Deep Learning Local Optima

Alireza Shafaei - Dec. 2015

Saddle Points

A critical point $\|\nabla f(x)\| \to 0$

Saddle Points

A critical point $\|\nabla f(x)\| \to 0$

With Hessian

Local minimum: $\forall_i \ \lambda_i > 0$

Saddle Points

A critical point $$\|\nabla f(x)\| \to 0$$

With Hessian

Local minimum: $\forall_i \ \lambda_i > 0$

Local maximum: $\forall_i \ \lambda_i < 0$

Saddle Points

With Hessian

...

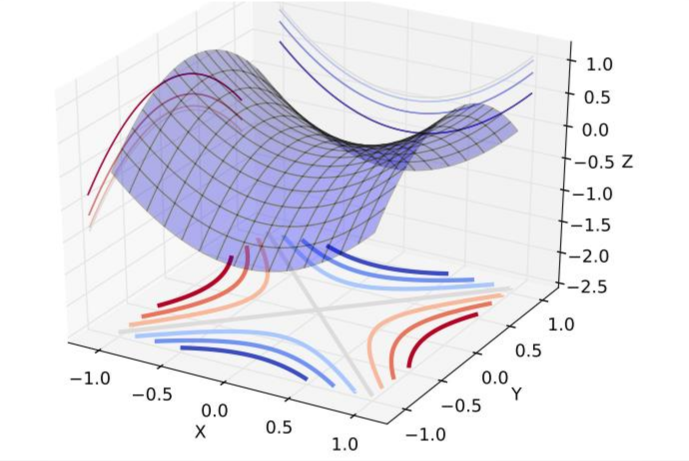
Saddle point with min-max structure $\forall_i \ \lambda_i \neq 0$

Saddle Points

With Hessian

...

Saddle point with min-max structure $\forall_i \ \lambda_i \neq 0$



Saddle Points

With Hessian

...

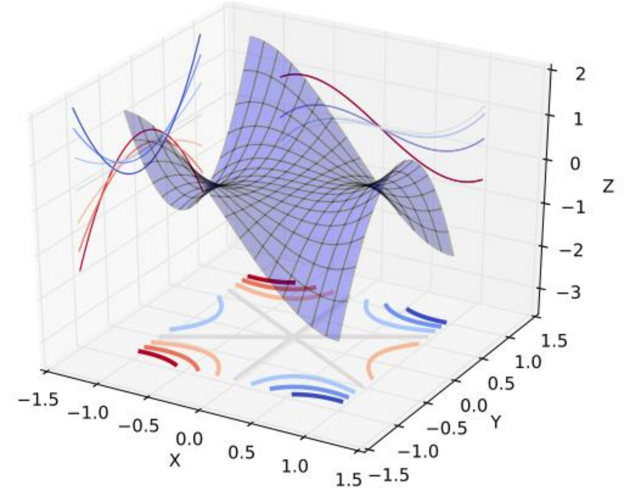
Saddle point with Monkey structure: $\exists_i \ \lambda_i = 0$

Saddle Points

With Hessian

...

Saddle point with Monkey structure: $\exists_i \ \lambda_i = 0$



Optimization Algorithms around Saddle Points

Gradient Descent

Steps proportional to the eigenvalues of the Hessian.

Optimization Algorithms around Saddle Points

Gradient Descent

Steps proportional to the eigenvalues of the Hessian.

Slows down near the saddle point!

