Convergence of Sharpness-Aware Minimization

Peter Bartlett
Google Research and UC Berkeley

IHP
October 6, 2022

High-dimensional prediction with deep networks

Deep learning

- Deep learning has raised many interesting new questions

High-dimensional prediction with deep networks

Deep learning

- Deep learning has raised many interesting new questions
  - Efficient nonconvex optimization (empirical risk minimization with nonlinearly parameterized functions)

High-dimensional prediction with deep networks

Deep learning

- Deep learning has raised many interesting new questions
  - Efficient nonconvex optimization (empirical risk minimization with nonlinearly parameterized functions)
  - Good prediction despite overfitting and no explicit regularization

High-dimensional prediction with deep networks

Deep learning

- Deep learning has raised many interesting new questions
  - Efficient nonconvex optimization (empirical risk minimization with nonlinearly parameterized functions)
  - Good prediction despite overfitting and no explicit regularization
- Optimization methodology affects statistical performance

High-dimensional prediction with deep networks

Deep learning

- Deep learning has raised many interesting new questions
  - Efficient nonconvex optimization (empirical risk minimization with nonlinearly parameterized functions)
  - Good prediction despite overfitting and no explicit regularization
- Optimization methodology affects statistical performance
  - e.g., gradient flow motivates the study of minimum norm interpolation

High-dimensional prediction with deep networks

Deep learning

- Deep learning has raised many interesting new questions
  - Efficient nonconvex optimization (empirical risk minimization with nonlinearly parameterized functions)
  - Good prediction despite overfitting and no explicit regularization
- Optimization methodology affects statistical performance
  - e.g., gradient flow motivates the study of minimum norm interpolation
  - e.g., discrete time gradient descent and stochastic gradient descent as gradient flow on penalized losses

High-dimensional prediction with deep networks

Deep learning

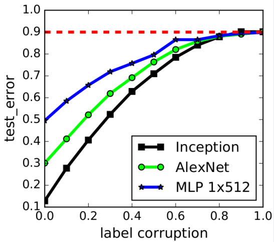
- Deep learning has raised many interesting new questions
  - Efficient nonconvex optimization (empirical risk minimization with nonlinearly parameterized functions)
  - Good prediction despite overfitting and no explicit regularization
- Optimization methodology affects statistical performance
  - e.g., gradient flow motivates the study of minimum norm interpolation
  - e.g., discrete time gradient descent and stochastic gradient descent as gradient flow on penalized losses
  - e.g., implicit regularization of gradient flow in neural networks

High-dimensional prediction with deep networks

Deep learning

- Deep learning has raised many interesting new questions
  - Efficient nonconvex optimization (empirical risk minimization with nonlinearly parameterized functions)
  - Good prediction despite overfitting and no explicit regularization
- Optimization methodology affects statistical performance
  - e.g., gradient flow motivates the study of minimum norm interpolation
  - e.g., discrete time gradient descent and stochastic gradient descent as gradient flow on penalized losses
  - e.g., implicit regularization of gradient flow in neural networks
- This talk: optimization for non-linear and high-dimensional prediction
  1. Benign overfitting in a non-linear setting
  2. 'Sharpness-Aware Minimization'

Overfitting in Deep Networks



- Deep networks can be trained to zero training error (for *regression* loss)
- ... with near state-of-the-art performance
- ... even for *noisy* problems.
- No tradeoff between fit to training data and complexity!
- *Benign overfitting.*

(Zhang, Bengio, Hardt, Recht, Vinyals, 2017)

also (Belkin, Hsu, Ma, Mandal, 2018)

Benign Overfitting

Intuition

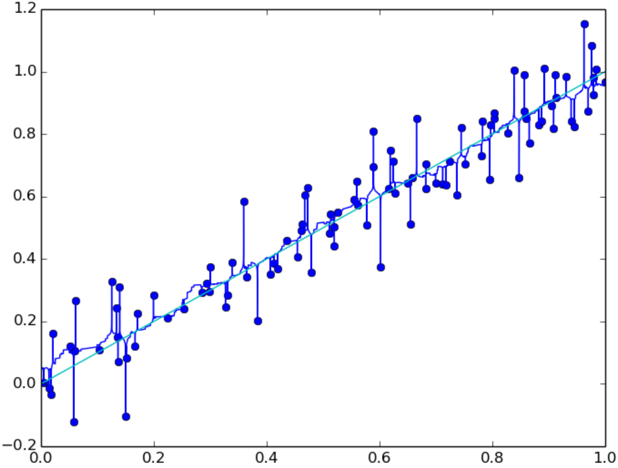
- Benign overfitting prediction rule $\widehat{f}$ decomposes as

$$\widehat{f} = \widehat{f}_0 + \Delta.$$

- $\widehat{f}_0$ = simple component useful for *prediction*.
- Δ = spiky component useful for *benign overfitting*.
- Classical statistical learning theory applies to $\widehat{f}_0$.
- Δ is not useful for prediction, but it is benign.

(Deep learning: a statistical viewpoint. B., Montanari, Rakhlin. *Acta Numerica*. 2021)

Benign Overfitting



Benign Overfitting

Linear Regression

(B, Long, Lugosi, Tsigler, 2019), (B, Tsigler, 2020)

- Benign overfitting prediction rule $\widehat{f}$ decomposes as

$$\widehat{f} = \widehat{f}_0 + \Delta.$$

- $\widehat{f}_0$ = *prediction* component:
  k^*-dim subspace corresponding to $\lambda_1, \ldots, \lambda_{k^*}$.
- Δ = *benign overfitting* component:
  orthogonal subspace. Δ is benign only if $R_{k^*} \gg n$.

Here,
$\lambda_1, \lambda_2, \ldots$ are the eigenvalues of the covariate covariance,
k^* is defined in terms of an effective rank of the covariance in the low-variance orthogonal subspace, and
R_{k^*} is another effective rank in that subspace.

Benign overfitting

- Benign overfitting in classical settings:
  - Kernel smoothing [Belkin, Hsu, Mitra, 2018; Belkin, Rakhlin, Tsybakov, 2018; Chhor, Sigalla, Tsybakov, 2022; . . .]
  - Linear regression [Hastie, Montanari, Rosset, Tibshirani, 2019; Bartlett, Long, Lugosi, Tsigler, 2019; Bartlett, Tsigler, 2020; Koehler, Zhou, Sutherland, Srebro, 2021; . . .]
  - Kernel regression [Liang, Rakhlin, 2018; Belkin, Hsu, Mitra, 2018; Mei, Montanari, 2019; Liang, Rakhlin, Zhai, 2020; Mei, Misiakiewicz, Montanari, 2021; . . .]
  - Logistic regression [Montanari, Ruan, Sohn, Yan, 2019; Liang, Sur, 2020; Chatterji, Long, 2021; Muthukumar, Narang, Subramanian, Belkin, Hsu, Sahai, 2021; . . .]

Benign overfitting

- Benign overfitting in classical settings:
  - Kernel smoothing [Belkin, Hsu, Mitra, 2018; Belkin, Rakhlin, Tsybakov, 2018; Chhor, Sigalla, Tsybakov, 2022; . . .]
  - Linear regression [Hastie, Montanari, Rosset, Tibshirani, 2019; Bartlett, Long, Lugosi, Tsigler, 2019; Bartlett, Tsigler, 2020; Koehler, Zhou, Sutherland, Srebro, 2021; . . .]
  - Kernel regression [Liang, Rakhlin, 2018; Belkin, Hsu, Mitra, 2018; Mei, Montanari, 2019; Liang, Rakhlin, Zhai, 2020; Mei, Misiakiewicz, Montanari, 2021; . . .]
  - Logistic regression [Montanari, Ruan, Sohn, Yan, 2019; Liang, Sur, 2020; Chatterji, Long, 2021; Muthukumar, Narang, Subramanian, Belkin, Hsu, Sahai, 2021; . . .]
- Benign overfitting in *neural networks*?
  (beyond the 'neural tangent kernel' approximation)

