Regularization

Seyoung Yun

- http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture7.pdf

- N. Srivstava et al, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting” http://jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf

- Sergey Ioffe and Christian Szegedy“Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift” https://arxiv.org/abs/1502.03167

- C. Zhang et al “Understanding deep learning requires rethinking generalization” https://arxiv.org/abs/1611.03530

Regularization

- **“A regularizer is anything that hurts the training process”**
  C. Zhung at ICLR2017 (https://www.youtube.com/watch?v=kCj51pTQPKI)

- data augmentation

- **weight decay - with an additional cost**

- **dropout - by adding random noise**

Linear Regression

- RSS: cost of linear regression

$$\mathcal{L}(w, b) = \sum_{i=1}^{m} (y^{(i)} - w^\top x^{(i)} - b)^2$$

- regularizer

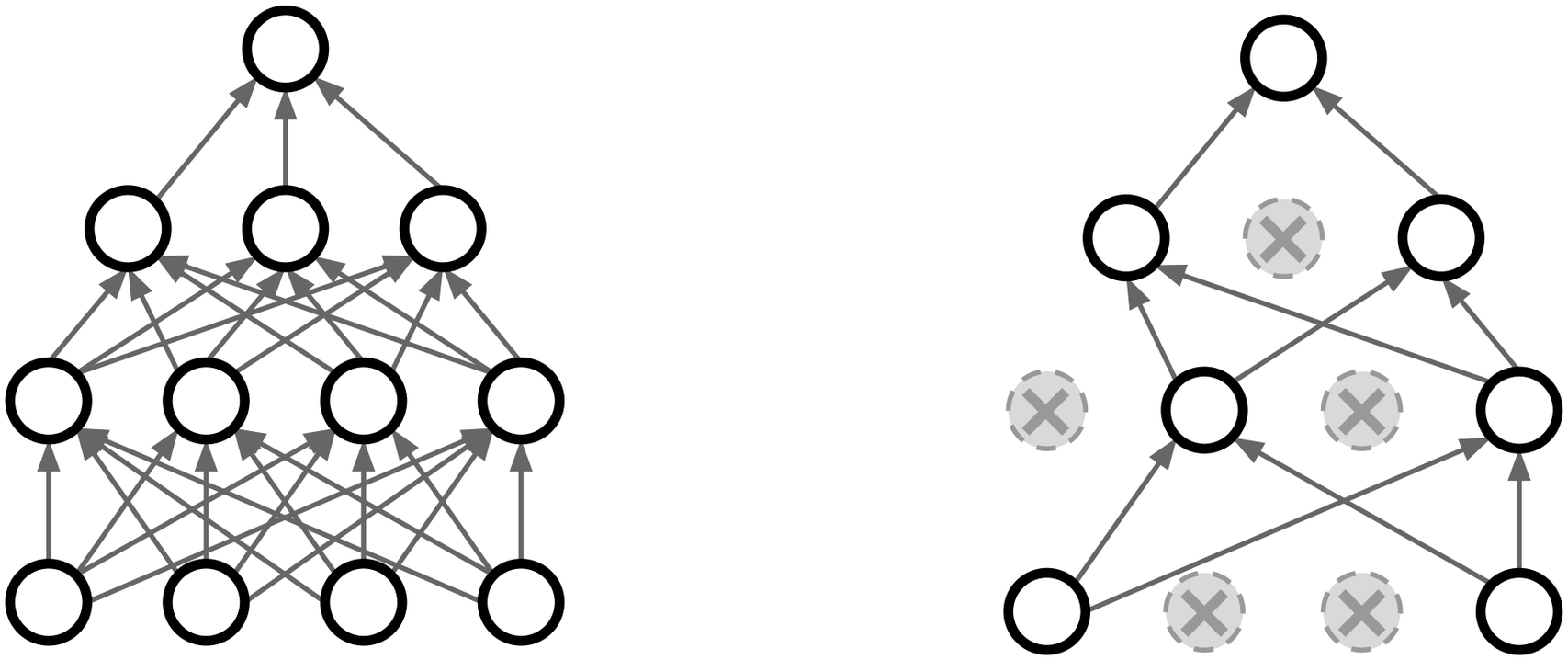
$$\mathcal{L}(w, b) = \frac{1}{m} \sum_{i=1}^{m} (y^{(i)} - w^\top x^{(i)} - b)^2 + \frac{\lambda}{2m} \|w\|_2^2$$

or

$$\mathcal{L}(w, b) = \frac{1}{m} \sum_{i=1}^{m} (y^{(i)} - w^\top x^{(i)} - b)^2 + \frac{\lambda}{2m} \|w\|_1$$