Optimization Algorithms around Saddle Points

Newton method

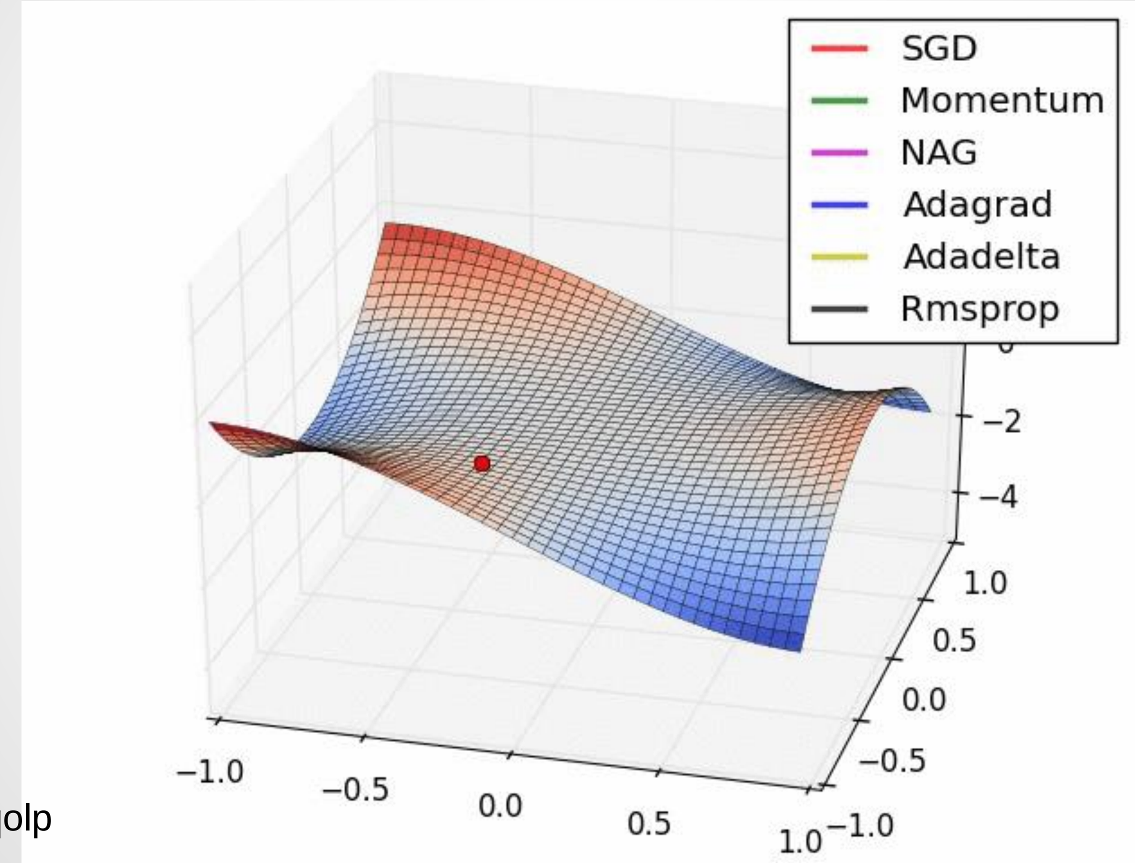
Takes steps proportional to the inverse of eigenvalues.

Optimization Algorithms around Saddle Point

Newton method

- Takes steps proportional to the inverse of eigenvalues.
- Steps are in the opposite direction if an eigenvalue is negative!
- Attracted to saddle points!

Optimization Algorithms around Saddle Point



http://imgur.com/a/Hqolp

Where are we headed?

Deep networks probably have exponentially (in dimension) many saddle points with high errors!

Where are we headed?

Deep networks probably have exponentially (in dimension) many saddle points with high errors!

- Statistical Physics (spin-glass)
- Random Matrix Theory
- Previous theoretical work on neural networks.

Where are we headed?

- Deep networks probably have exponentially (in dimension) many saddle points with high errors!
  - Statistical Physics (spin-glass)
  - Random Matrix Theory
  - Previous theoretical work on neural networks.
- We probably don’t have to worry about bad local minima in large networks.

Spin-Glass Model

$$H = \sum_{i,k=1,N} J_{ik}\sigma_i\sigma_k \qquad \sigma \in \{+1,-1\}$$

$$\Sigma \in \{+1,-1\}^N$$

Random Matrix Theory

In large Gaussian random matrices, the probability of an eigenvalue to be positive or negative is ½.

$$\Pr[\forall_i \; \lambda_i > 0] = (\frac{1}{2})^N$$

Random Matrix Theory

In large Gaussian random matrices, the probability of an eigenvalue to be positive or negative is ½.

$$\Pr[\forall_i \ \lambda_i > 0] = (\frac{1}{2})^N$$

The Hessian at a random point on a spin-glass model with Gaussian edges is a Gaussian random matrix.

Y. Dauphin, … , and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization” NIPS, 2014.

Existence of saddle points in deep networks is empirically shown.

Y. Dauphin, … , and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization” NIPS, 2014.

Existence of saddle points in deep networks is empirically shown.

(re)introduce a saddle-free Newton method.

Y. Dauphin, … , and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization” NIPS, 2014.

Existence of saddle points in deep networks is empirically shown.