Benign overfitting

- Benign overfitting in classical settings:
  - Kernel smoothing [Belkin, Hsu, Mitra, 2018; Belkin, Rakhlin, Tsybakov, 2018; Chhor, Sigalla, Tsybakov, 2022; . . .]
  - Linear regression [Hastie, Montanari, Rosset, Tibshirani, 2019; Bartlett, Long, Lugosi, Tsigler, 2019; Bartlett, Tsigler, 2020; Koehler, Zhou, Sutherland, Srebro, 2021; . . .]
  - Kernel regression [Liang, Rakhlin, 2018; Belkin, Hsu, Mitra, 2018; Mei, Montanari, 2019; Liang, Rakhlin, Zhai, 2020; Mei, Misiakiewicz, Montanari, 2021; . . .]
  - Logistic regression [Montanari, Ruan, Sohn, Yan, 2019; Liang, Sur, 2020; Chatterji, Long, 2021; Muthukumar, Narang, Subramanian, Belkin, Hsu, Sahai, 2021; . . .]
- Benign overfitting in *neural networks*?
(beyond the 'neural tangent kernel' approximation)

Spencer Frei

Niladri Chatterji

Benign overfitting without linearity: neural network classifiers trained by gradient descent for noisy linear data. COLT 2022. arXiv:2202.05928

Benign overfitting without linearity

Spencer Frei

Niladri Chatterji

Benign overfitting without linearity: neural network classifiers trained by gradient descent for noisy linear data. COLT 2022. arXiv:2202.05928

Outline

- Noisy classification with two-layer neural networks trained by GD

Benign overfitting without linearity

Spencer Frei

Niladri Chatterji

Benign overfitting without linearity: neural network classifiers trained by gradient descent for noisy linear data. COLT 2022. arXiv:2202.05928

Outline

- Noisy classification with two-layer neural networks trained by GD
- Benign overfitting

Benign overfitting without linearity

Spencer Frei

Niladri Chatterji

Benign overfitting without linearity: neural network classifiers trained by gradient descent for noisy linear data. COLT 2022. arXiv:2202.05928

Outline

- Noisy classification with two-layer neural networks trained by GD
- Benign overfitting
- Proof ideas

Goal and technical challenges

Goal

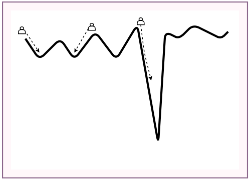
Understand how benign overfitting can occur in *neural networks trained by gradient descent* to get insight into 'modern' ML.

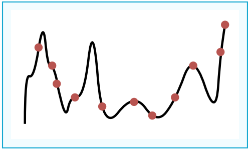
Goal and technical challenges

Goal

Understand how benign overfitting can occur in *neural networks trained by gradient descent* to get insight into 'modern' ML.

Technical challenges:





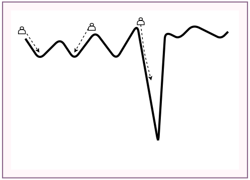
- Understand non-convex learning dynamics of neural network training.

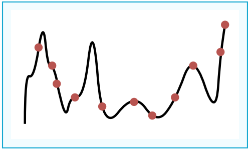
Goal and technical challenges

Goal

Understand how benign overfitting can occur in *neural networks trained by gradient descent* to get insight into 'modern' ML.

Technical challenges:





- Understand non-convex learning dynamics of neural network training.
- Understand generalization of interpolating classifiers for noisy data when hypothesis class has unbounded capacity.

Distributional setting

- Mixture of two log-concave isotropic clusters:
  - Cluster centered at $+\mu \in \mathbb{R}^p$, clean label $+1$
  - Cluster centered at $-\mu \in \mathbb{R}^p$, clean label -1
- Allow for constant fraction η of training labels to be flipped ($\tilde{\mathsf{P}}_{\mathsf{cl}}$: 'clean' distribution, P_{ns}: 'noisy' distribution)
- Assume $\|\mu\|$ grows with dimension p.

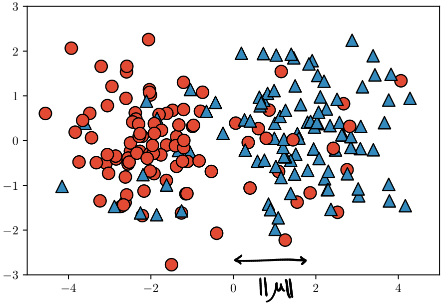
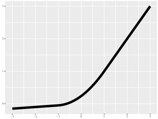


Figure: $\mathsf{P}_{\mathsf{clust}} = \mathsf{N}(0, I_2)$ with $\|\mu\| = 1.9$ and 15% of the labels flipped.

Model and optimization definitions

- We consider γ-*leaky,* H-*smooth activations* ϕ, satisfying for all $z \in \mathbb{R}$,

$$0 < \gamma \le \phi'(z) \le 1, \quad |\phi''(z)| \le H.$$

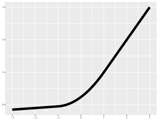


Two-layer neural networks trained by GD

Model and optimization definitions

- We consider γ*-leaky,* H*-smooth activations* ϕ, satisfying for all $z \in \mathbb{R}$,

$$0 < \gamma \leq \phi'(z) \leq 1, \quad |\phi''(z)| \leq H.$$



Two-layer neural networks trained by GD

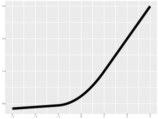
- Network with m neurons, first layer weights $W \in \mathbb{R}^{m \times p}$, second layer weights $\{a_j\}_{j=1}^m$ (fixed at initialization),

$$f(x; W) := \sum_{j=1}^m a_j \phi(\langle w_j, x \rangle).$$

Model and optimization definitions

- We consider γ*-leaky, H-smooth activations* ϕ, satisfying for all $z \in \mathbb{R}$,

$$0 < \gamma \leq \phi'(z) \leq 1, \quad |\phi''(z)| \leq H.$$



Two-layer neural networks trained by GD

- Network with m neurons, first layer weights $W \in \mathbb{R}^{m \times p}$, second layer weights $\{a_j\}_{j=1}^m$ (fixed at initialization),

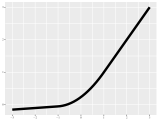
$$f(x; W) := \sum_{j=1}^m a_j \phi(\langle w_j, x \rangle).$$

- Initialize $[W^{(0)}]_{r,s} \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0, \omega_{\text{init}}^2)$, $a_j \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}(\{1/\sqrt{m}, -1/\sqrt{m}\})$.

Model and optimization definitions

- We consider γ-*leaky*, H-*smooth activations* ϕ, satisfying for all $z \in \mathbb{R}$,

$$0 < \gamma \le \phi'(z) \le 1, \quad |\phi''(z)| \le H.$$



Two-layer neural networks trained by GD

- Network with m neurons, first layer weights $W \in \mathbb{R}^{m \times p}$, second layer weights $\{a_j\}_{j=1}^m$ (fixed at initialization),

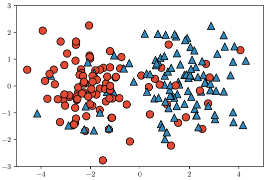
$$f(x; W) := \sum_{j=1}^m a_j \phi(\langle w_j, x \rangle).$$

- Initialize $[W^{(0)}]_{r,s} \overset{\text{i.i.d.}}{\sim} \mathrm{N}(0, \omega_{\text{init}}^2)$, $a_j \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}(\{1/\sqrt{m}, -1/\sqrt{m}\})$.
- For $\ell(z) = \log(1 + \exp(-z))$, data $\{(x_i, y_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} \mathrm{P}_{\text{ns}}$, $\alpha > 0$,

$$W^{(t+1)} = W^{(t)} - \alpha \nabla \widehat{L}(W^{(t)}) = W^{(t)} - \alpha \nabla \Big(\frac{1}{n} \sum_{i=1}^n \ell(y_i f(x_i; W^{(t)})) \Big).$$

The setting

For failure probability $\delta \in (0, 1)$, large $C > 1$:

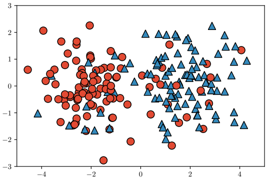


$\mathsf{P}_{\mathsf{clust}} = \mathsf{N}(0, I_2)$ with $\|\mu\| = 1.9$ and 15% of the labels flipped.

(A1) Number of samples $n \geq C \log(1/\delta)$.