Weight Decay

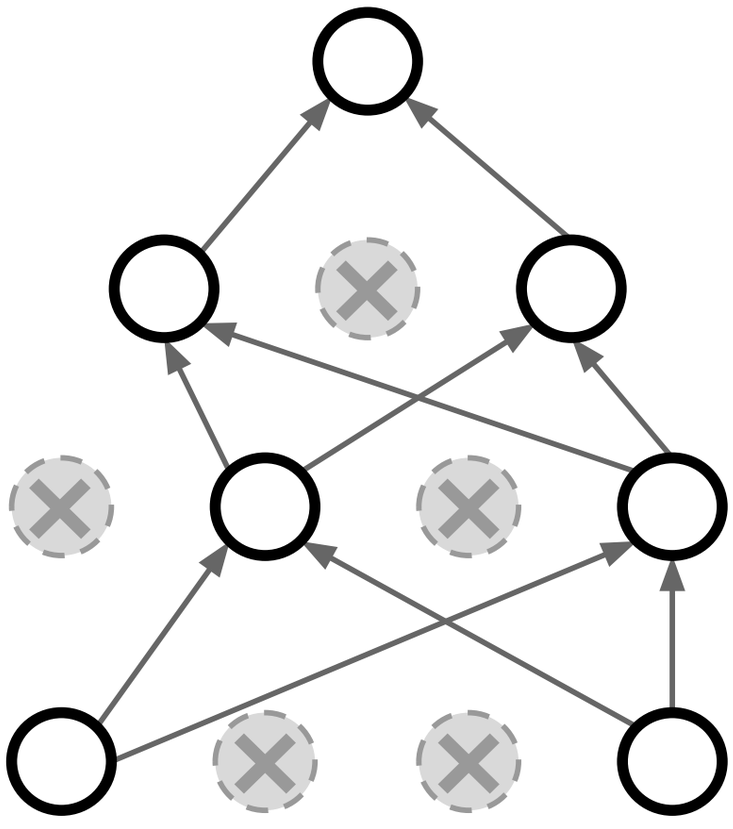
Dropout



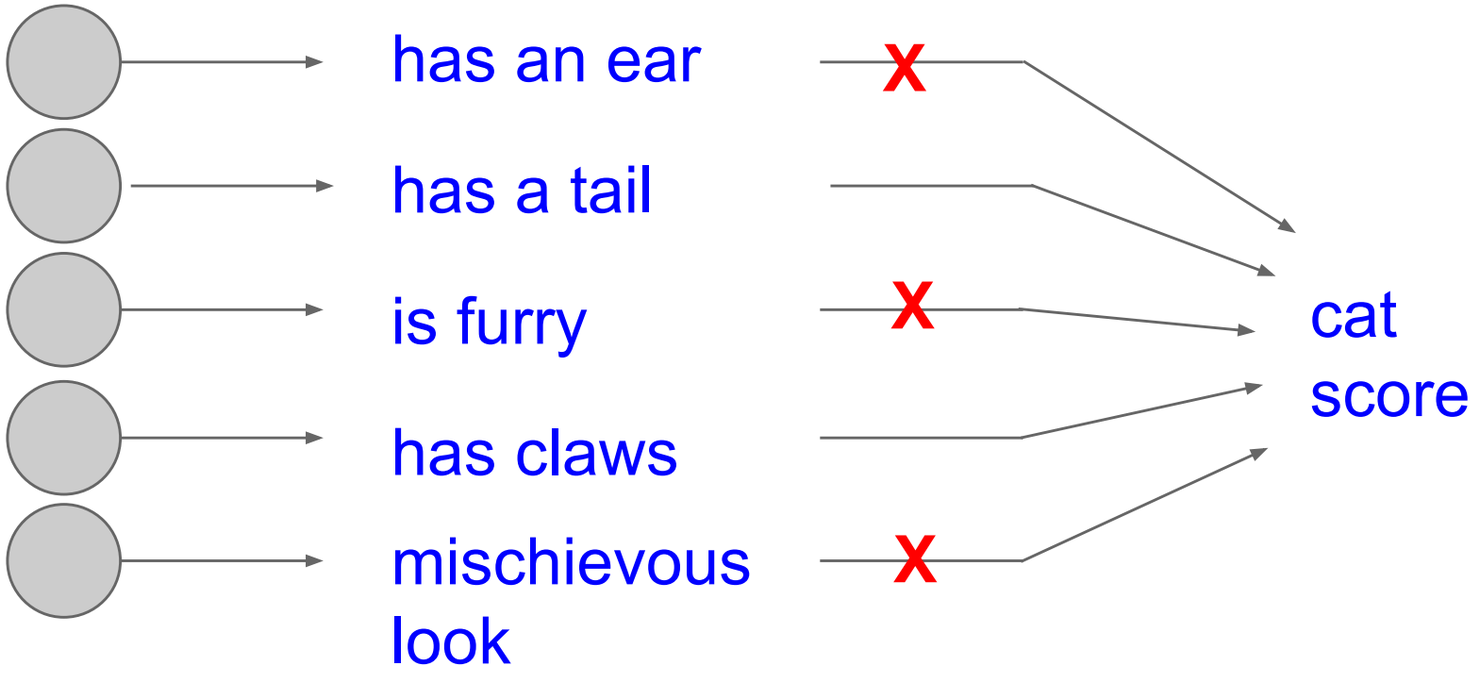