(re)introduce a saddle-free Newton method.

Compare on optimization of MLP, RNN, Deep autoencoders.

Y. Dauphin, … , and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization” NIPS, 2014.

Existence of saddle points in deep networks.

- α : Fraction of the negative eigenvalues of the Hessian at the critical point.
- ϵ : Error at the critical point.

Y. Dauphin, … , and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization” NIPS, 2014.

Existence of saddle points in deep networks.

- α : Fraction of the negative eigenvalues of the Hessian at the critical point.
- ϵ : Error at the critical point.

In Hamiltonian of a Gaussian Spin-Glass model there’s a positive correlation between these two measure.

Y. Dauphin, … , and Y. Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization" NIPS, 2014.

Existence of saddle points in deep networks.

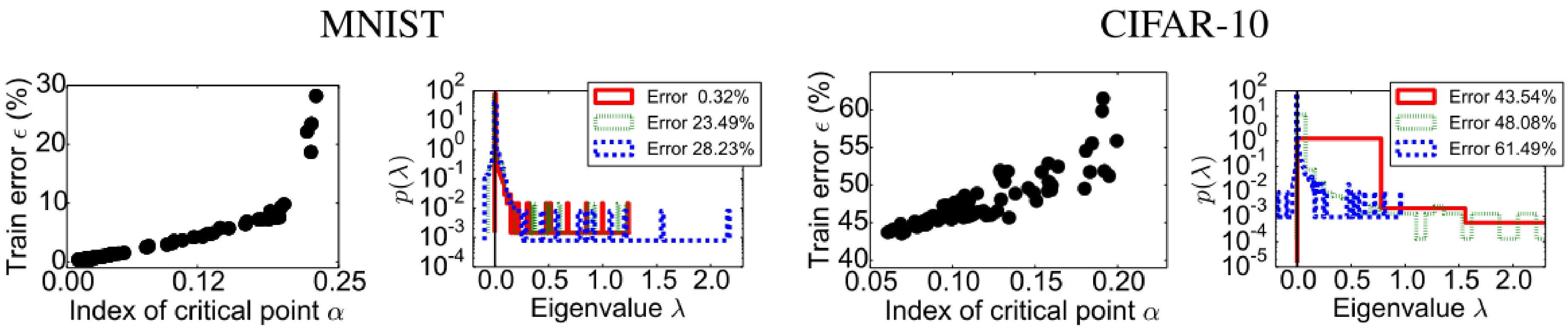
- α : Fraction of the negative eigenvalues of the Hessian at the critical point.
- ϵ : Error at the critical point.

In Hamiltonian of a Gaussian Spin-Glass model there's a positive correlation between these two measure.

Is it the same for neural networks?

Y. Dauphin, … , and Y. Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization" NIPS, 2014.

Existence of saddle points in deep networks.



Y. Dauphin, … , and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization” NIPS, 2014.

(re)introduce a saddle-free Newton method.

- A corrected Newton method where steps are proportional to $\dfrac{1}{|\lambda_i|}$

Y. Dauphin, … , and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization” NIPS, 2014.

(re)introduce a saddle-free Newton method.

