Neural Network Regularization

Part B - Dropout

CS1090B Data Science II – Spring 2025

Pavlos Protopapas, and Chris Gumb

Songhan Hu
Qingdao, China

Outline

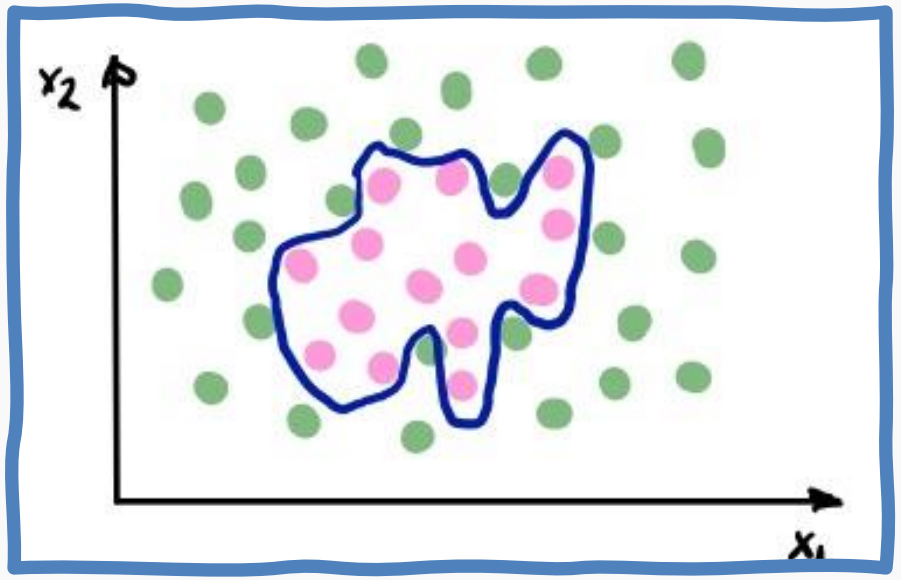
- Regularization of NN
  - Norm Penalties
  - Early Stopping
  - Data Augmentation
  - **Dropout**

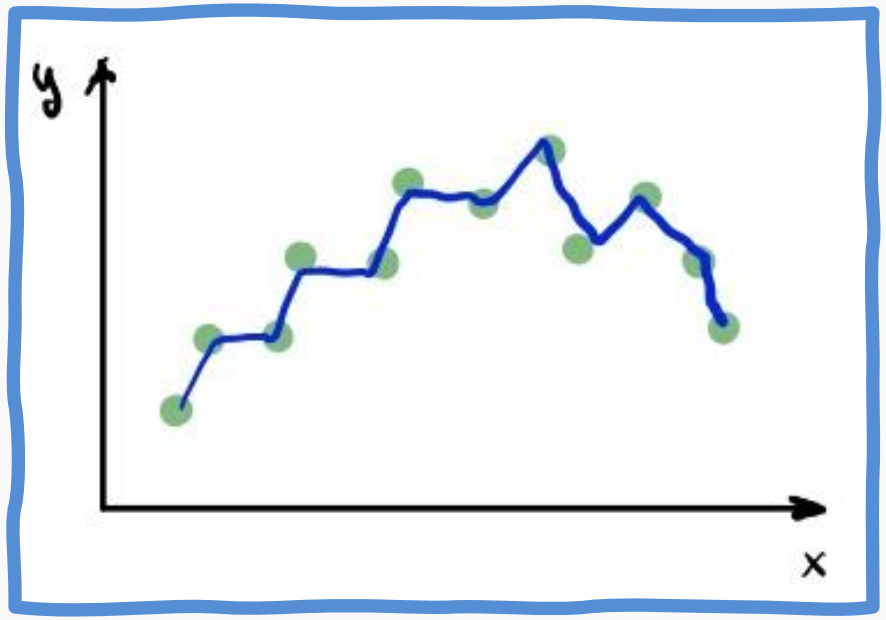
Co-adaptation

Overfitting occurs when the model is sensitive to slight variations on the input and therefore it fits the noise.

L1 and L2 regularizations 'shrink' the weights to avoid this problem.

However, in a large network many units can collaborate to respond to the input while the weights can remain relatively small. This is called co-adaptation.