- In each forward pass, randomly erase neurons

Dropout

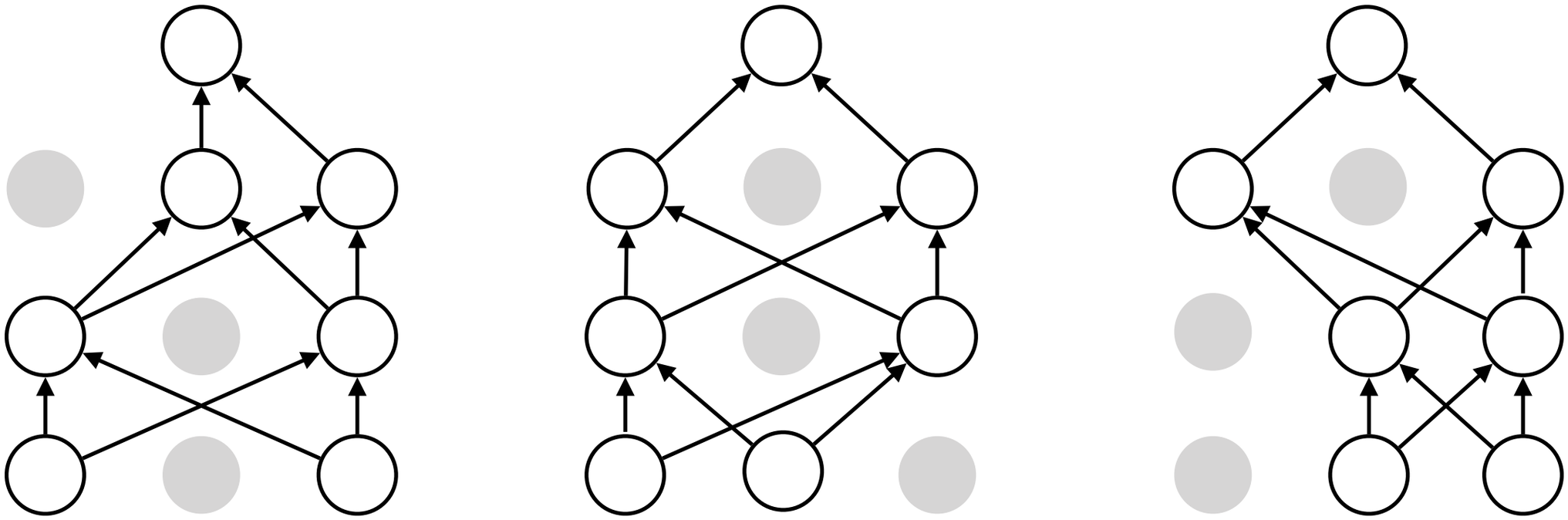
- Why can this be good?