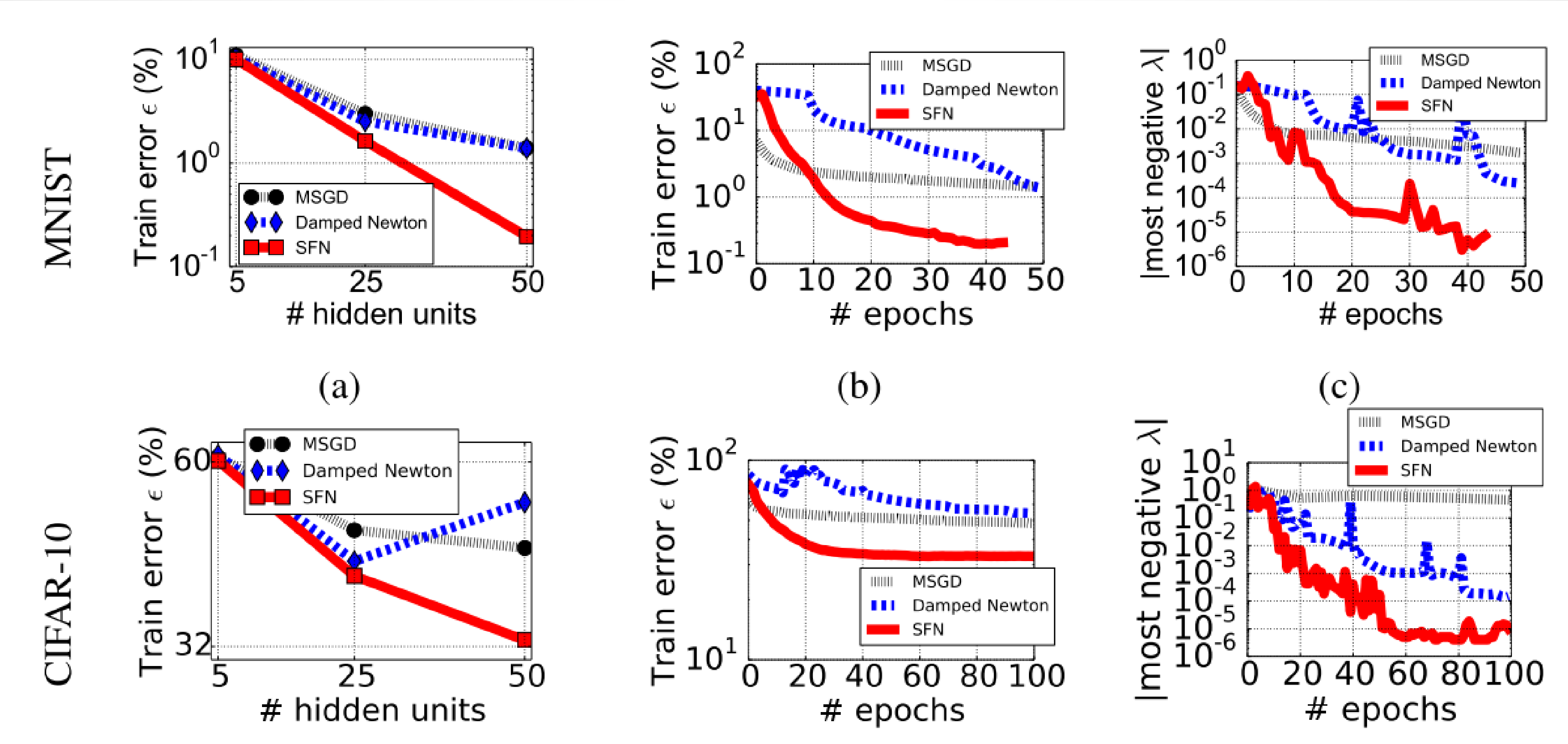
- A corrected Newton method where steps are proportional to $\dfrac{1}{|\lambda_i|}$
- First to justify this heuristic? (mathematically derived)

Y. Dauphin, … , and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization” NIPS, 2014.

(re)introduce a saddle-free Newton method.

- A corrected Newton method where steps are proportional to $\frac{1}{|\lambda_i|}$
- First to justify this heuristic? (mathematically derived)
- In practice optimize in a lower-dimensional Krylov subspace.

Y. Dauphin, … , and Y. Bengio, "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization" NIPS, 2014.



Y. Dauphin, … , and Y. Bengio, “Identifying and attacking the saddle point problem in high-dimensional non-convex optimization” NIPS, 2014.

Conclusion

- If deep networks are similar to spin-glass models we will need a better plan to optimize.
- A justifiable setting from which a saddle-free Newton method could be derived.

A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, and Y. LeCun, "The Loss Surfaces of Multilayer Networks," JMLR, 2015.

Make a connection between deep networks with ReLU and spherical spin-glass models.

A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, and Y. LeCun, "The Loss Surfaces of Multilayer Networks," JMLR, 2015.

Make a connection between deep networks with ReLU and spherical spin-glass models.

- Assuming (i) variable independence (ii) redundancy in network parametrization (iii) uniformity.
- Using absolute value loss and hinge loss.

A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, and Y. LeCun, "The Loss Surfaces of Multilayer Networks," JMLR, 2015.

- Make a connection between deep networks with ReLU and spherical spin-glass models.
  - Assuming (i) variable independence (ii) redundancy in network parametrization (iii) uniformity.
  - Using absolute value loss and hinge loss.
- In large networks we probably don't have to worry about bad local minima.