Game Time

How would you stop neuron co-adaptation?

A. Regularize even more

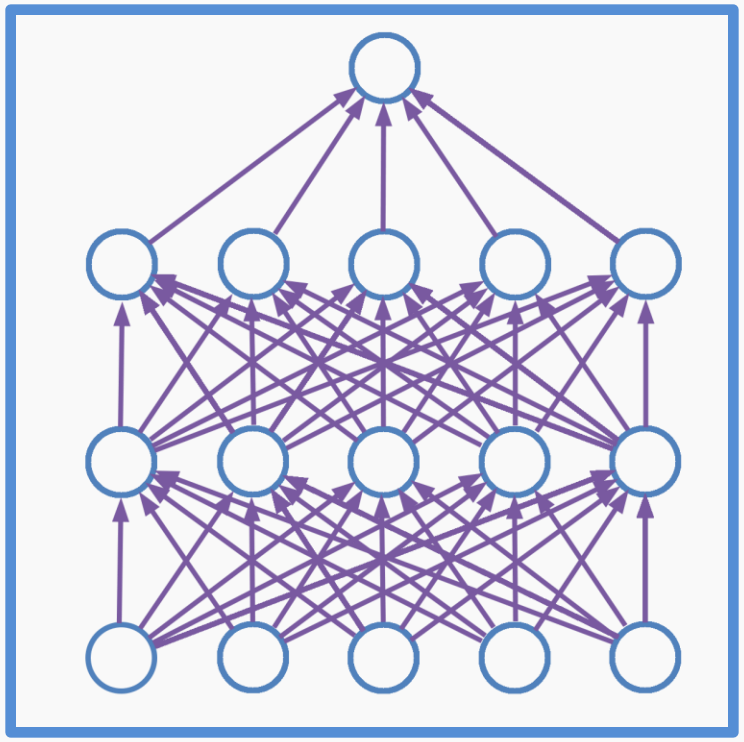
B. Force some of the neurons not to participate

C. Force some of the neurons not to participate for some of the training data

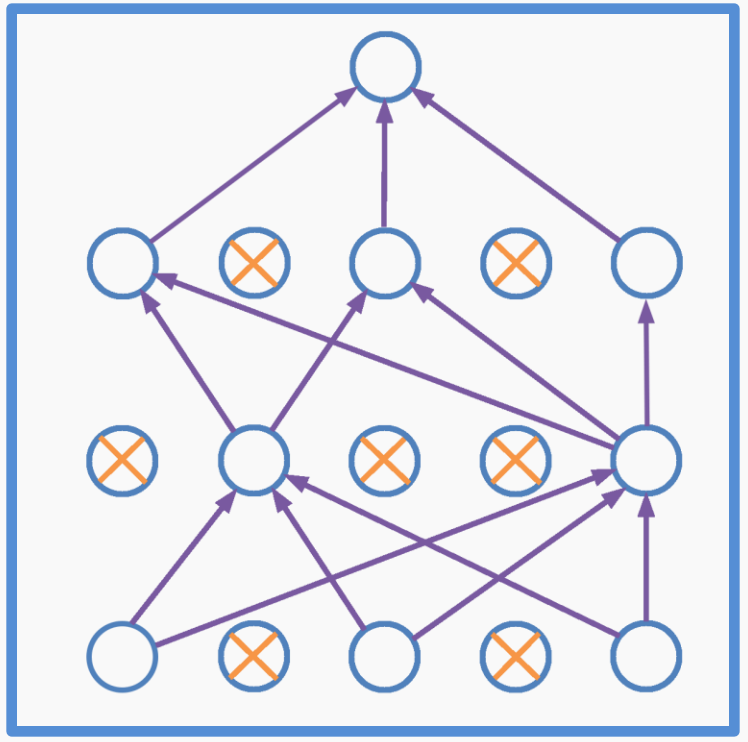
D. Force some of the neurons not to participate randomly per batch

Dropout

- Randomly set some neurons and their connections to zero (i.e. “dropped”)
- Prevent overfitting by reducing co-adaptation of neurons
- Like training many random sub-networks