The setting

For failure probability $\delta \in (0,1)$, large $C > 1$:



$\mathrm{P_{clust}} = \mathrm{N}(0, I_2)$ with $\|\mu\| = 1.9$ and 15% of the labels flipped.

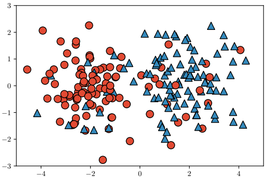
(A1) Number of samples $n \geq C \log(1/\delta)$.

(A2) Mean separation $\|\mu\| = \Theta(p^{\frac{1}{3}})$.

- Holds for more general $\|\mu\| = \omega_p(1)$.

The setting

For failure probability $\delta \in (0, 1)$, large $C > 1$:



$P_{\text{clust}} = N(0, I_2)$ with $\|\mu\| = 1.9$ and 15% of the labels flipped.

(A1) Number of samples $n \geq C \log(1/\delta)$.

(A2) Mean separation $\|\mu\| = \Theta(p^{\frac{1}{3}})$.

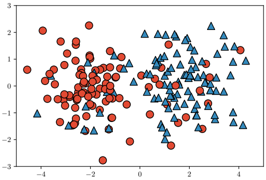
- Holds for more general $\|\mu\| = \omega_p(1)$.

(A3) Dimension $p \gtrsim n^3$.

- Ensures all samples are $\approx$ orthogonal.

The setting

For failure probability $\delta \in (0,1)$, large $C > 1$:



$P_{\text{clust}} = N(0, I_2)$ with $\|\mu\| = 1.9$ and 15% of the labels flipped.

(A1) Number of samples $n \geq C \log(1/\delta)$.

(A2) Mean separation $\|\mu\| = \Theta(p^{\frac{1}{3}})$.

- Holds for more general $\|\mu\| = \omega_p(1)$.

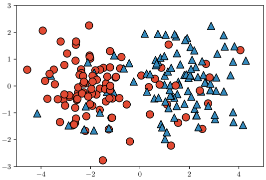
(A3) Dimension $p \gtrsim n^3$.

- Ensures all samples are $\approx$ orthogonal.

(A4) Noise rate $\eta \leq 1/C$.

The setting

For failure probability $\delta \in (0, 1)$, large $C > 1$:



$P_{\text{clust}} = N(0, I_2)$ with $\|\mu\| = 1.9$ and 15% of the labels flipped.

(A1) Number of samples $n \geq C \log(1/\delta)$.

(A2) Mean separation $\|\mu\| = \Theta(p^{\frac{1}{3}})$.

- Holds for more general $\|\mu\| = \omega_p(1)$.

(A3) Dimension $p \gtrsim n^3$.

- Ensures all samples are $\approx$ orthogonal.

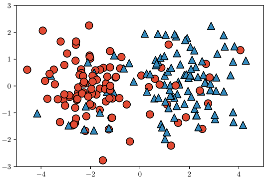
(A4) Noise rate $\eta \leq 1/C$.

(A5) Large step-size relative to initialization: $\alpha \geq \omega_{\text{init}} \sqrt{mp}$.

- Ensures 'feature-learning' (non-NTK) after one step.

The setting

For failure probability $\delta \in (0,1)$, large $C > 1$:



$\mathsf{P}_{\text{clust}} = \mathsf{N}(0, I_2)$ with $\|\mu\| = 1.9$ and 15% of the labels flipped.

(A1) Number of samples $n \geq C \log(1/\delta)$.

(A2) Mean separation $\|\mu\| = \Theta(p^{\frac{1}{3}})$.

- Holds for more general $\|\mu\| = \omega_p(1)$.

(A3) Dimension $p \gtrsim n^3$.

- Ensures all samples are $\approx$ orthogonal.

(A4) Noise rate $\eta \leq 1/C$.

(A5) Large step-size relative to initialization: $\alpha \geq \omega_{\text{init}} \sqrt{mp}$.

- Ensures 'feature-learning' (non-NTK) after one step.

- Networks of arbitrary width $m \geq 1$.

Benign overfitting in neural networks trained by GD

For $C > 1$ large enough under Assumptions (A1) through (A5):

Theorem (Frei, Chatterji, B, 2022)

For $0 < \varepsilon < 1/2n$, by running GD with stepsize α, for $T \geq C\alpha^{-1}\varepsilon^{-2}$ iterations, with high probability over the random initialization and sample:

Benign overfitting in neural networks trained by GD

For $C > 1$ large enough under Assumptions (A1) through (A5):

Theorem (Frei, Chatterji, B, 2022)

For $0 < \varepsilon < 1/2n$, by running GD with stepsize α, for $T \geq C\alpha^{-1}\varepsilon^{-2}$ iterations, with high probability over the random initialization and sample:

1. $y_i = \mathrm{sgn}\big(f(x_i; W^{(T)})\big)$ for all i with $\boxed{\text{training loss } \widehat{L}(W^{(T)}) \leq \varepsilon}$.

Benign overfitting in neural networks trained by GD

For $C > 1$ large enough under Assumptions (A1) through (A5):

Theorem (Frei, Chatterji, B, 2022)

For $0 < \varepsilon < 1/2n$, by running GD with stepsize α, for $T \geq C\alpha^{-1}\varepsilon^{-2}$ iterations, with high probability over the random initialization and sample:

1. $y_i = \operatorname{sgn}(f(x_i; W^{(T)}))$ for all i with $\boxed{\text{training loss } \widehat{L}(W^{(T)}) \leq \varepsilon}$.
2. The test error satisfies

$$\mathbb{P}_{(x,y)\sim \mathsf{P}_{\mathrm{ns}}}\left[y \neq \operatorname{sgn}(f(x; W^{(T)}))\right] \leq \eta + \boxed{2\exp\left(-c \cdot np^{\frac{1}{3}}\right)}.$$

Benign overfitting in neural networks trained by GD

For $C > 1$ large enough under Assumptions (A1) through (A5):

Theorem (Frei, Chatterji, B, 2022)

For $0 < \varepsilon < 1/2n$, by running GD with stepsize α, for $T \geq C\alpha^{-1}\varepsilon^{-2}$ iterations, with high probability over the random initialization and sample:

1. $y_i = \mathrm{sgn}(f(x_i; W^{(T)}))$ for all i with $\boxed{\text{training loss } \widehat{L}(W^{(T)}) \leq \varepsilon}$.
2. The test error satisfies

$$\mathbb{P}_{(x,y)\sim \mathsf{P}_{\mathrm{ns}}}\left[y \neq \mathrm{sgn}(f(x; W^{(T)}))\right] \leq \eta + \boxed{2\exp\left(-c \cdot np^{\frac{1}{3}}\right)}.$$

- Training error is ≈ 0 with noisy labels $\boxed{\text{(overfitting)}}$, yet still generalizing near Bayes-optimal $\boxed{\text{(benign)}}$.

Benign overfitting in neural networks trained by GD

For $C > 1$ large enough under Assumptions (A1) through (A5):

Theorem

(Frei, Chatterji, B, 2022)

For $0 < \varepsilon < 1/2n$, by running GD with stepsize α, for $T \geq C\alpha^{-1}\varepsilon^{-2}$ iterations, with high probability over the random initialization and sample:

1. $y_i = \mathrm{sgn}(f(x_i; W^{(T)}))$ for all i with $\boxed{\text{training loss } \widehat{L}(W^{(T)}) \leq \varepsilon}$.
2. The test error satisfies

$$\mathbb{P}_{(x,y)\sim \mathsf{P}_{\mathrm{ns}}}\left[y \neq \mathrm{sgn}(f(x; W^{(T)}))\right] \leq \eta + \boxed{2\exp\left(-c \cdot np^{\frac{1}{3}}\right)}.$$

- Training error is ≈ 0 with noisy labels $\boxed{\text{(overfitting)}}$, yet still generalizing near Bayes-optimal $\boxed{\text{(benign)}}$.
- *Any* width $m \geq 1$: no dependence on m (except $\alpha \geq \omega_{\mathrm{init}}\sqrt{mp}$).