A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, and Y. LeCun, "The Loss Surfaces of Multilayer Networks," JMLR, 2015.

- Make a connection between deep networks with ReLU and spherical spin-glass models.
  - Assuming (i) variable independence (ii) redundancy in network parametrization (iii) uniformity.
  - Using absolute value loss and hinge loss.
- In large networks we probably don't have to worry about bad local minima.
- Finding a global minimum on the training set is not useful.

A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, and Y. LeCun, “The Loss Surfaces of Multilayer Networks,” JMLR, 2015.

Y. N. Dauphin, H. De Vries, and Y. Bengio, “Equilibrated adaptive learning rates for non-convex optimization,” in NIPS, 2015.

Following the same path from the last year.

Y. N. Dauphin, H. De Vries, and Y. Bengio, “Equilibrated adaptive learning rates for non-convex optimization,” in NIPS, 2015.

Following the same path from the last year.

In previous work used a low-dimensional Krylov subspace.

Y. N. Dauphin, H. De Vries, and Y. Bengio, "Equilibrated adaptive learning rates for non-convex optimization," in NIPS, 2015.

Following the same path from the last year.

In previous work used a low-dimensional Krylov subspace.

If limit ourselves to diagonal preconditioners can we get a similar conditioning as inverse Hessian with absolute eigenvalues?

Y. N. Dauphin, H. De Vries, and Y. Bengio, “Equilibrated adaptive learning rates for non-convex optimization,” in NIPS, 2015.

Following the same path from the last year.

In previous work used a low-dimensional Krylov subspace.

If limit ourselves to diagonal preconditioners can we get a similar conditioning as inverse Hessian with absolute eigenvalues?

Equilibrated Stochastic Gradient Descent!

Y. N. Dauphin, H. De Vries, and Y. Bengio, "Equilibrated adaptive learning rates for non-convex optimization," in NIPS, 2015.

Following the same path from the last year.

In previous work used a low-dimensional Krylov subspace.

If limit ourselves to diagonal preconditioners can we get a similar conditioning as inverse Hessian with absolute eigenvalues?

Equilibrated Stochastic Gradient Descent!

AdaDelta, AdaGrad, RMSProp.

Y. N. Dauphin, H. De Vries, and Y. Bengio, "Equilibrated adaptive learning rates for non-convex optimization," in NIPS, 2015.

Following the same path from the last year.

In previous work used a low-dimensional Krylov subspace.

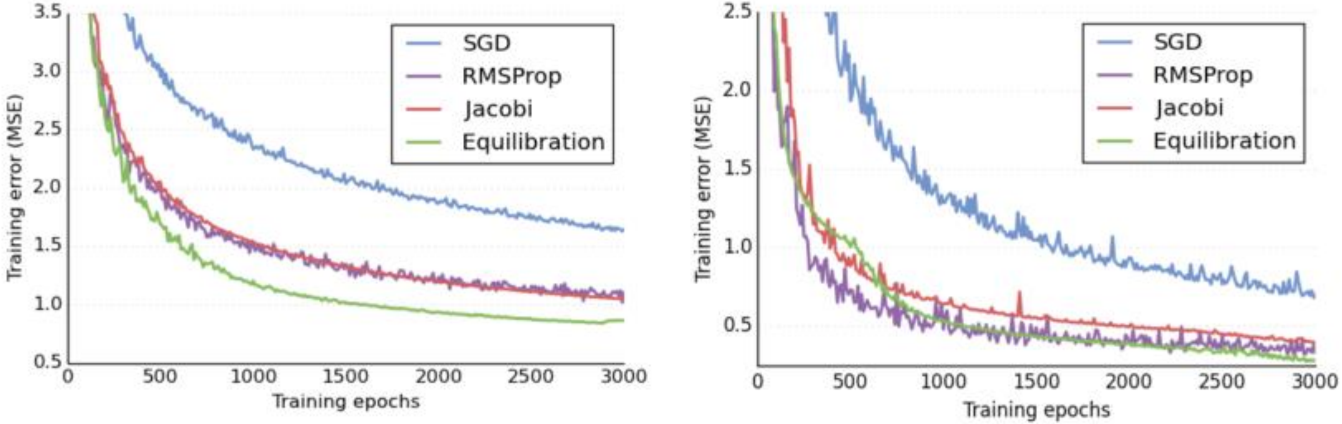
If limit ourselves to diagonal preconditioners can we get a similar conditioning as inverse Hessian with absolute eigenvalues?

