Over-Parameterized Learning and Stochastic Gradient Descent

Jeffrey Mei, Cody Melcher, Kamaljeet Singh

Bias-Variance Trade-Off

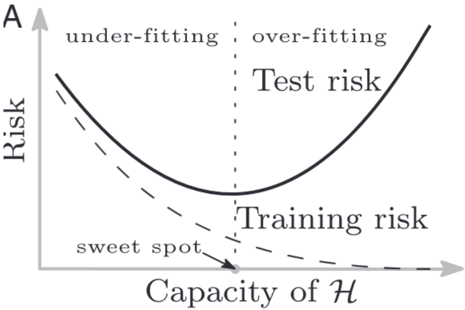


Figure: Bias-Variance Trade-Off [1]

- increasing model complexity can lead to **overfitting**
- basis for many methods: lasso, cross-validation, ensemble methods, AIC, BIC, ...
- fails to explain success of neural networks ...

[1] [Belkin et al., 2018]

Double-Descent

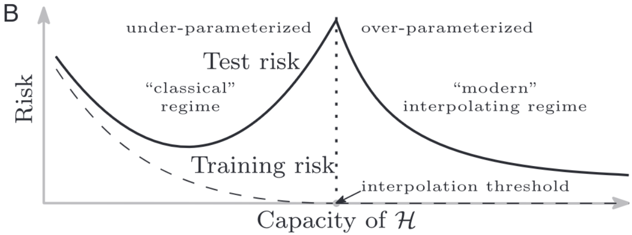
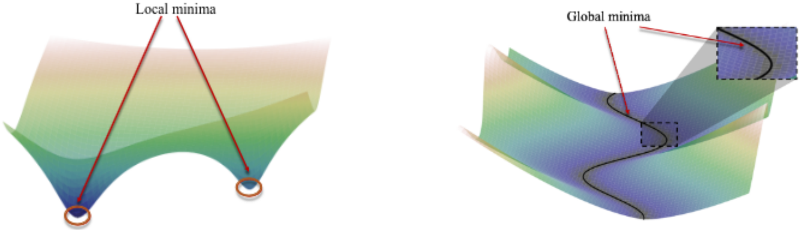


Figure: Double-Descent

- bias-variance trade-off is only *half* the picture!
- monotonic improvement with increasing model complexity
- **interpolation threshold**: model complexity with no training error
- most theory lies on the left of the interpolation threshold
- **over-parameterized**: right of interpolation threshold

Local Minima ≈ Global Minima



(a) Under-parameterized models (b) Over-parameterized models

Under-Parameterized

- SGD often gets stuck in local minima
- motivates momentum

Over-Parameterized

- minima are likely to be global

Exponential Convergence

Under-Parameterized

- non-exponential convergence rate
- variable step size

Over-Parameterized

- exponential convergence rate
- constant step size

Saturation

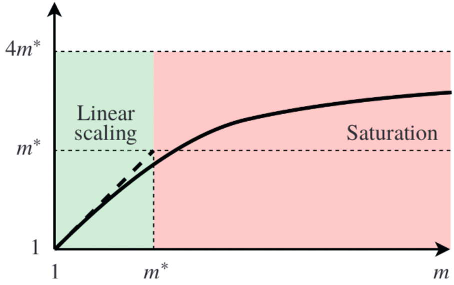


Figure:
x-axis: number of iterations with batch size m
y-axis: number of iterations with batch size 1

Under-Parameterized

- 1 iteration of batch size $m \approx$ m iterations of batch size 1

Over-Parameterized

- moderate mini-batch SGD $\approx$ full gradient descent

SGD Over-Parameterized

Under-Parameterized

- non-exponential convergence
- local minima are not global
- linear batch size

Over-Parameterized

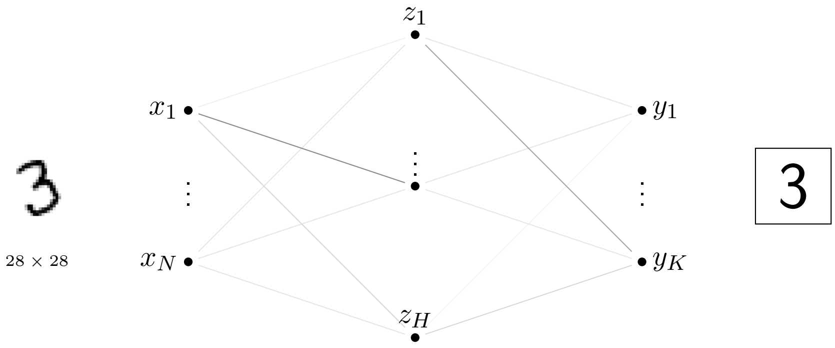
- exponential convergence
- local minima are global
- batch size saturation

Artificial Neural Networks Crash Course

- key technology behind many AI advances
- *enormous fitting capacity*: can memorize noise
- network model, where edges represent parameters
- how does SGD relate to neural networks?