Standard Neural Network



After Applying Dropout

Dropout

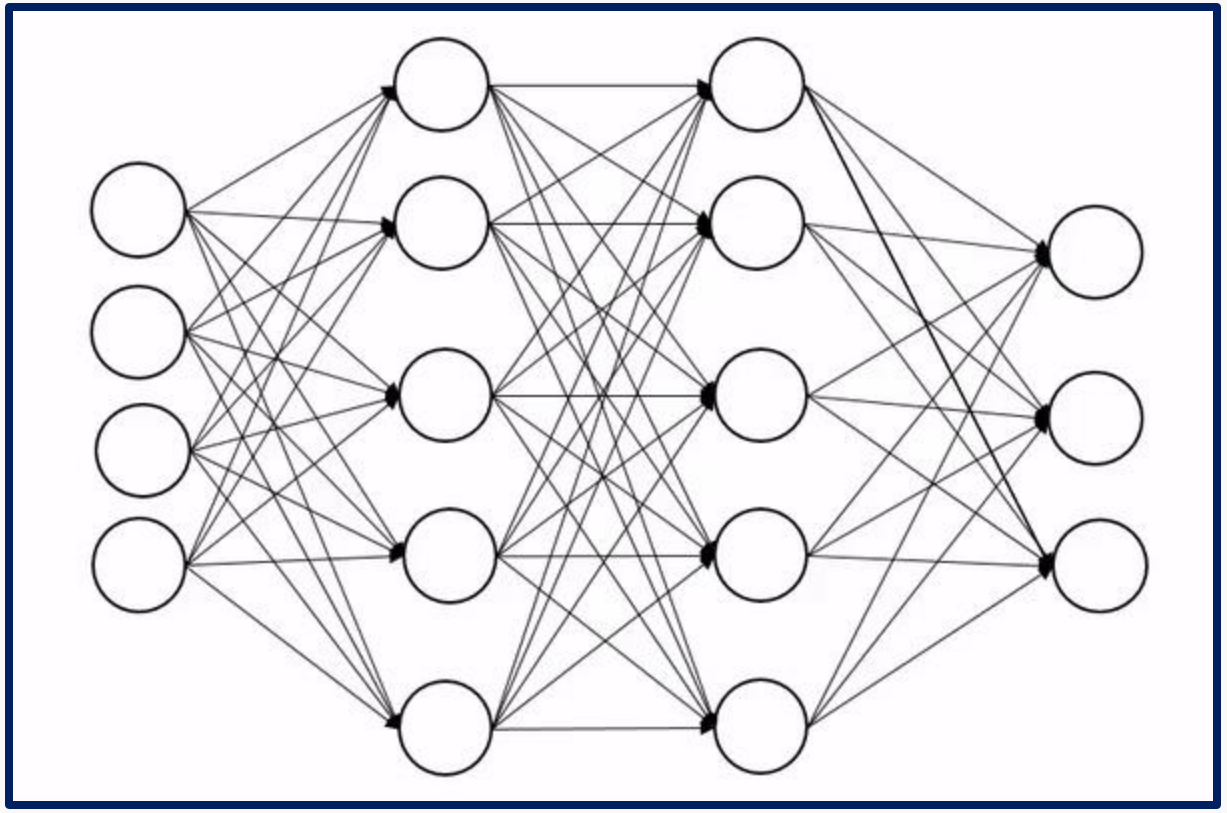
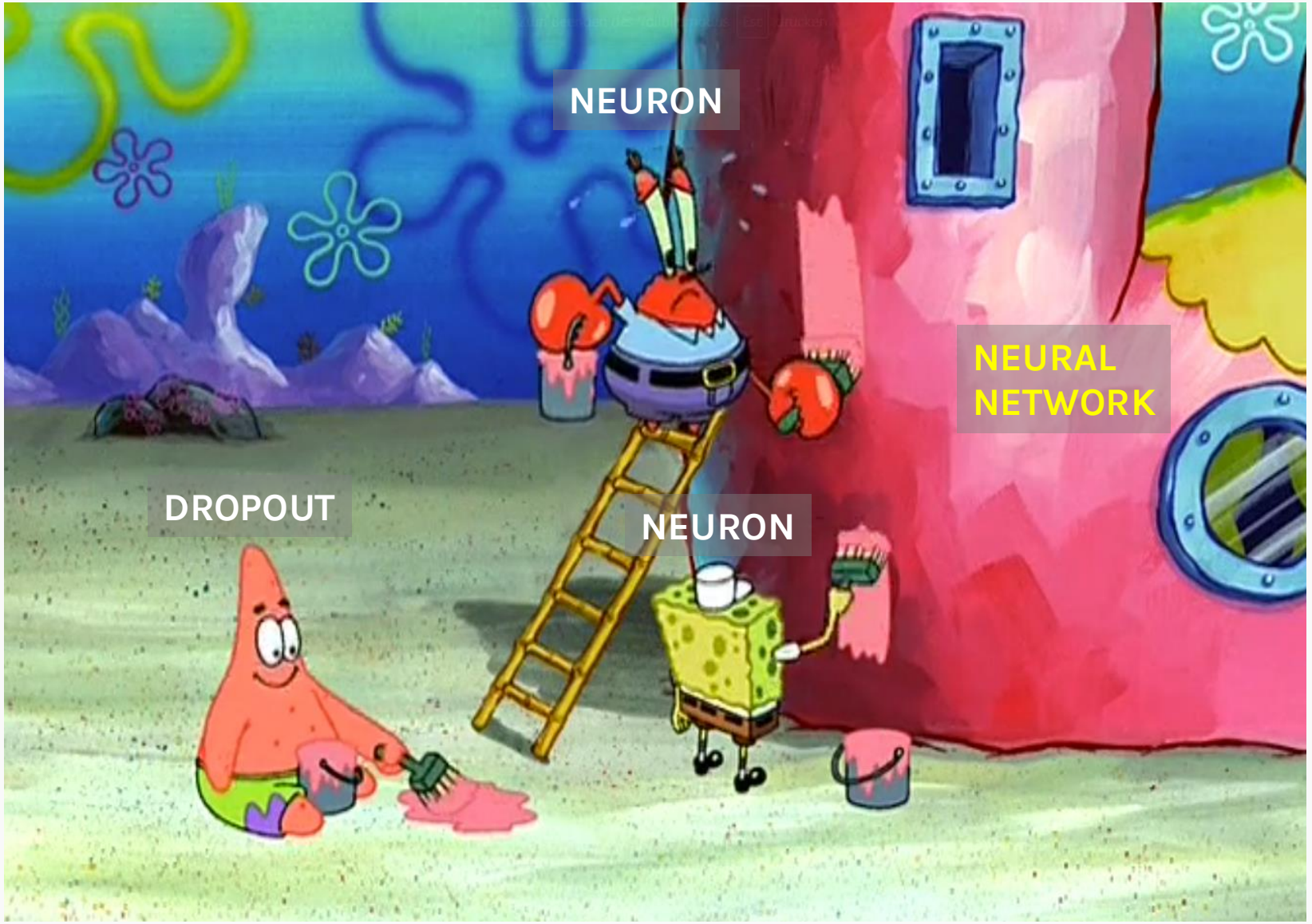