Benign overfitting and uniform convergence

Theorem (Frei, Chatterji, B, 2022)

For $0 < \varepsilon < 1/2n$, by running GD with l.r. α, for $T \geq C\alpha^{-1}\varepsilon^{-2}$ iterations, w.h.p. over the random initialization and sample:

1. $y_i = \operatorname{sgn}(f(x_i; W^{(T)}))$ for all i with $\boxed{\text{training loss } \widehat{L}(W^{(T)}) \leq \varepsilon}$.
2. The test error satisfies

$$\mathbb{P}_{(x,y)\sim \mathsf{P}_{\mathrm{ns}}}\left[y \neq \operatorname{sgn}(f(x; W^{(T)}))\right] \leq \eta + \boxed{2\exp\left(-c \cdot np^{\frac{1}{3}}\right)}.$$

Benign overfitting and uniform convergence

Theorem (Frei, Chatterji, B, 2022)

For $0 < \varepsilon < 1/2n$, by running GD with l.r. α, for $T \geq C\alpha^{-1}\varepsilon^{-2}$ iterations, w.h.p. over the random initialization and sample:

1. $y_i = \operatorname{sgn}(f(x_i; W^{(T)}))$ for all i with $\boxed{\text{training loss } \widehat{L}(W^{(T)}) \leq \varepsilon}$.
2. The test error satisfies

$$\mathbb{P}_{(x,y)\sim \mathsf{P}_{\mathrm{ns}}}\left[y \neq \operatorname{sgn}(f(x; W^{(T)}))\right] \leq \eta + \boxed{2\exp\left(-c \cdot np^{\frac{1}{3}}\right)}.$$

- As $\varepsilon \to 0$, $\boxed{\|W^{(T)}\| \to \infty}$.

Benign overfitting and uniform convergence

Theorem (Frei, Chatterji, B, 2022)

For $0 < \varepsilon < 1/2n$, by running GD with l.r. α, for $T \geq C\alpha^{-1}\varepsilon^{-2}$ iterations, w.h.p. over the random initialization and sample:

1. $y_i = \operatorname{sgn}(f(x_i; W^{(T)}))$ for all i with $\boxed{\text{training loss } \widehat{L}(W^{(T)}) \leq \varepsilon}$.
2. The test error satisfies

$$\mathbb{P}_{(x,y)\sim \mathsf{P}_{\mathrm{ns}}}\left[y \neq \operatorname{sgn}(f(x; W^{(T)}))\right] \leq \eta + \boxed{2\exp\left(-c \cdot np^{\frac{1}{3}}\right)}.$$

- As $\varepsilon \to 0$, $\boxed{\|W^{(T)}\| \to \infty}$.
- Predictor has **unbounded norm**, neural net can be **arbitrarily wide**, $\boxed{\text{achieves} \approx 0 \text{ training loss}}$, $\boxed{\text{generalizes near-optimally}}$—Bayes error $\geq \eta = \Omega(1)$.
  - Many ways to overfit: $p \gg n$, width $\gg 1$, ...

Proof outline

By strong log-concavity, suffices to derive normalized margin bound:

Lemma

Suppose that $\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathsf{P}}_{\mathsf{cl}}}[\tilde{y}f(x;W)] \geq 0$. *Then there exists a universal constant* $c > 0$ *such that*

$$\mathbb{P}_{(x,y)\sim\mathsf{P}_{\mathsf{ns}}}\left(y \neq \operatorname{sgn}(f(x;W))\right) \leq \eta + 2\exp\left(-c\left(\frac{\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathsf{P}}_{\mathsf{cl}}}[\tilde{y}f(x;W)]}{\|W\|_F}\right)^2\right)$$

Proof outline

By strong log-concavity, suffices to derive normalized margin bound:

Lemma

Suppose that $\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathsf{P}}_{\mathrm{cl}}}[\tilde{y}f(x;W)] \geq 0$. *Then there exists a universal constant* $c > 0$ *such that*

$$\mathbb{P}_{(x,y)\sim\mathsf{P}_{\mathrm{ns}}}\left(y \neq \mathrm{sgn}(f(x;W))\right) \leq \eta + 2\exp\left(-c\left(\frac{\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathsf{P}}_{\mathrm{cl}}}[\tilde{y}f(x;W)]}{\|W\|_F}\right)^2\right)$$

- Benign overfitting occurs if we can show:
  1. Normalized margin on *clean* points is large:

$$\frac{\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathsf{P}}_{\mathrm{cl}}}[\tilde{y}f(x;W^{(T)})]}{\|W^{(T)}\|_F} \gg 0.$$

Proof outline

By strong log-concavity, suffices to derive normalized margin bound:

Lemma

Suppose that $\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathsf{P}}_{\mathsf{cl}}}[\tilde{y}f(x;W)] \geq 0$*. Then there exists a universal constant* $c > 0$ *such that*

$$\mathbb{P}_{(x,y)\sim\mathsf{P}_{\mathsf{ns}}}\big(y \neq \mathrm{sgn}(f(x;W))\big) \leq \eta + 2\exp\left(-c\left(\frac{\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathsf{P}}_{\mathsf{cl}}}[\tilde{y}f(x;W)]}{\|W\|_F}\right)^2\right)$$

- Benign overfitting occurs if we can show:
  1. Normalized margin on *clean* points is large:
  $$\frac{\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathsf{P}}_{\mathsf{cl}}}[\tilde{y}f(x;W^{(T)})]}{\|W^{(T)}\|_F} \gg 0.$$
  2. Empirical risk can be driven to zero:
  $$y_i = \mathrm{sgn}\big(f(x_i;W^{(T)})\big) \text{ for all } i, \quad \text{and} \quad \widehat{L}(W^{(T)}) \approx 0.$$

Gradient descent ensures good generalization performance

Lemma

For any $t \geq 1$, for a step size large relative to random initialization,

$$\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathsf{P}}_{\mathsf{cl}}}\left[\frac{\tilde{y}f(x;W^{(t)})}{\|W^{(t)}\|_F}\right] \gtrsim \sqrt{np^{1/3}} \gg 0,$$

$$\mathbb{P}_{(x,y)\sim\mathsf{P}_{\mathsf{ns}}}(y \neq \mathrm{sgn}(f(x;W^{(t)}))) \leq \eta + 2\exp\left(-c\cdot np^{1/3}\right).$$

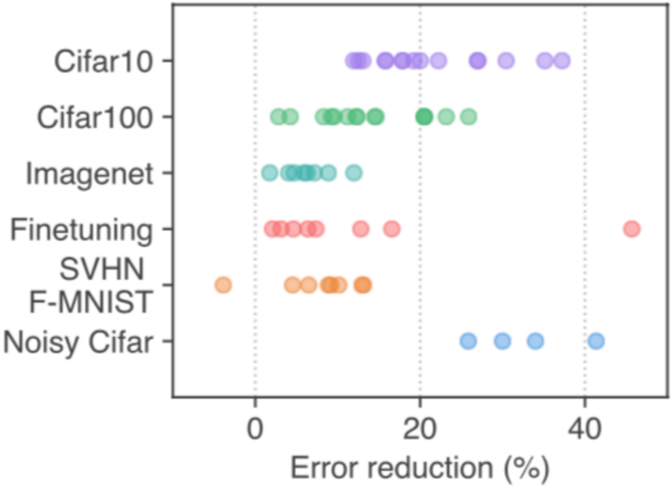
- Gradient descent produces a particular neural network which will classify well, regardless of $\|W^{(t)}\|_F$, with sub-polynomial samples.

Outline

Optimization for high-dimensional prediction

1. Benign overfitting in a non-linear setting
2. ‘Sharpness-Aware Minimization’

Sharpness-Aware Minimization: Prediction Performance



Foret, Kleiner, Mobahi, Neyshabur. 2021

Sharpness-Aware Minimization

Sharpness-Aware Minimization for Efficiently Improving Generalization.
Pierre Foret, Ariel Kleiner, Hossein Mobahi, Behnam Neyshabur. ICLR21.

Sharpness-Aware Minimization

Sharpness-Aware Minimization for Efficiently Improving Generalization.
Pierre Foret, Ariel Kleiner, Hossein Mobahi, Behnam Neyshabur. ICLR21.

- The story: For an empirical loss ℓ defined on a parameter space: $\min_w \max_{\|\epsilon\| \leq \rho} \ell(w + \epsilon)$.

Sharpness-Aware Minimization

Sharpness-Aware Minimization for Efficiently Improving Generalization.
Pierre Foret, Ariel Kleiner, Hossein Mobahi, Behnam Neyshabur. ICLR21.

