m_t = \beta_1 * m_{t-1} + (1 - \beta_1) * \nabla w_t$$

$$v_t = \beta_2 * v_{t-1} + (1 - \beta_2) * (\nabla w_t)^2$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \qquad \hat{v}_t = \frac{v_t}{1 - \beta_2^t}$$

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} * \hat{m}_t$$

Setting up the Problem

We have a loss function defined below, Where **X** is an *n x d* Matrix (*n* > *d*) of features & **y** is a *n-dimensional* vector of labels in {-1,1}. Our aim is to find the best Linear Classifier **w**.

$$\text{minimize}_w \quad R_S[w] := \tfrac{1}{2}\|Xw - y\|_2^2$$

Generalized solution by Non-Adaptive Family

Most common non-adaptive methods will find the same solution for the least squares objective.

Claim:
Any gradient or stochastic gradient of $\mathbf{R_S}$ must lie in the span of the rows of **X**. Therefore, any method that is initialized in the row span of **X** (say, for instance at $\mathbf{w} = 0$) and uses only linear combinations of gradients, stochastic gradients, and previous iterates must also lie in the row span of **X**. The unique solution that lies in the row span of **X** also happens to be the solution with minimum Euclidean norm. We thus denote $\mathbf{w}^{SGD} = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{y}$

Proof

$$R = (Xw - y)^T(Xw - y) =$$
$$(Xw)^T(Xw - y) - y^T(Xw - y) =$$
$$(Xw)^T(Xw) - (Xw)^T y - y^T(Xw) + y^T y$$
$$w^T X^T Xw - 2y^T Xw + y^T y$$
$$w^T X^T Xw - 2(X^T y)^T w + y^T y$$

Using

$$\frac{d}{dw} y^T w = \frac{d}{dw} w^T y = y$$
$$\frac{d}{dw} w^T Xw = (X + X^T)w$$

We get:

$$\frac{d}{dw} R = (X^T X + XX^T)w - 2X^T y = 2X^T Xw - 2X^T y$$

Thus, $\frac{d}{dw} R \in$ span{ rows of X }, since for arbitrary vector a, $X^T a$ for is a linear combination of the rows of X.

Proof (contd...)

As seen above, that the unique solution lies in the Row Space of **X**. Now we shall see that if we initialize **w** in Row Space of X (i.e $\mathbf{w_0}$=0 ; as Row Space is a subspace), then the final solution given by Non-Adaptive Gradient Descent will also lie inside the Row Space of **X**.

$\mathbf{w_1} = \mathbf{w_0} - \alpha \nabla \mathbf{R_S}[w_0]$ **;** as $\mathbf{w_0}$ & $\nabla \mathbf{R_S}[w_0]$ both lies in Row Space of **X**, $\mathbf{w_1}$ also lies in Row Space of **X**.

Similarly $\mathbf{w_k}$, i,e the optimal solution given by GD family also lies in the Row Space of **X**.

Special case solution by Adaptive Family

It is difficult to derive the general form of solution, we can analyze special cases.

Lemma 3.1 *Suppose there exists a scalar c such that $X\ \mathrm{sign}(X^T y) = cy$. Then, when initialized at $w_0 = 0$, AdaGrad, Adam, and RMSProp all converge to the unique solution $w \propto \mathrm{sign}(X^T y)$.*

In other words, whenever there exists a solution of $Xw = y$ that is proportional to $\mathrm{sign}(X^T y)$, this is precisely the solution to which all of the adaptive gradient methods converge.

As it's clearly mentioned in the above lemma that Adaptive methods converge to the unique solution iff there exist a scalar "c", hence it can't be generalized.

Proof is given on the next slide.

Proof We prove this lemma by showing that the entire trajectory of the algorithm consists of iterates whose components have constant magnitude. In particular, we will show that

$$w_k = \lambda_k \operatorname{sign}(X^T y)\,,$$

for some scalar λ_k. The initial point $w_0 = 0$ satisfies the assertion with $\lambda_0 = 0$.