Forces the network to have a redundant representation;
Prevents co-adaptation of features



Ensemble of models



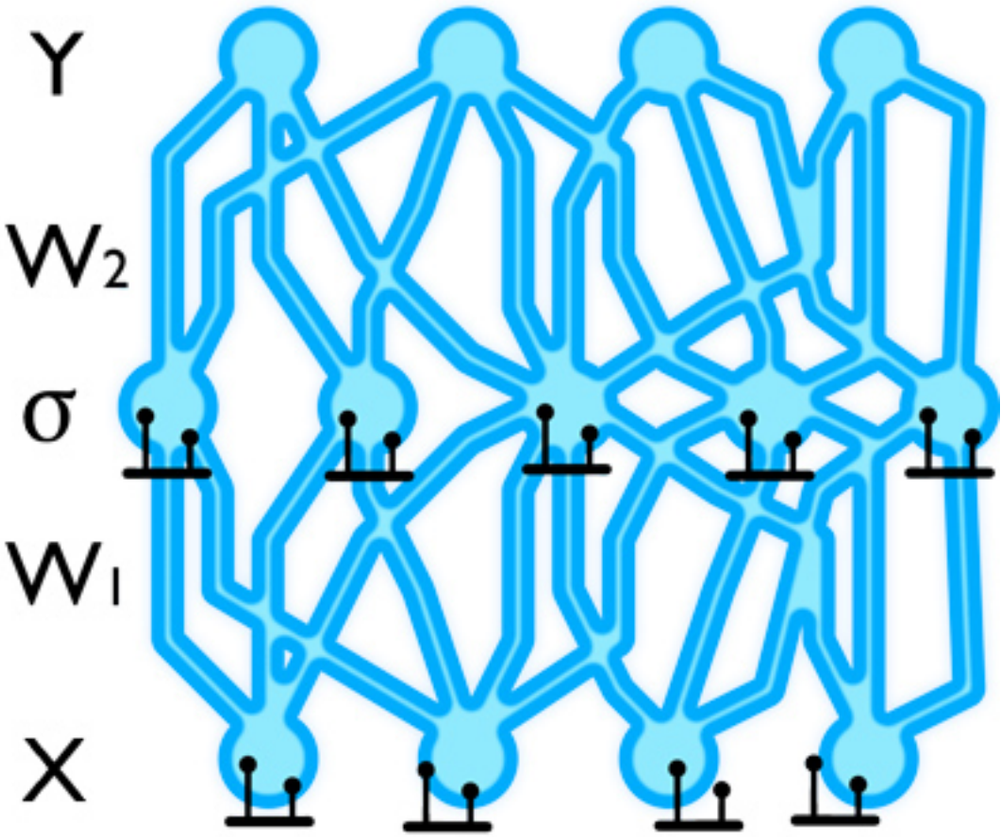
- Dropout is training a large ensemble of models (that share parameters).