- The story: For an empirical loss ℓ defined on a parameter space: $\min_w \max_{\|\epsilon\| \leq \rho} \ell(w + \epsilon)$.
- The rationale:

$$\max_{\|\epsilon\| \leq \rho} \ell(w + \epsilon) = \underbrace{\max_{\|\epsilon\| \leq \rho} \ell(w + \epsilon) - \ell(w)}_{\textit{sharpness}} + \ell(w).$$

Sharpness-Aware Minimization

Sharpness-Aware Minimization for Efficiently Improving Generalization.
Pierre Foret, Ariel Kleiner, Hossein Mobahi, Behnam Neyshabur. ICLR21.

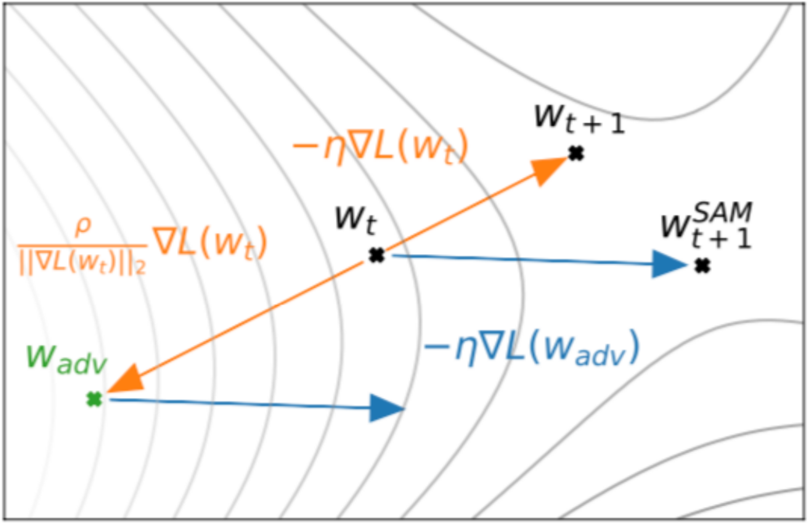
- The story: For an empirical loss ℓ defined on a parameter space: $\min_w \max_{\|\epsilon\| \leq \rho} \ell(w+\epsilon)$.
- The rationale:

$$\max_{\|\epsilon\| \leq \rho} \ell(w+\epsilon) = \underbrace{\max_{\|\epsilon\| \leq \rho} \ell(w+\epsilon) - \ell(w)}_{\textit{sharpness}} + \ell(w).$$

- The reality: First order simplification:

$$w_{t+1} = w_t - \eta \nabla \ell \left(w_t + \rho \frac{\nabla \ell(w_t)}{\|\nabla \ell(w_t)\|} \right).$$

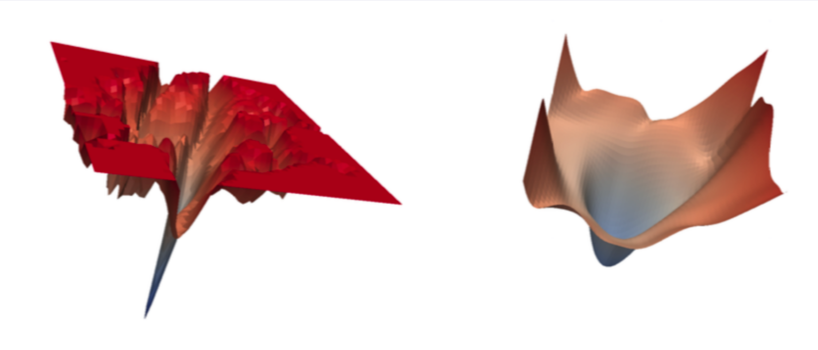
Sharpness-Aware Minimization



Foret, Kleiner, Mobahi, Neyshabur. 2021

Visualizing SAM Minima

ResNet trained with SGD versus SAM



Foret, Kleiner, Mobahi, Neyshabur. 2021

Convergence of Sharpness-Aware Minimization

Phil Long

Olivier Bousquet

- The dynamics of sharpness-aware minimization: bouncing across ravines and drifting towards wide minima. B., Long, Bousquet. arXiv:2210.xxxxx

Outline

Convergence of Sharpness-Aware Minimization

Phil Long

Olivier Bousquet

- The dynamics of sharpness-aware minimization: bouncing across ravines and drifting towards wide minima. B., Long, Bousquet. arXiv:2210.xxxxx

Outline

- SAM with a quadratic criterion: Bouncing across ravines
  - Stationary points
  - A non-convex gradient descent
  - SAM oscillates around minimum

Convergence of Sharpness-Aware Minimization

Phil Long

Olivier Bousquet

- The dynamics of sharpness-aware minimization: bouncing across ravines and drifting towards wide minima. B., Long, Bousquet. arXiv:2210.xxxxx

Outline

- SAM with a quadratic criterion: Bouncing across ravines
  - Stationary points
  - A non-convex gradient descent
  - SAM oscillates around minimum
- Beyond quadratic: Drifting towards wide minima
  - SAM near a smooth minimum
  - Descending the gradient of the spectral norm of the Hessian

Convergence of Sharpness-Aware Minimization

Phil Long

Olivier Bousquet

- The dynamics of sharpness-aware minimization: bouncing across ravines and drifting towards wide minima. B., Long, Bousquet. arXiv:2210.xxxxx

Outline

- SAM with a quadratic criterion: Bouncing across ravines
    - Stationary points
    - A non-convex gradient descent
    - SAM oscillates around minimum
- Beyond quadratic: Drifting towards wide minima
    - SAM near a smooth minimum
    - Descending the gradient of the spectral norm of the Hessian
- Open questions

SAM with a quadratic criterion

SAM

For a loss function $\ell : \mathbb{R}^d \to \mathbb{R}$, SAM starts with an initial parameter vector $w_0 \in \mathbb{R}^d$ and updates

$$w_{t+1} = w_t - \eta \nabla \ell \left(w_t + \rho \frac{\nabla \ell(w_t)}{\|\nabla \ell(w_t)\|} \right).$$

where $\eta, \rho > 0$ are step size parameters.

SAM with a quadratic criterion

SAM

For a loss function $\ell : \mathbb{R}^d \to \mathbb{R}$, SAM starts with an initial parameter vector $w_0 \in \mathbb{R}^d$ and updates

$$w_{t+1} = w_t - \eta \nabla \ell \left(w_t + \rho \frac{\nabla \ell(w_t)}{\|\nabla \ell(w_t)\|} \right).$$

where $\eta, \rho > 0$ are step size parameters.

SAM with quadratic loss

Fix $\Lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_d)$ with $\lambda_1 \geq \cdots \lambda_d \geq 0$ and consider loss

$$\ell(w) = \frac{1}{2} w^\top \Lambda w.$$

Then $\nabla \ell(w) = \Lambda w$ and $w_{t+1} = \left(I - \eta \Lambda - \frac{\eta \rho}{\|\Lambda w_t\|} \Lambda^2 \right) w_t$.

Bouncing across ravines

Theorem

(B., Long, Bousquet, 2022)

There is an absolute constant c such that for any eigenvalues $\lambda_1 > \lambda_2 \geq ... \geq \lambda_d > 0$, any neighborhood size $\rho > 0$, and any step size $0 < \eta < \frac{1}{2\lambda_1}$, for all small enough $\epsilon, \delta > 0$, if w_0 is sampled from a continuous probability distribution over $\mathbb{R}^d$ (density bounded above by A; $\|w_0\|$ not too big; $|w_{0,1}|$ not too small), then with probability $1 - \delta$, for all t sufficiently large (polynomial in d, $1/(\eta\lambda_d)$, λ_1/λ_d and $1/(\lambda_1^2/\lambda_2^2 - 1)$, polylogarithmic in other parameters), for some

$$w^* \in \left\{ \pm \frac{\eta\rho\lambda_1}{2 - \eta\lambda_1} e_1 \right\}$$

and for all $s \geq t$, $\|w_{2s} - w^*\| \leq \epsilon$ and $\|w_{2s+1} + w^*\| \leq \epsilon$.