Now, assume the assertion holds for all $k \leq t$. Observe that

$$\begin{aligned}
\nabla R_S(w_k + \gamma_k(w_k - w_{k-1})) &= X^T(X(w_k + \gamma_k(w_k - w_{k-1})) - y) \\
&= X^T\left\{(\lambda_k + \gamma_k(\lambda_k - \lambda_{k-1}))X \operatorname{sign}(X^T y) - y\right\} \\
&= \left\{(\lambda_k + \gamma_k(\lambda_k - \lambda_{k-1}))c - 1\right\} X^T y \\
&= \mu_k X^T y,
\end{aligned}$$

where the last equation defines μ_k. Hence, letting $g_k = \nabla R_S(w_k + \gamma_k(w_k - w_{k-1}))$, we also have

$$\mathrm{H}_k = \operatorname{diag}\left(\left\{\sum_{s=1}^{k} \eta_s\, g_s \circ g_s\right\}^{1/2}\right) = \operatorname{diag}\left(\left\{\sum_{s=1}^{k} \eta_s \mu_s^2\right\}^{1/2} |X^T y|\right) = \nu_k \operatorname{diag}\left(|X^T y|\right),$$

where $|u|$ denotes the component-wise absolute value of a vector and the last equation defines ν_k.

In sum,

$$\begin{aligned}
w_{k+1} &= w_k - \alpha_k \mathrm{H}_k^{-1} \nabla f(w_k + \gamma_k(w_k - w_{k-1})) + \beta_t \mathrm{H}_k^{-1} \mathrm{H}_{k-1}(w_k - w_{k-1}) \\
&= \left\{\lambda_k - \frac{\alpha_k \mu_k}{\nu_k} + \frac{\beta_k \nu_{k-1}}{\nu_k}(\lambda_k - \lambda_{k-1})\right\} \operatorname{sign}(X^T y),
\end{aligned}$$

proving the claim.

Overfitting for Adaptivity

The above Lemma allows us to construct a particularly pernicious generative model where AdaGrad fails to find a solution that generalizes. This example uses infinite dimensions for simplicity.

For i = 1,...,n, sample the label y_i to be 1 with probability p and -1 with probability 1- p for some p > 1/2. Let x_i be an infinite dimensional vector with entries

$$x_{ij} = \begin{cases} y_i & j = 1 \\ 1 & j = 2, 3 \\ 1 & j = 4 + 5(i-1), \ldots, 4 + 5(i-1) + 2(1 - y_i) \\ 0 & \text{otherwise} \end{cases}$$

Take ***n*** samples and consider the AdaGrad solution for minimizing $||\mathbf{Xw}-\mathbf{y}||_2^2$ First we show that the conditions of Lemma 3.1 hold. Let b = $\mathbf{\Sigma}_i \mathbf{y}_i$ and assume for the sake of simplicity that b > 0. This will happen with arbitrarily high probability for large enough ***n***. Define $\mathbf{u} = \mathbf{X}^T \mathbf{y}$ and observe that,

$$u_j = \begin{cases} n & j = 1 \\ b & j = 2, 3 \\ y_j & \text{if } j > 3 \text{ and } x_{\lfloor \frac{j+1}{5} \rfloor, j} = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\text{sign}(u_j) = \begin{cases} 1 & j = 1 \\ 1 & j = 2, 3 \\ y_j & \text{if } j > 3 \text{ and } x_{\lfloor \frac{j+1}{5} \rfloor, j} = 1 \\ 0 & \text{otherwise} \end{cases}$$

Hence, the AdaGrad solution $\mathbf{w}^{\mathbf{ada}}$ α sign($\mathbf{u}$). In particular $\mathbf{w}^{\mathbf{ada}}$, has all of its components equal to $\pm\tau$ for some positive constant τ . But for a new data point, $\mathbf{x}^{\mathbf{test}}$, the only features that are nonzero in both $\mathbf{x}^{\mathbf{test}}$ and $\mathbf{w}^{\mathbf{ada}}$ are the first three. In particular, we have