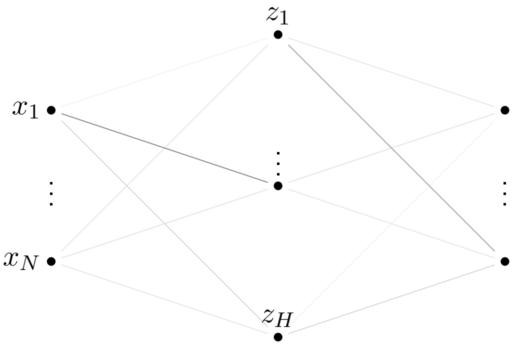
Training Artificial Neural Networks

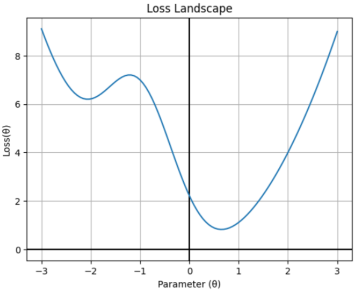
1. forward propagation: calculate error (to adjust weights)
2. back propagation: adjust weights (using SGD)
3. repeat until convergence



Training Artificial Neural Networks

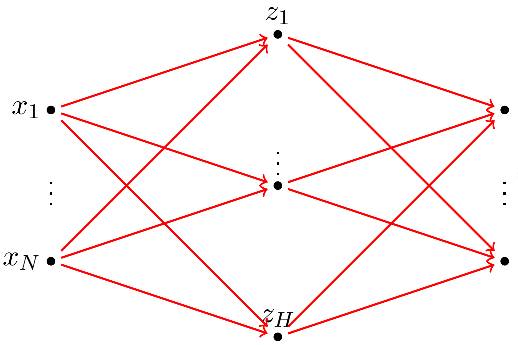
1. forward propagation: calculate error (to adjust weights)
2. back propagation: adjust weights (using SGD)
3. repeat until convergence

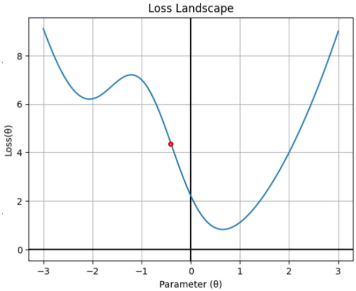




Training Artificial Neural Networks

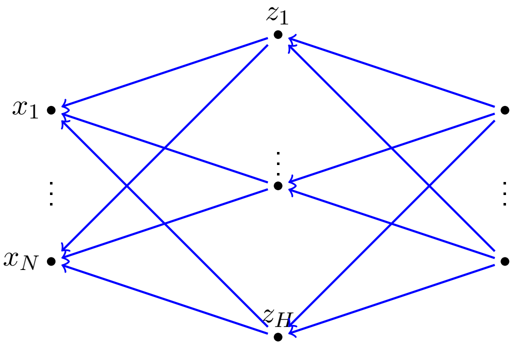
1. forward propagation: calculate error (to adjust weights)
2. back propagation: adjust weights (using SGD)
3. repeat until convergence

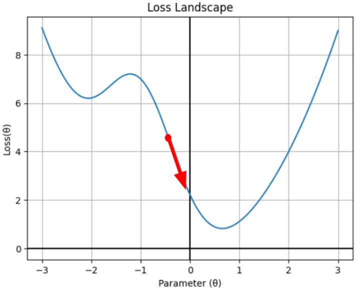




Training Artificial Neural Networks

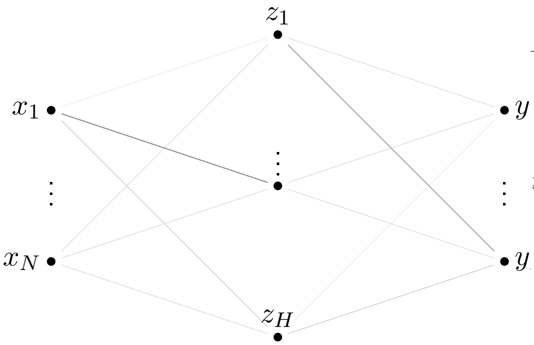
1. forward propagation: calculate error (to adjust weights)
2. back propagation: adjust weights (using SGD)
3. repeat until convergence