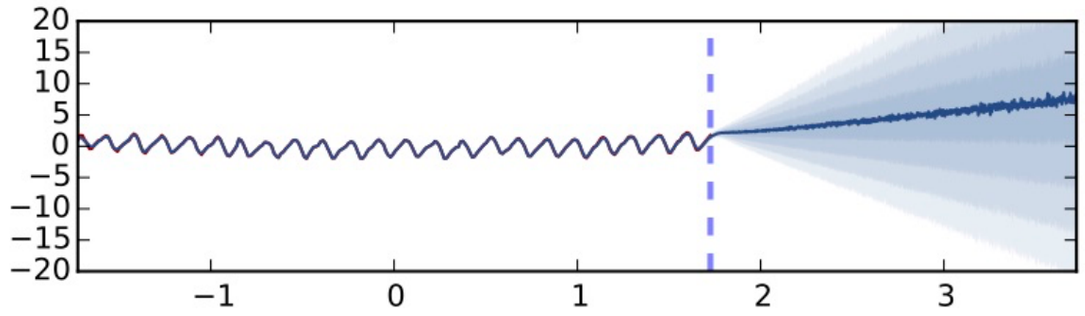
- Each binary mask is one model

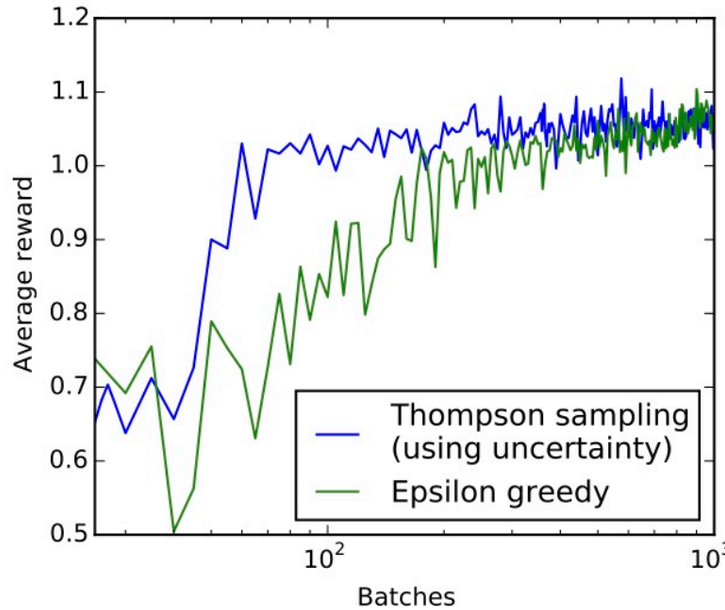
Dropout: Test time

- No dropout at test time
- scaling by dropout probability

