Energy Landscapes of Deep Networks

Pratik Chaudhari

February 29, 2016

References

Today...

- Introduction to spin glasses
- Write a deep network as a spin glass
- Analyze local minima of such networks
- Modify the energy landscape of deep networks
An introduction to spin glasses

**Ising model**

- Model from statistical mechanics to explain magnetism

\[-H(\sigma) = \sum_{(i,j)} J \sigma_i \sigma_j\]

- Hamiltonian gives the “energy” of a configuration
- Thermodynamically, the model likes to find the “ground state”

\[P(\text{configuration } \sigma) = \frac{1}{Z} e^{-\beta H(\sigma)}\]

- “inverse temperature”
- normalizing constant, also called “partition function”
- very hard to compute

\[Z = \sum_{\sigma \in \{-1, 1\}^n} e^{-\beta H(\sigma)}\]

- Boltzmann’s constant
- “temperature”
Ising model

\[-H(\sigma) = \sum_{(i,j)} J \sigma_i \sigma_j\]

Why interesting?

- Behaves like a magnet!
- Beyond a temperature threshold, spins no longer aligned together
- Great model for analysis of graphical models

Would also be a good model for deep nets!

But very hard to solve for...

\[Z = \sum_{\sigma \in \{-1,1\}^n} e^{-\beta H(\sigma)}\]

large correlations at low temperature

complete disorder at high temperature

exponentially many terms
Enter spin glasses…

2-spin glass

$$-H(\sigma) = \sum_{(i_1,i_2 \leq n)} J_{i_1,i_2} \sigma_{i_1} \sigma_{i_2}$$

Hamiltonian is also a random variable

- is now a random variable with exponential tails
- usually Gaussian
- couples all possible pairs of spins

We will need a p-spin glass

$$-H_p(\sigma) = \sum_{1 \leq i_1,\ldots,i_p \leq n} J_{i_1,\ldots,i_p} \sigma_{i_1} \sigma_{i_2} \ldots \sigma_{i_p}$$

Why call them “glasses”?

- Long range coupling, not just neighbors
- Glass is really just a liquid that flows very slowly

- Crystals: cooling a liquid very quickly, short range coupling
- Glasses have long range coupling
A model for deep networks

Loss function (zero-one loss) is a spin glass Hamiltonian:

\[ H(\sigma) \overset{\text{law}}{=} \frac{J}{n(p-1)/2} \sum_{i_1,\ldots,i_p=1}^n J_{i_1,\ldots,i_p} \sigma_{i_1} \cdots \sigma_{i_p} \]

- Active paths in the network
- Unique weights in the network
- zero-mean standard Gaussian

sparse, random connections, threshold nonlinearities

n neurons on each layer

p layers

X: input

h: hidden units

scalar output
Random sparse deep network

\[ Y = W^\top_p \sigma(W_{p-1} \sigma(W_{p-2} \ldots \sigma(W_1 X)) \ldots) \]

- \( W_p \in \mathbb{R}^n \)
- \( \sigma \) is a decision mechanism
- \( W_i \in [-1,1]^{n \times n} \)
- \( Y \in \mathbb{R} \)
- \( X \in \mathbb{R}^n \)

Some assumptions on weights

- on an average, \( "d" \) non-zeros
  \[ d = n^\alpha; \quad \alpha < 1/5 \]
- weights are not all close to zero
  \[ \mathbb{E}(W_{k,ij}) = 0 \]
  \[ \text{var}(W_{k,ij}) \geq n^{-2/5} \]

Choromanska et. al. use a non-sparse model with ReLU nonlinearities

\[ p = \log_d n = \frac{1}{\alpha} \]
Random sparse deep network

Can write it as …

\[
Y = \sum_{i=1}^{n} \left( \sum_{\gamma \in \Gamma_i} X_i \left( \prod_{k=1}^{p} w^k_{\gamma} \right) \right)
\]