Equilibrated Stochastic Gradient Descent!

AdaDelta, AdaGrad, RMSProp.

Works as well or better than RMSProp.

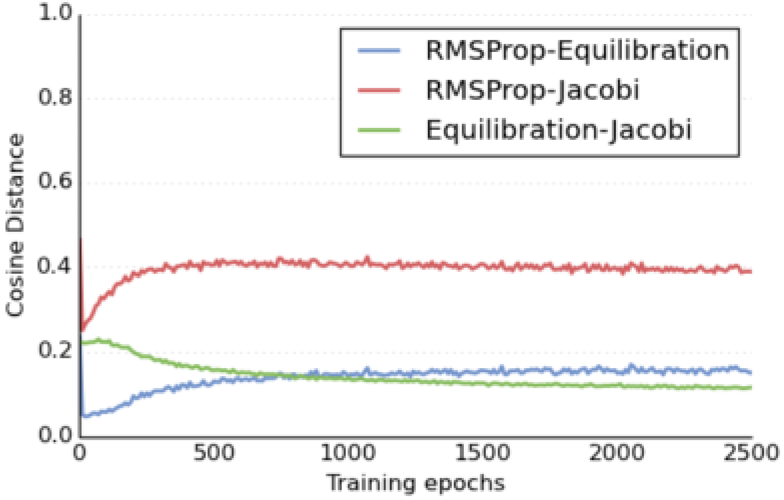
Y. N. Dauphin, H. De Vries, and Y. Bengio, “Equilibrated adaptive learning rates for non-convex optimization,” in NIPS, 2015.



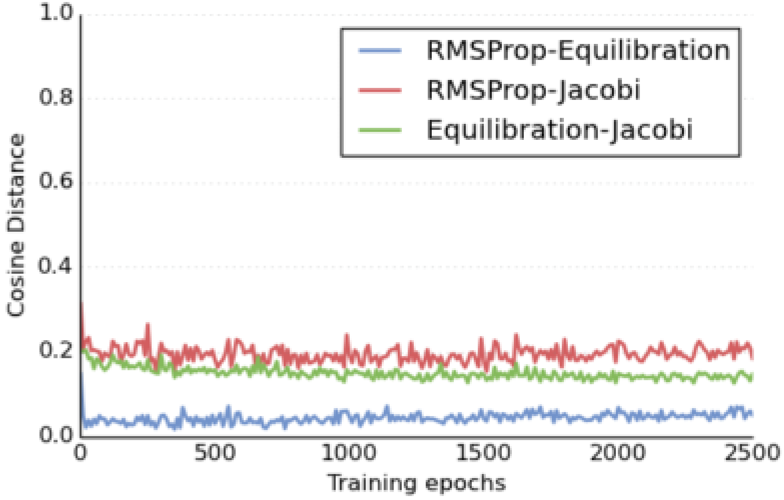
(a) MNIST (b) CURVES

Figure 3: Learning curves for deep auto-encoders on a) MNIST and b) CURVES comparing the different preconditioned SGD methods.

Y. N. Dauphin, H. De Vries, and Y. Bengio, “Equilibrated adaptive learning rates for non-convex optimization,” in NIPS, 2015.



(a) MNIST



(b) CURVES

Conclusion

The first group argue saddle points are an important problem in deep networks.

Conclusion

The first group argue saddle points are an important problem in deep networks.

The second group show as the network grows you probably do not need to worry about bad local minima. If you get past the saddle points, and settle for a lower index critical point, you’re probably in a good local minima.

Conclusion

The first group introduced an impractical saddle-free Newton method with limited experiments on arguably small networks.

Conclusion

The first group introduced an impractical saddle-free Newton method with limited experiments on arguably small networks.

Later they took an adaptive approach (diagonal preconditioning) and introduced ESGD which behaves similarly to RMSProp, an approach that is apparently not well understood.

Conclusion

- The second group made a direct connection to spin-glass models with unrealistic assumptions.
  - Input independent activation of ReLUs.
  - Independent inputs for each path on the network.

Conclusion

- The second group made a direct connection to spin-glass models with unrealistic assumptions.
  - Input independent activation of ReLUs.
  - Independent inputs for each path on the network.
- Working on ways to relax these assumptions?

Thank you!