Dropout: uncertainty







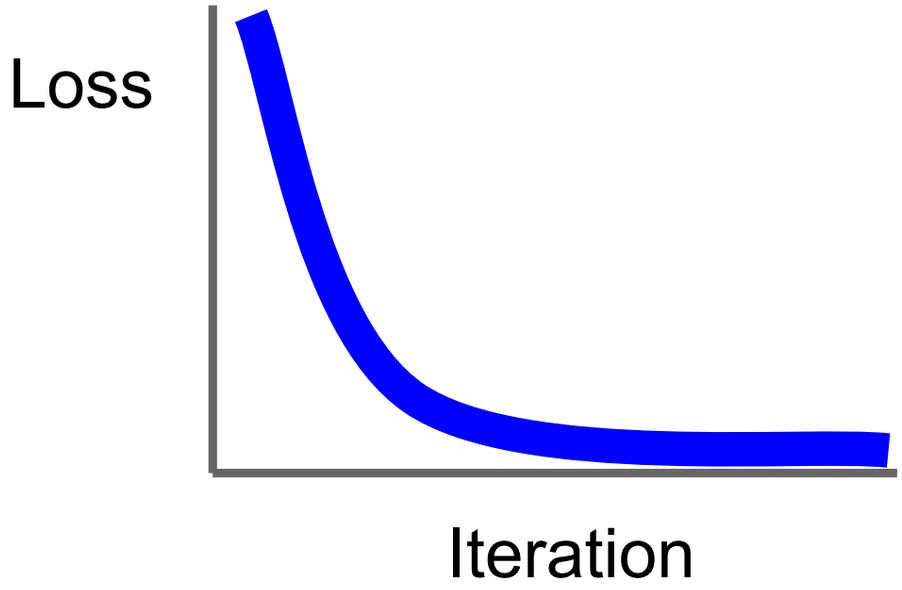
- Dropout generates ensemble of models

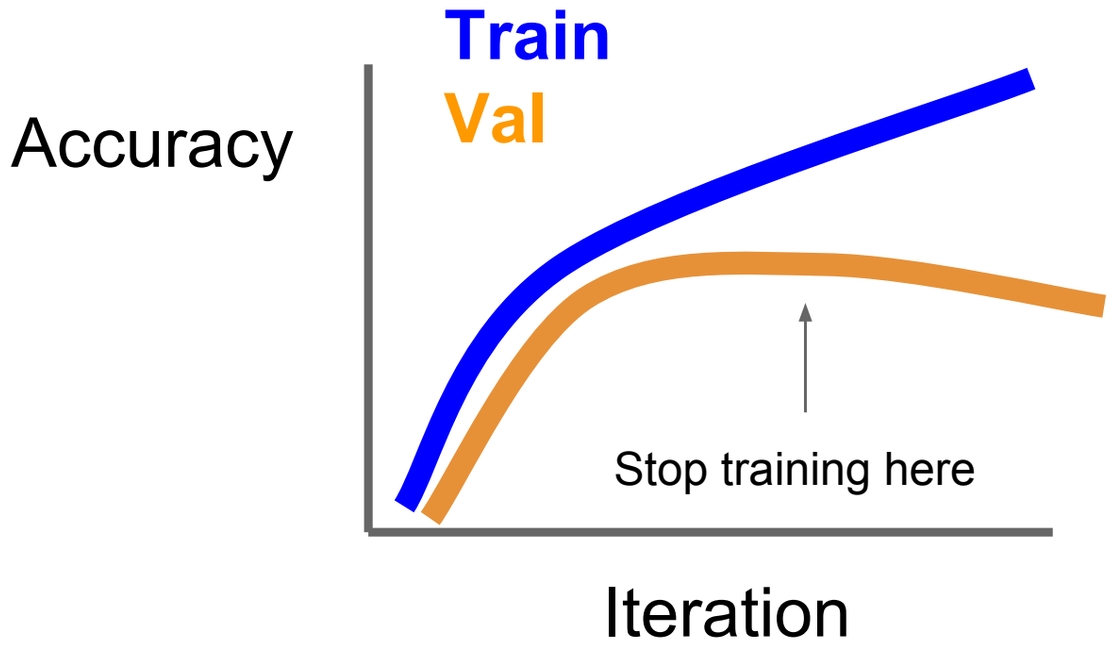
- From the ensemble, estimate mean and variance of the output