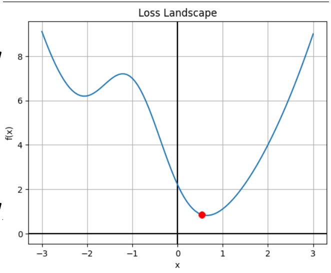




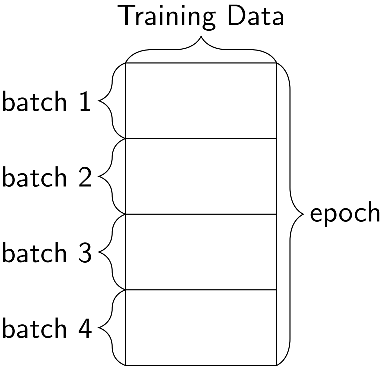
Training Artificial Neural Networks

1. forward propagation: calculate error (to adjust weights)
2. back propagation: adjust weights (using SGD)
3. repeat until convergence





Key Terms



- **batch:** number of training examples in SGD
- **iterations:** number of parameter updates
- **epochs:** number of passes through training data

Numerical Analysis

Numerical Experiment:

1. reproduce double-descent curve
2. compare under/over-parameterized models
   - can we observe batch size saturation?

Double Descent Curve

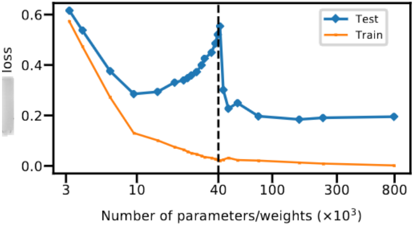


Figure: Expectation

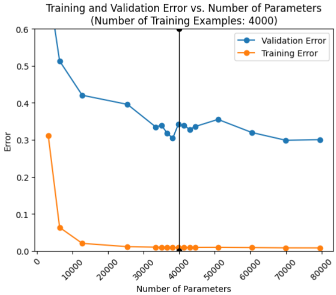


Figure: Reality

- our double descent curve is not as dramatic

Batch Saturation

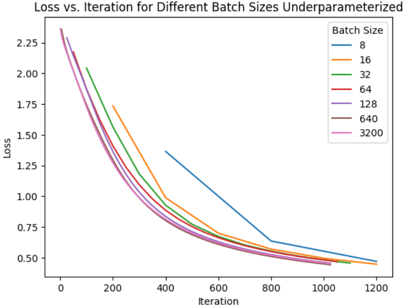


Figure: Underparameterized

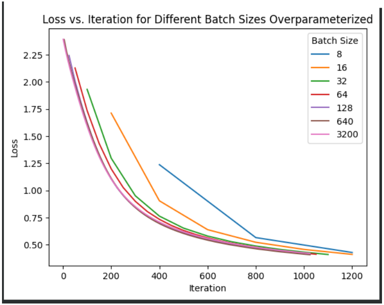


Figure: Overparameterized

- overparameterized is clustered near full batch

Estimating m^*

- Paper gives critical batch size as $m^* = \frac{\beta}{\lambda_1 - \lambda_k} + 1$. Seems nice to know ie can pick it to maximize efficiency.
- β is smoothing parameter, λ_1, λ_k largest and smallest strictly positive eigenvalues.
- Estimate β ala Lipschitz: Product of spectral norms of weight matrices and norms of activation functions.
- Estimate λ_1, λ_k via eigenvalues of final weight matrix.

Parameter	**Underparameterized**	**Overparameterized**
β	8.5223	8.9454
λ_1	1.7334	1.6857
λ_k	0.7199	0.8965
m^*	**9.4094**	**12.3491**

Conclusion

SGD behaves very differently in the over-parameterized regime

- batch size saturation: moderate batch sizes $\approx$ full gradient descent

Future Directions:

- Test convergence rates against methods FISTA etc.
- numerical experiments to show local is global
- compare SGD to SAGA, FISTA, etc. in overparameterized regimes

References

Belkin, M. (2021).
Fit without fear: remarkable mathematical phenomena of deep learning through the prism of interpolation.
Acta Numerica, 30:203–248.

Belkin, M., Ma, S., and Mandal, S. (2018).
To understand deep learning we need to understand kernel learning.
arXiv:1802.01396 [cs, stat].