SAM with a quadratic criterion

A reparameterization

Define $v_t = \nabla \ell(w_t) = \Lambda w_t$. Then

$$v_{t+1} = \left(I - \eta\Lambda - \frac{\eta\rho}{\|v_t\|}\Lambda^2 \right) v_t,$$

SAM with a quadratic criterion

A reparameterization

Define $v_t = \nabla \ell(w_t) = \Lambda w_t$. Then

$$v_{t+1} = \left(I - \eta \Lambda - \frac{\eta \rho}{\|v_t\|} \Lambda^2 \right) v_t,$$

so, for all i and all t, we have

$$\begin{aligned} v_{t+1,i} &= \left(1 - \eta \lambda_i - \frac{\eta \rho \lambda_i^2}{\|v_t\|} \right) v_{t,i} \\ &= (1 - \eta \lambda_i) \left(1 - \frac{\gamma_i}{\|v_t\|} \right) v_{t,i}, \end{aligned}$$

where $\gamma_i := \dfrac{\eta \rho \lambda_i^2}{1 - \eta \lambda_i}$.

SAM with a quadratic criterion

A reparameterization

Define $v_t = \nabla \ell(w_t) = \Lambda w_t$. Then

$$v_{t+1} = \left(I - \eta\Lambda - \frac{\eta\rho}{\|v_t\|}\Lambda^2 \right) v_t,$$

so, for all i and all t, we have

$$\begin{aligned} v_{t+1,i} &= \left(1 - \eta\lambda_i - \frac{\eta\rho\lambda_i^2}{\|v_t\|} \right) v_{t,i} \\ &= (1 - \eta\lambda_i) \left(1 - \frac{\gamma_i}{\|v_t\|} \right) v_{t,i}, \end{aligned}$$

where $\gamma_i := \dfrac{\eta\rho\lambda_i^2}{1 - \eta\lambda_i}$.

Nonlinear recurrence, but coupled only by $\|v_t\|$.

SAM with a quadratic criterion

Define $\beta_i = \frac{1 - \eta\lambda_i}{2 - \eta\lambda_i}\gamma_i = \frac{\eta\rho\lambda_i^2}{2 - \eta\lambda_i}$.

Solutions are in the eigenvector directions, β_i from the minimum

The set of non-zero solutions $(v_1^2, \ldots, v_d^2)$ to $\forall i,\ v_{t+1,i}^2 = v_{t,i}^2$ is

$$\bigcup_{i=1}^{d} \mathrm{co}\{\beta_i^2 e_j : \beta_j = \beta_i\},$$

where $\mathrm{co}(S)$ denotes the convex hull of a set S and e_j is the jth basis vector in $\mathbb{R}^d$.

SAM with a quadratic criterion

Define $\alpha_i = \dfrac{(1-\eta\lambda_1)\gamma_1 + (1-\eta\lambda_i)\gamma_i}{1-\eta\lambda_1 + 1-\eta\lambda_i}$.

Recall $\beta_i = \dfrac{1-\eta\lambda_i}{2-\eta\lambda_i}\gamma_i$.

SAM with a quadratic criterion

Define $\alpha_i = \dfrac{(1-\eta\lambda_1)\gamma_1 + (1-\eta\lambda_i)\gamma_i}{1-\eta\lambda_1 + 1 - \eta\lambda_i}$. Recall $\beta_i = \dfrac{1-\eta\lambda_i}{2-\eta\lambda_i}\gamma_i$.

If $\lambda_1 > \lambda_2$, then $\beta_d \leq \cdots \leq \beta_1 < \alpha_d \leq \cdots \alpha_2 \leq \alpha_1 = \gamma_1$.

SAM with a quadratic criterion

Define $\alpha_i = \dfrac{(1-\eta\lambda_1)\gamma_1 + (1-\eta\lambda_i)\gamma_i}{1-\eta\lambda_1 + 1-\eta\lambda_i}$. Recall $\beta_i = \dfrac{1-\eta\lambda_i}{2-\eta\lambda_i}\gamma_i$.

If $\lambda_1 > \lambda_2$, then $\beta_d \le \cdots \le \beta_1 < \alpha_d \le \cdots \alpha_2 \le \alpha_1 = \gamma_1$.

Norm of v versus β_i determines how components grow

$\|v_t\| > \beta_i$ iff $v_{t+1,i}^2 < v_{t,i}^2$.

SAM with a quadratic criterion

Define $\alpha_i = \frac{(1-\eta\lambda_1)\gamma_1 + (1-\eta\lambda_i)\gamma_i}{1-\eta\lambda_1 + 1 - \eta\lambda_i}$. Recall $\beta_i = \frac{1-\eta\lambda_i}{2-\eta\lambda_i}\gamma_i$.

If $\lambda_1 > \lambda_2$, then $\beta_d \leq \cdots \leq \beta_1 < \alpha_d \leq \cdots \alpha_2 \leq \alpha_1 = \gamma_1$.

Norm of v versus β_i determines how components grow

$\|v_t\| > \beta_i$ iff $v_{t+1,i}^2 < v_{t,i}^2$.

Norm of v versus α_i determines relative growth

If $\lambda_1 > \lambda_2$, then for $i \in \{2, \ldots, d\}$, $\|v_t\| < \alpha_i$ iff $\frac{v_{t+1,1}^2}{v_{t+1,i}^2} > \frac{v_{t,1}^2}{v_{t,i}^2}$.

SAM with a quadratic criterion

Define $\alpha_i = \dfrac{(1-\eta\lambda_1)\gamma_1 + (1-\eta\lambda_i)\gamma_i}{1-\eta\lambda_1 + 1-\eta\lambda_i}$.

Recall $\beta_i = \dfrac{1-\eta\lambda_i}{2-\eta\lambda_i}\gamma_i$.

If $\lambda_1 > \lambda_2$, then $\beta_d \leq \cdots \leq \beta_1 < \alpha_d \leq \cdots \alpha_2 \leq \alpha_1 = \gamma_1$.

Norm of v versus β_i determines how components grow

$\|v_t\| > \beta_i$ iff $v_{t+1,i}^2 < v_{t,i}^2$.

Norm of v versus α_i determines relative growth

If $\lambda_1 > \lambda_2$, then for $i \in \{2, \ldots, d\}$, $\|v_t\| < \alpha_i$ iff $\frac{v_{t+1,1}^2}{v_{t+1,i}^2} > \frac{v_{t,1}^2}{v_{t,i}^2}$.

Define $b = (1-\eta\lambda_1)\gamma_1$.

$\|v_t\| \leq b$ implies $\|v_{t+1}\| \leq b$ (and the decay to b is exponentially fast).

A non-convex gradient descent

Lemma

For $u_t := (-1)^t w_t$, if $\|w_t\| > 0$,

$$u_{t+1} = u_t - \eta\rho\nabla J(u_t),$$

A non-convex gradient descent

Lemma

For $u_t := (-1)^t w_t$, if $\|w_t\| > 0$,

$$u_{t+1} = u_t - \eta\rho\nabla J(u_t),$$

where

$$J(u) = \frac{1}{2}u^\top Cu - \|\Lambda u\|, \qquad C = \mathrm{diag}\left(\frac{\lambda_1^2}{\beta_1}, \ldots, \frac{\lambda_d^2}{\beta_d}\right).$$

A non-convex gradient descent

Lemma

For $u_t := (-1)^t w_t$, if $\|w_t\| > 0$,

$$u_{t+1} = u_t - \eta\rho\nabla J(u_t),$$

where

$$J(u) = \frac{1}{2}u^\top C u - \|\Lambda u\|, \qquad C = \operatorname{diag}\left(\frac{\lambda_1^2}{\beta_1}, \ldots, \frac{\lambda_d^2}{\beta_d}\right).$$

Also,

$$J(u_{t+1}) - J(u_t) \leq -\frac{1}{2\rho}\sum_{i=1}^{d} u_{t,i}^2 \left(1 - \frac{\beta_i}{\|\Lambda u_t\|}\right)^2 (2 - \eta\lambda_i)^2 \lambda_i.$$

A non-convex gradient descent

Properties of J

$\nabla J(u) = 0$ iff for some i, $\|u\| = \beta_i/\lambda_i$ and $u \in \mathrm{span}\{e_j : \beta_j = \beta_i\}$.