- Class-label
- Sum over the output neurons
  - Dimensions of \( X \)
- Sum over all “active” paths in the network
  - \( i^{th} \) element dimension of \( X \)
  - \( i^{th} \) set of active paths from \( i^{th} \) output neuron to \( Y \)

*but introduces weird couplings / disorder*

*these are our spins!*

*product of all the weights along path \( \gamma \)*
Two ways to remove correlations —

- Assume independence of paths

\[ X_{i_1} \rightarrow X_{i_1, \ldots, i_p} \]

- We shall explicitly account for them

\[ Y = \sum_{i, i_1, \ldots, i_p=1}^{n} \gamma_{i, i_1} X_i W_{i, i_1} \gamma_{i_1, \ldots, i_p} W_{i_1, \ldots, i_p} := Z_{i, i_1} \]

\[ | \sum_{i} Z_i | \leq n^{-1/5} \]

first edge, i.e., between layer 1 and layer 2
net “coupling” of a path due to the sum over all outputs is small
path from layer 2 to \( Y \)

Choromanska et. al.
Model for deep networks

For a 0-1 / logistic loss function, can show:

\[
H(\sigma) \overset{\text{law}}{\approx} \frac{J}{n(p-1)/2} \sum_{i_1,\ldots,i_p=1}^{n} J_{i_1,\ldots,i_p} \sigma_{i_1} \ldots \sigma_{i_p}
\]

- estimate the number of active paths in the sparse network
- zero-mean standard Gaussian

Because of sparsity, can estimate the probability that a neuron of any layer is active or not

Key Idea —- Can now analyze a “generic” neural network

(for analysis) assume spherical constraint

\[
\sigma \in S^{n-1}(\sqrt{n})
\]
Number of critical points

All critical points:
\[ \text{crt}(B) = |\{ \sigma : \nabla H(\sigma) = 0, \ H(\sigma) \in nB \}| \]

Critical points of index \( k \):
\[ \text{crt}_k(B) = \left| \left\{ \sigma : \nabla H(\sigma) = 0, \ H(\sigma) \in nB, \ i \left( \nabla^2 H(\sigma) \right) = k \right\} \right| \]

- any interval of real line
- gradient is zero
- energy lies in interval \( B \)
- counts saddle points in interval \( B \)
- \( k \) eigenvalues of Hessian have negative real part
Number of critical points

crt(B) = all critical points
crt_k(B) = saddle points of index k

\[
\lim_{n \to \infty} \frac{1}{n} \log \mathbb{E} \text{crt}_k(u) = \Theta_k(u), \quad \text{for all } u \in \mathbb{R}, k \in \mathbb{N}
\]

\[
\lim_{n \to \infty} \frac{1}{n} \log \mathbb{E} \text{crt}(u) = \Theta(u), \quad \text{for all } u \in \mathbb{R}, k \in \mathbb{N}
\]

exponentially many local minima and saddle points!

Energy barriers for \( p = 3 \)

\( \Theta(u) \)

- \( E_0 = -1.657 \)
- \( -E_{\text{inf}} = -1.633 \)

\( H/n \)

- \( k = 0 \)
- \( k = 1 \)
- \( k = 4 \)
- \( k = 10 \)
Number of critical points

crt(B) = all critical points
crt_k(B) = saddle points of index k

Corollary

\[
\lim_{n \to \infty} \frac{1}{n} \log \mathbb{E} \text{crt}_k(\mathbb{R}) = \frac{1}{2} \log(p - 2) - \frac{p - 3}{p - 1}
\]

\[
\lim_{n \to \infty} \frac{1}{n} \log \mathbb{E} \text{crt}(\mathbb{R}) = \frac{1}{2} \log(p - 2)
\]

exponentially many local minima as well as saddle points in total!
Location of critical points

**Theorem**

\[
\lim_{n \to \infty} \sup \frac{1}{n^2} \log P (\exists \text{ a saddle point of index } k \text{ above energy } - n(E_\infty - \epsilon)) < 0
\]

\[
\lim_{n \to \infty} \sup \frac{1}{n} \log P (\exists \text{ a saddle point of index } \geq k \text{ above energy } - n(E_k + \epsilon)) < 0
\]