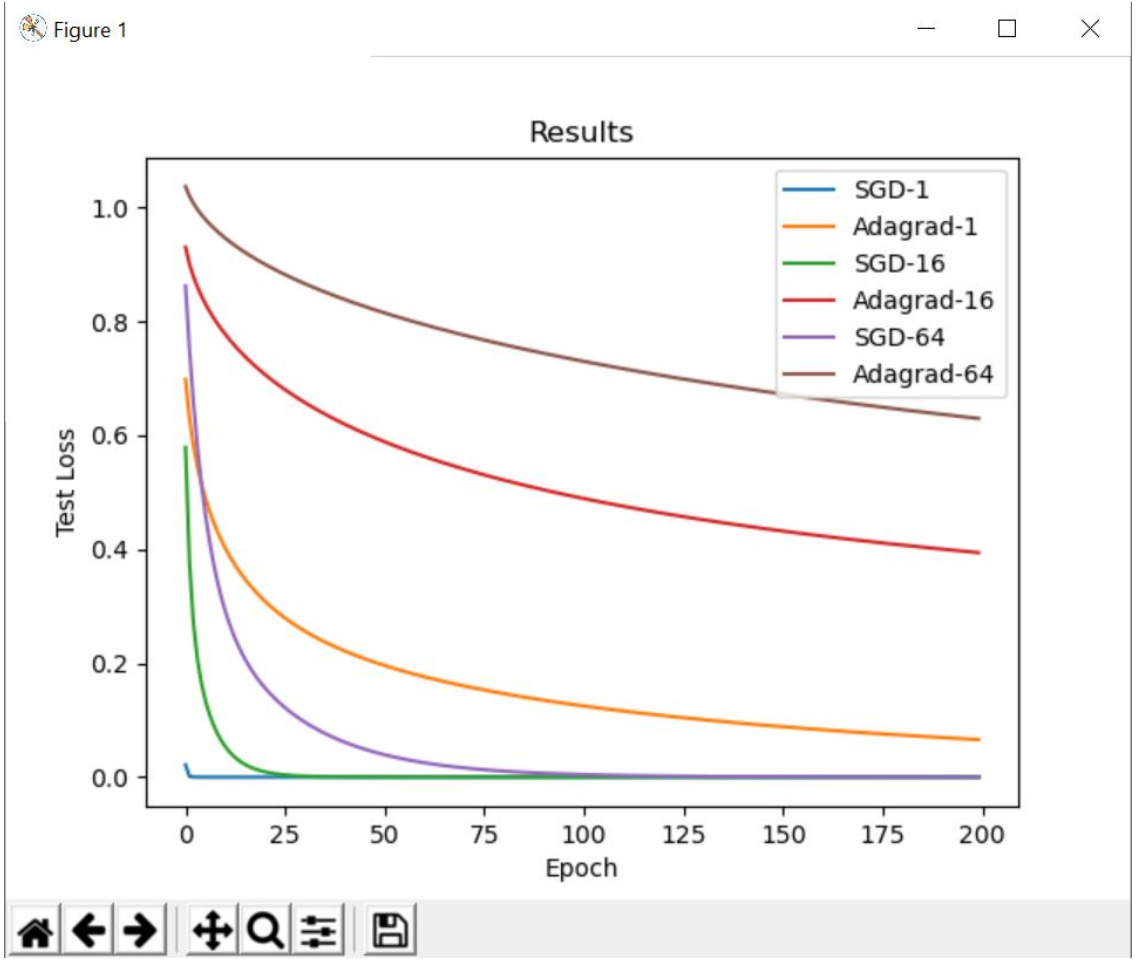
$$\langle w^{\text{ada}}, x^{\text{test}} \rangle = \tau(y^{(test)} + 2) > 0$$

Therefore, the AdaGrad solution will label all unseen data as a positive example!

Experimental Results

No. of samples = 2000

No. of Dimensions = 6n



Conclusion

- Experimental evidence & Mathematical proofs demonstrates that adaptive methods do not generalizes the solution well as compared to non-adaptive ones, hence, not advantageous for machine learning, still the Adam algorithm remains incredibly popular.
- Even when the adaptive methods achieve the same training loss or lower than non-adaptive methods, the development or test performance is worse.
- Adaptive methods often display faster initial progress on the training set, but their performance quickly plateaus on the development set.

References

- AdaGrad, RMSProp, Adam - NPTEL
  https://nptel.ac.in/courses/106/105/106105215/
- Gradient Descent - MIT OCW
  https://ocw.mit.edu/courses/mathematics/18-065-matrix-methods-in-data-analysis-signal-processing-and-machine-learning-spring-2018/video-lectures/lecture-26-structure-of-neural-nets-for-deep-learning/
- Stochastic Gradient Descent - Visualization
  http://fa.bianp.net/teaching/2018/COMP-652/stochastic_gradient.html
- All Gradient Formulae - MEDIUM
  https://towardsdatascience.com/learning-parameters-part-5-65a2f3583f7d

Thank You