A non-convex gradient descent

Properties of J

$\nabla J(u) = 0$ iff for some i, $\|u\| = \beta_i/\lambda_i$ and $u \in \mathrm{span}\{e_j : \beta_j = \beta_i\}$.

For unit norm $\widehat{u}$ satisfying $\nabla J(\beta_i/\lambda_i \widehat{u}) = 0$,

$$\nabla^2 J\left(\frac{\beta_i}{\lambda_i}\widehat{u}\right) = \Lambda^2 \left(\sum_{j:\beta_j \neq \beta_i} \left(\frac{1}{\beta_j} - \frac{1}{\beta_i}\right) e_j e_j^\top + \frac{1}{\beta_i}\widehat{u}\widehat{u}^\top \right),$$

which has $|\{j : \beta_j < \beta_i\}| + 1$ positive eigenvalues, $|\{j : \beta_j > \beta_i\}|$ negative eigenvalues, and $|\{j : \beta_j = \beta_i\}| - 1$ zero eigenvalues.

A non-convex gradient descent

Properties of J

$\nabla J(u) = 0$ iff for some i, $\|u\| = \beta_i/\lambda_i$ and $u \in \mathrm{span}\{e_j : \beta_j = \beta_i\}$.

For unit norm $\widehat{u}$ satisfying $\nabla J(\beta_i/\lambda_i \widehat{u}) = 0$,

$$\nabla^2 J\left(\frac{\beta_i}{\lambda_i}\widehat{u}\right) = \Lambda^2 \left(\sum_{j:\beta_j \neq \beta_i} \left(\frac{1}{\beta_j} - \frac{1}{\beta_i}\right) e_j e_j^\top + \frac{1}{\beta_i} \widehat{u}\widehat{u}^\top \right),$$

which has $|\{j : \beta_j < \beta_i\}| + 1$ positive eigenvalues, $|\{j : \beta_j > \beta_i\}|$ negative eigenvalues, and $|\{j : \beta_j = \beta_i\}| - 1$ zero eigenvalues.

The set of all stationary points with only non-negative eigenvalues is

$$M = \left\{ u \in \mathbb{R}^d : \|u\| = \frac{\beta_1}{\lambda_1},\, u \in \mathrm{span}\{e_j : \beta_j = \beta_1\} \right\},$$

and this is the set of global minima. There are no other local minima.

A non-convex gradient descent

Lemma

For $\epsilon > 0$, and $\|v_{T_0}\| \leq b$,

$$\left|\left\{t \geq T_0 : \|v_t\| \geq (1+\epsilon)\beta_1\right\}\right| \leq \frac{2}{\eta\epsilon^2\lambda_1\beta_1}\left(\max_{\|\Lambda w\| \leq b} J(w) - \min_w J(w)\right)$$
$$\leq \frac{3\beta_1}{\eta\epsilon^2\lambda_1\beta_d}.$$

Recall:

- $\beta_d \leq \cdots \leq \beta_1 < \alpha_d \leq \cdots \alpha_2 \leq \alpha_1 = \gamma_1$,
- Norm of v versus β_i determines how components grow, and
- Norm of v versus α_i determines relative growth compared to the leading component.

Bouncing across ravines

Theorem (B., Long, Bousquet, 2022)

There is an absolute constant c such that for any eigenvalues $\lambda_1 > \lambda_2 \geq ... \geq \lambda_d > 0$, any neighborhood size $\rho > 0$, and any step size $0 < \eta < \frac{1}{2\lambda_1}$, for all small enough $\epsilon, \delta > 0$, if w_0 is sampled from a continuous probability distribution over $\mathbb{R}^d$ (density bounded above by A; $\|w_0\|$ not too big; $|w_{0,1}|$ not too small), then with probability $1 - \delta$, for all t sufficiently large (polynomial in d, $1/(\eta\lambda_d)$, λ_1/λ_d and $1/(\lambda_1^2/\lambda_2^2 - 1)$, polylogarithmic in other parameters), for some

$$w^* \in \left\{ \pm \frac{\eta\rho\lambda_1}{2 - \eta\lambda_1} e_1 \right\}$$

and for all $s \geq t$, $\|w_{2s} - w^*\| \leq \epsilon$ and $\|w_{2s+1} + w^*\| \leq \epsilon$.

Bouncing across ravines

SAM's asymptotic behavior

For some

$$w^* \in \left\{ \pm \frac{\eta\rho\lambda_1}{2 - \eta\lambda_1} e_1 \right\},$$

and for all $s \geq t$, $w_{2s} \approx w^*$ and $w_{2s+1} \approx -w^*$.

Bouncing across ravines

SAM's asymptotic behavior

For some
$$w^* \in \left\{ \pm \frac{\eta\rho\lambda_1}{2-\eta\lambda_1} e_1 \right\},$$
and for all $s \geq t$, $w_{2s} \approx w^*$ and $w_{2s+1} \approx -w^*$.

- This is not the solution to the motivating minimax optimization problem: for $\ell(w) = w^\top \Lambda w/2$,

$$\arg\min_{w} \max_{\|\epsilon\| \leq \rho} \ell(w+\epsilon) = 0.$$

Bouncing across ravines

SAM's asymptotic behavior

For some

$$w^* \in \left\{ \pm \frac{\eta\rho\lambda_1}{2 - \eta\lambda_1} e_1 \right\},$$

and for all $s \geq t$, $w_{2s} \approx w^*$ and $w_{2s+1} \approx -w^*$.

- This is not the solution to the motivating minimax optimization problem: for $\ell(w) = w^\top \Lambda w/2$,

$$\arg\min_{w} \max_{\|\epsilon\| \leq \rho} \ell(w + \epsilon) = 0.$$

- SAM's gradient-based approach leads to oscillations around the minimum.
  These oscillations have an impact for a non-quadratic loss.

Outline

- SAM with a quadratic criterion: Bouncing across ravines
  - Stationary points
  - A non-convex gradient descent
  - SAM oscillates around minimum
- Beyond quadratic: Drifting towards wide minima
  - SAM near a smooth minimum
  - Descending the gradient of the spectral norm of the Hessian
- Open questions

SAM: Beyond Quadratic

Locally quadratic objective function

Consider a smooth objective ℓ with a slowly varying (B-Lipschitz) third derivative:

$$\left\| D^3\ell(w) - D^3\ell(w') \right\| \leq B\|w - w'\|.$$

SAM: Beyond Quadratic

Locally quadratic objective function

Consider a smooth objective ℓ with a slowly varying (B-Lipschitz) third derivative:

$$\left\|D^3\ell(w) - D^3\ell(w')\right\| \leq B\|w - w'\|.$$

Consider a local minimum $w_z \in \mathbb{R}^d$:

$$\nabla\ell(w_z) = 0, \qquad H := \nabla^2\ell(w_z) = \mathrm{diag}(\lambda_1, \ldots, \lambda_d),$$

with $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$.

SAM: Beyond Quadratic

Locally quadratic objective function

Consider a smooth objective ℓ with a slowly varying (B-Lipschitz) third derivative:

$$\left\| D^3\ell(w) - D^3\ell(w') \right\| \leq B\|w - w'\|.$$

Consider a local minimum $w_z \in \mathbb{R}^d$:

$$\nabla\ell(w_z) = 0, \qquad H := \nabla^2\ell(w_z) = \mathrm{diag}(\lambda_1, \ldots, \lambda_d),$$

with $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$.
Near w_z, ℓ is close to

$$\ell_q(w) = \ell(w_z) + \frac{1}{2}(w - w_z)^\top H(w - w_z).$$

SAM: Beyond Quadratic

Locally quadratic objective function

Consider an overparameterized setting, with
$\lambda_1 > \lambda_2 \geq \cdots \geq \lambda_k > \lambda_{k+1} = \cdots = \lambda_d = 0$ for $k > 1$.
Suppose

- w_0 satisfies $e_i^\top (w_0 - w_z) = 0$ for $i = k+1, \ldots, d$,
- SAM is initialized at w_0 and applied to the quadratic objective ℓ_q.