- Local minima exist throughout \([-nE_0, -nE_\infty]\)
- SGD might think low-index saddle points are minima
- jumping between local minima is hard…
- “normalized” width of energy bands is small, can’t check numerically

i.e., decreases exponentially

“layered” structure

index \( \leq k \) saddle points between \([-nE_0, -nE_k]\)

SGD finds one of these

energy bands

have to climb the energy barrier to reach a saddle point, then climb down again

fortunately, width of bands decreases with \( k \)
Experimental evaluation: 2-spin glass, simulated annealing

same as number of neurons “n”

Choromanska et. al.
Experimental evaluation: 1-layer MNIST network, SGD

Choromanska et. al.
Experimental evaluation: 1-layer MNIST network, SGD

- SGD finds critical points of low normalized index, i.e., does not get stuck at saddle points
- Of course, there are those exponentially many local minima

Choromanska et. al.

should become worse with more layers due to correlations

variance increases with “n”?
Trivialization of critical points

Simple example

\[-H = \frac{1}{2} \sigma^\top J \sigma + h^\top \sigma; \quad |\sigma| = n\]

- random matrix, each entry is Gaussian
- random magnetic field, Gaussian

\[\text{var}(J_{ij}) = \frac{1 + \delta_{ij}}{n}\]

\[\text{var}(h_i) = \nu^2\]

Three distinct regimes —

- asymptotically \(\mathcal{O}(n)\) critical points for \(\nu = \mathcal{O}(n^{-1/2})\)
- Number of critical points decreases with magnetic field
- Total trivialization, 2 critical points for \(\nu = \mathcal{O}(n^{-1/6})\)
Trivialization of critical points

\[ \nu = O(n^{-1/2}) \]

\[ \nu = O(n^{-1/6}) \]
Trivialization: Changing the number of critical points

\[ -\tilde{H}(\sigma) = -H(\sigma) + \sum_i h_i \sigma_i \]

Variance of magnetic field: \( \nu_c = \sqrt{p(p-2)} \)

Three distinct regimes —

- Exponentially many critical points with small noise \( \nu < \nu_c \)  
- Polynomialsly-many critical points near threshold \( |\nu - \nu_c| \approx \mathcal{O}(n^{-1}) \)  
- Total trivialization, 2 critical points \( \nu > \nu_c + \Omega(n^{-1}) \)

Important regime

Want to avoid this
Monte-Carlo simulation on a spin glass

\[-\tilde{H}(\sigma) = -H(\sigma) + \sum_i h_i \sigma_i\]

$n = 100, p = 3$

- Local-minima separated by high energy barriers
- Saddle points are the “canyons” above

- Almost constant Hamiltonian
- All points land at the same local minimum
Monte-Carlo simulation on a spin glass

\[-\tilde{H}(\sigma) = -H(\sigma) + \sum_i h_i \sigma_i\]

- Very few local minima
- Much larger proportion of saddle points

(for a slightly different model) can prove that in this regime, only low order saddle points and local minima persist.
Distribution of local minima concentrates

\[
\lim_{n \to \infty} P \left( \min_{\sigma} H(\sigma) - m_n \geq t \right) = \exp \left( -\frac{1}{c_p} e^{c_p t} \right)
\]

Subag, Zeitouni (2015)

Gumbel distribution

Probability density of Hamiltonian at local minima

Gumbel density

\[
\mu = -1.58 \\
\beta = 0.03
\]
Distribution of local minima concentrates

\[
\lim_{n \to \infty} P \left( \min_{\sigma} H(\sigma) - m_n \geq t \right) = \exp \left( -\frac{1}{c_p} e^{c_p t} \right)
\]

Subag, Zeitouni (2015)
Perturbations of local minima

**Theorem**

\[ \Delta H_n(0) = \frac{1}{n} \left| H_{\text{quad}}(Y_\sigma) - H_{\text{quad}}(0) \right| \]