Y. Gal and Z. Ghahrami, “Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning,” https://arxiv.org/abs/1506.02142

SGD and Early stopping

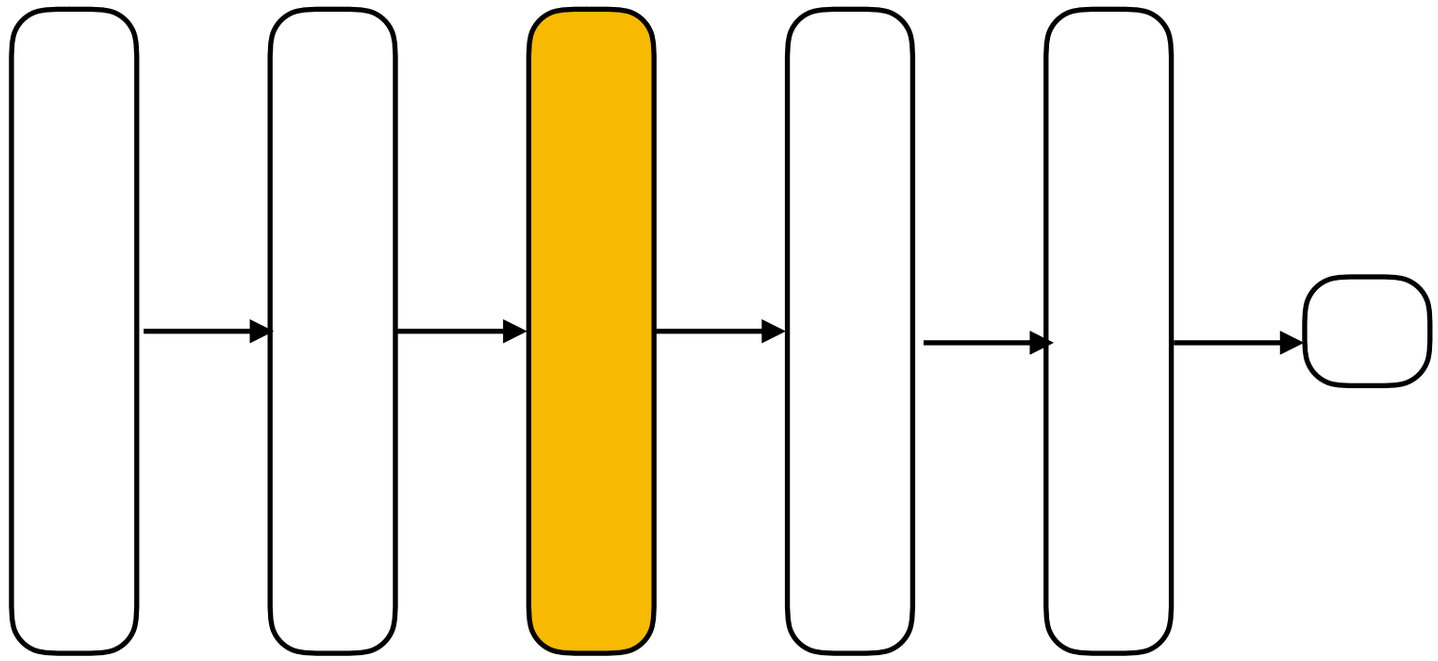
- SGD adds noises to the network -> a regularizer
- Early stopping





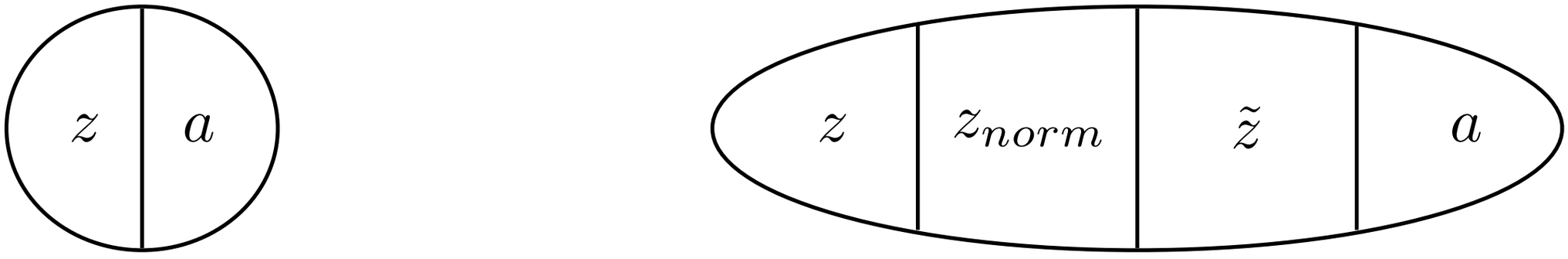
Batch Normalization

- Covariate shift



- “The change in the distributions of layers’ inputs presents a problem because the layers need to continuously adapt to the new distribution. When the input distribution to a learning system changes, it is said to experience covariate shift”

Batch Normalization



Training with Batch Norm.

- Original: $\{w^{[1]}, b^{[1]}, \ldots, w^{[L]}, b^{[L]}\}$
- With Batch Norm.: $\{w^{[1]}, b^{[1]}, \beta^{[1]}, \gamma^{[1]}, \ldots, w^{[L]}, b^{[L]}, \beta^{[L]}, \gamma^{[L]}\}$

Batch Norm at test time

Batch Norm as regularization

- Each mini-batch is scaled by the mean/variance computed on just that mini-batch.

- This adds some noise to the values $z^{[l]}$ within that minibatch. So similar to dropout, it adds some noise to each hidden layer's activations.

- This has a slight regularization effect.