Then for all t, the condition $e_i^\top (w_t - w_z) = 0$ for $i > k$ continues to hold, and SAM converges to the set

$$\left\{ w_z \pm \frac{\beta_1}{\lambda_1} e_1 \right\}.$$

SAM: Beyond Quadratic

Locally quadratic objective function

Consider an overparameterized setting, with
$\lambda_1 > \lambda_2 \geq \cdots \geq \lambda_k > \lambda_{k+1} = \cdots = \lambda_d = 0$ for $k > 1$.
Suppose

- w_0 satisfies $e_i^\top (w_0 - w_z) = 0$ for $i = k+1, \ldots, d$,
- SAM is initialized at w_0 and applied to the quadratic objective ℓ_q.

Then for all t, the condition $e_i^\top (w_t - w_z) = 0$ for $i > k$ continues to hold, and SAM converges to the set

$$\left\{ w_z \pm \frac{\beta_1}{\lambda_1} e_1 \right\}.$$

- What is the impact of bouncing over the ravine?

SAM: Drifting Towards Wide Minima

Theorem (B., Long, Bousquet, 2022)

For $s_t \in \{-1, 1\}$, consider the point $w_t = w_z + \frac{s_t \beta_1}{\lambda_1} e_1$

Then, if $B\eta\rho \leq 1$, SAM's update on ℓ gives (for some $\|\zeta\| \leq 1$)

$$w_{t+1} - w_t = -2\frac{\eta\rho\lambda_1 s_t}{2-\eta\lambda_1} e_1 - \frac{\eta\rho^2}{2}\left(1 + \frac{\eta\lambda_1}{2-\eta\lambda_1}\right)^2 \nabla\lambda_{\max}(\nabla^2\ell(w_z)) + \eta\rho^2\left(\frac{(1+\eta\lambda_1)^3\rho}{6} + 2(2\lambda_1 + B\rho)\eta\right) B\zeta.$$

SAM: Drifting Towards Wide Minima

Theorem (B., Long, Bousquet, 2022)

For $s_t \in \{-1, 1\}$, consider the point $w_t = w_z + \frac{s_t \beta_1}{\lambda_1} e_1 = w_z + \frac{\eta\rho\lambda_1 s_t}{2 - \eta\lambda_1} e_1$.

Then, if $B\eta\rho \leq 1$, SAM's update on ℓ gives (for some $\|\zeta\| \leq 1$)

$$w_{t+1} - w_t = -2\frac{\eta\rho\lambda_1 s_t}{2 - \eta\lambda_1} e_1 - \frac{\eta\rho^2}{2}\left(1 + \frac{\eta\lambda_1}{2 - \eta\lambda_1}\right)^2 \nabla \lambda_{\max}(\nabla^2 \ell(w_z)) + \eta\rho^2 \left(\frac{(1+\eta\lambda_1)^3 \rho}{6} + 2(2\lambda_1 + B\rho)\eta\right) B\zeta.$$

The gradient steps have:

- A component that maintains the oscillation in the e_1 direction,

SAM: Drifting Towards Wide Minima

Theorem (B., Long, Bousquet, 2022)

For $s_t \in \{-1, 1\}$, consider the point $w_t = w_z + \frac{s_t \beta_1}{\lambda_1} e_1 = w_z + \frac{\eta\rho\lambda_1 s_t}{2 - \eta\lambda_1} e_1$.

Then, if $B\eta\rho \leq 1$, SAM's update on ℓ gives (for some $\|\zeta\| \leq 1$)

$$w_{t+1} - w_t = -2\frac{\eta\rho\lambda_1 s_t}{2 - \eta\lambda_1} e_1 - \frac{\eta\rho^2}{2}\left(1 + \frac{\eta\lambda_1}{2 - \eta\lambda_1}\right)^2 \nabla \lambda_{\max}(\nabla^2 \ell(w_z)) + \eta\rho^2\left(\frac{(1+\eta\lambda_1)^3\rho}{6} + 2(2\lambda_1 + B\rho)\eta\right) B\zeta.$$

The gradient steps have:

- A component that maintains the oscillation in the e_1 direction,
- A component pointing downhill in the spectral norm of the Hessian,

SAM: Drifting Towards Wide Minima

Theorem (B., Long, Bousquet, 2022)

For $s_t \in \{-1, 1\}$, consider the point $w_t = w_z + \frac{s_t \beta_1}{\lambda_1} e_1 = w_z + \frac{\eta\rho\lambda_1 s_t}{2 - \eta\lambda_1} e_1$.

Then, if $B\eta\rho \leq 1$, SAM's update on ℓ gives (for some $\|\zeta\| \leq 1$)

$$w_{t+1} - w_t = -2\frac{\eta\rho\lambda_1 s_t}{2 - \eta\lambda_1} e_1 - \frac{\eta\rho^2}{2}\left(1 + \frac{\eta\lambda_1}{2 - \eta\lambda_1}\right)^2 \nabla \lambda_{\max}(\nabla^2 \ell(w_z)) + \eta\rho^2\left(\frac{(1 + \eta\lambda_1)^3 \rho}{6} + 2(2\lambda_1 + B\rho)\eta\right) B\zeta.$$

The gradient steps have:

- A component that maintains the oscillation in the e_1 direction,
- A component pointing downhill in the spectral norm of the Hessian,
- For small stepsize parameters $\eta, \rho > 0$, a smaller component reflecting the change of third derivative.

Convergence of Sharpness-Aware Minimization

SAM versus gradient descent

Convergence of Sharpness-Aware Minimization

SAM versus gradient descent

- Far from a minimum, GD and SAM descend the gradient of the objective

Convergence of Sharpness-Aware Minimization

SAM versus gradient descent

- Far from a minimum, GD and SAM descend the gradient of the objective
- Near a minimum, SAM descends the gradient of the spectral norm of the Hessian.

Convergence of Sharpness-Aware Minimization

SAM versus gradient descent

- Far from a minimum, GD and SAM descend the gradient of the objective
- Near a minimum, SAM descends the gradient of the spectral norm of the Hessian.
- SAM uses *one additional gradient measurement per iteration* to compute a specific third derivative: the gradient of the second derivative in the leading eigenvector direction.

Convergence of Sharpness-Aware Minimization

SAM versus gradient descent

- Far from a minimum, GD and SAM descend the gradient of the objective
- Near a minimum, SAM descends the gradient of the spectral norm of the Hessian.
- SAM uses *one additional gradient measurement per iteration* to compute a specific third derivative: the gradient of the second derivative in the leading eigenvector direction.

- Statistical benefits of wide global minima of empirical risk?

Wide ~~global~~ minima of empirical risk?

Coding/information theory:

- Hinton and van Camp. Keeping the neural networks simple by minimizing the description length of the weights. COLT93.
- Hochreiter and Schmidhuber. Flat minima. Neural Comput. 1997.
- Negrea, Haghifam, Dziugaite, Khisti, Roy. Information-theoretic generalization bounds for SGLD via data-dependent estimates. NeurIPS 2019.
- Neu, Dziugaite, Haghifam, Roy. Information-theoretic generalization bounds for stochastic gradient descent. COLT 2021.

Wide ~~global~~ minima of empirical risk?

PAC-Bayes:

- Langford and Caruana. (Not) bounding the true error. NIPS 2002.
- Dziugaite, Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. UAI 2017.

Nonequilibrium statistical physics:

- Baldassi, Borgs, Chayes, Ingrosso, Lucibello, Saglietti, and Zecchina. Unreasonable effectiveness of learning neural networks: From accessible states and robust ensembles to basic algorithmic schemes. PNAS 2016.
- Chaudhari, Choromanska, Soatto, LeCun, Baldassi, Borgs, Chayes, Sagun, and Zecchina. Entropy-SGD: Biasing gradient descent into wide valleys. ICLR 2017.

Outline

- SAM with a quadratic criterion: Bouncing across ravines
  - Stationary points
  - A non-convex gradient descent
  - SAM oscillates around minimum
- Beyond quadratic: Drifting towards wide minima
  - SAM near a smooth minimum
  - Descending the gradient of the spectral norm of the Hessian
- Open questions

Optimization in High-Dimensional Prediction

Olivier Bousquet

Niladri Chatterji

Spencer Frei

Phil Long

- Benign overfitting without linearity: neural network classifiers trained by gradient descent for noisy linear data. Frei, Chatterji, B. COLT 2022 arXiv:2202.05928
- The dynamics of sharpness-aware minimization: bouncing across ravines and drifting towards wide minima. B., Long, Bousquet. arXiv:2210.xxxxx