Quadratic approximation at a critical point

\[ \lim_{n \to \infty} \Delta H_n(0) = \nu \]

\[ \|Y_\sigma\| \leq n^{-1/4} \]

\( \nu = O(1) \)

Roughly, magnetic field does not perturb the local minima, only changes the Hamiltonian
Annealing the magnetic field

\[-\tilde{H}(\sigma) = -H(\sigma) + \frac{p \alpha}{2\sqrt{n}} \sum_i \sigma_i^2 + \sum_i h_i \sigma_i\]

- Set a parameter: \(\tau \ll n\)
- Magnetic field for the polynomial regime is:
  \[\nu = Jp \left(1 + \tau \frac{\alpha^2}{n} + \frac{2\tau}{n}\right)^{1/2}\]
- Anneal magnetic field to zero as training progresses:
  \[\tau(k) = -\frac{n}{\alpha^2 + 2} \frac{k}{N}\]

\(\tilde{H}(\sigma)\) is L2 regularization, \(H(\sigma)\) is (same as before) random magnetic field.

Helps transition between:
- Polynomial regime: \(\tau \gg 1\)
- Exponential regime: \(\tau \to -\infty\)

But fix its direction, don’t resample!
Fully-connected "deep network": MNIST

\[
\text{mnistfc : input } \rightarrow \text{linear}_{64} \rightarrow \text{linear}_{10} \rightarrow \text{softmax}
\]
\[
\times 20
\]
relu nonlinearities

- Does not exactly match the theoretical model, so set

\[
p = 20, \quad n = \sqrt{\frac{\#\text{weights}}{p}}
\]
Fully-connected “deep network”: CIFAR-10

cifarfc : input $\rightarrow \underbrace{\text{linear}_{128}}_{\times 20} \rightarrow \text{linear}_{10} \rightarrow \text{softmax}$

relu nonlinearities

\[ \text{error} = k^{-c} \]

Trivialization is a great way to avoid bad initialization of weights
Convolution neural network: CIFAR-10

cifarconv : input → \( \text{conv}_{5 \times 5 \times 8} \) → \( \text{linear}_{128} \) → \( \text{linear}_{10} \) → softmax

relu nonlinearities, max-pooling

Exact same training parameters as previous experiment
Convolution neural network: CIFAR-10

cifarconv: input $\rightarrow \text{conv}_{5 \times 5 \times 8} \rightarrow \text{linear}_{128} \rightarrow \text{linear}_{10} \rightarrow \text{softmax}$

relu nonlinearities, max-pooling

Weights keep sloshing with resampling

Much larger minimum gradient

cifarconv: Alignment of weights

$|\Sigma_i h_i^i|$

- $\ell_2$, triv, anneal
- $\ell_2$, anneal, resample

cifarconv: Minimum gradient

$\min |g_k|$

- $\ell_2$
- $\ell_2$, triv, anneal
- $\ell_2$, anneal, resample
"Network-in-network" architecture: CIFAR-10

cifarnin : input $\rightarrow$ conv$_{5 \times 5 \times 192}$ $\rightarrow$ max-pool$_{3 \times 3}$ $\rightarrow$ dropout $\times 3$

$\rightarrow$ conv$_{3 \times 3 \times 192}$ $\rightarrow$ conv$_{1 \times 1 \times 192}$ $\rightarrow$ conv$_{1 \times 1 \times 10}$ $\rightarrow$ mean-pool$_{8 \times 8}$ $\rightarrow$ softmax $\times 2$

relu nonlinearities, batch-normalization, global average pooling

cifarnin: Validation error

Exact same parameters!
Conclusions

Spin glasses

• Good models for deep networks,
• energy landscape has similar properties

Gradient noise of right magnitude

dramatically helps very-deep networks, fully-connected layers

Theoretical goals

• Effect of regularization on the energy landscape
• Devise specialized, non-local optimization algorithms