Image Source


NEURON
NEURAL NETWORK
DROPOUT
NEURON

Dropout | Training

For each new example in a mini-batch (could be for one mini-batch depending on the implementation):

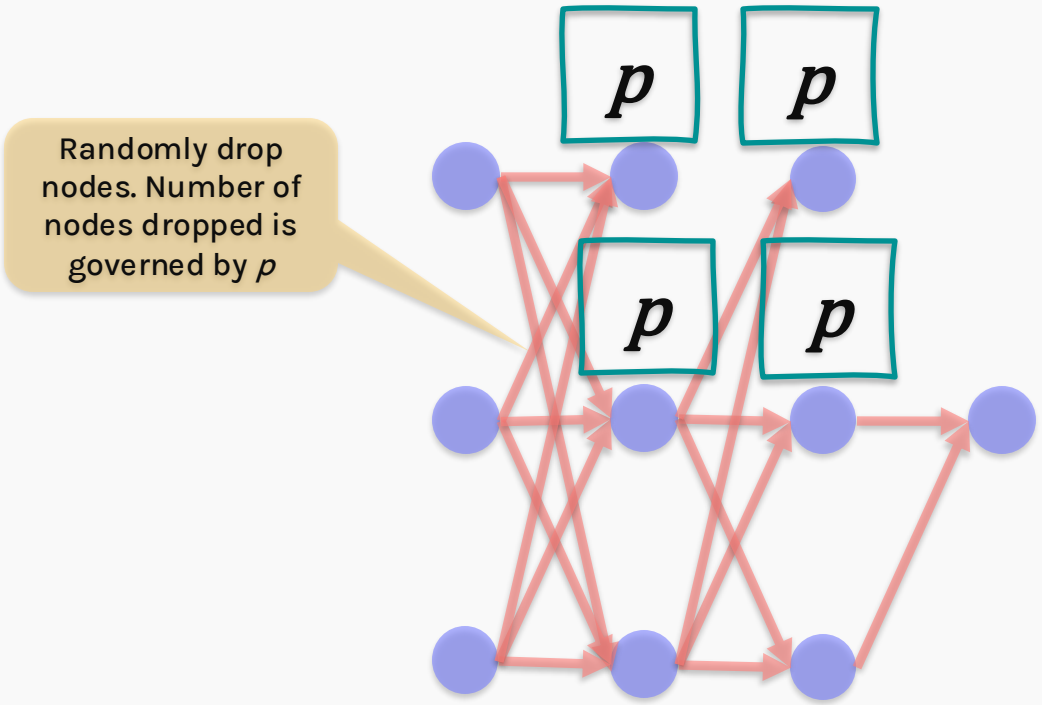
- Randomly sample a binary mask μ independently, where μ_i indicates if input/hidden node i is included
- Multiply output of node i with μ_i, and perform gradient update

Typically:

- Input nodes are included with prob=0.8 (as per original paper, but rarely used)
- Hidden nodes are included with prob=0.5

Dropout | Prediction

- We can think of dropout as training many of sub-networks



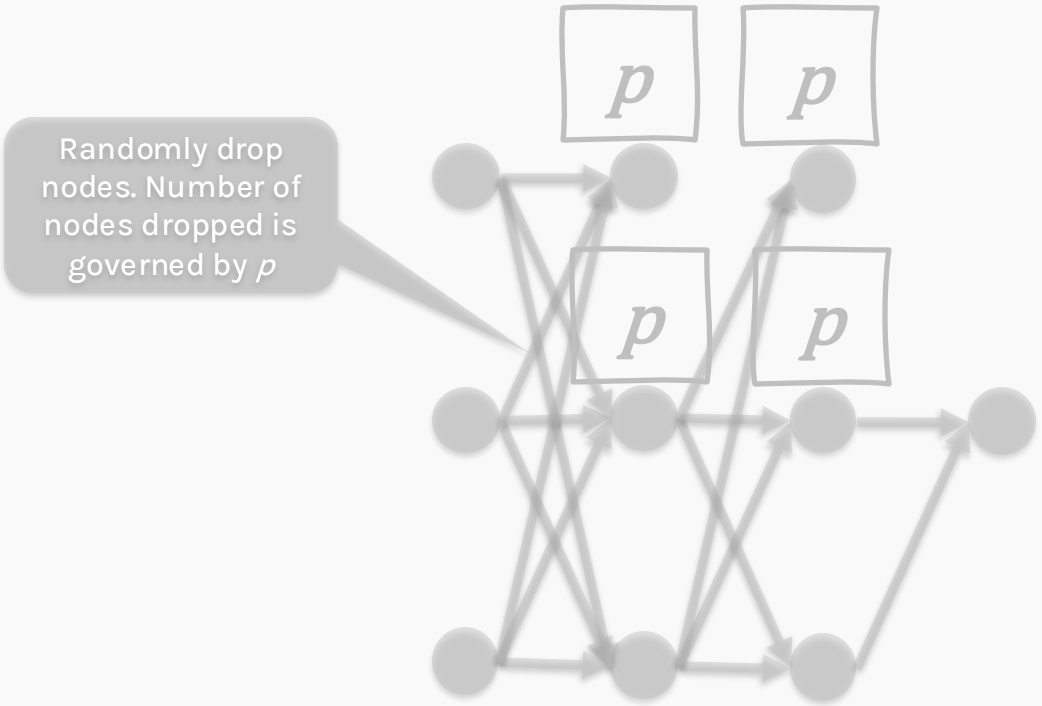
(a) At training time

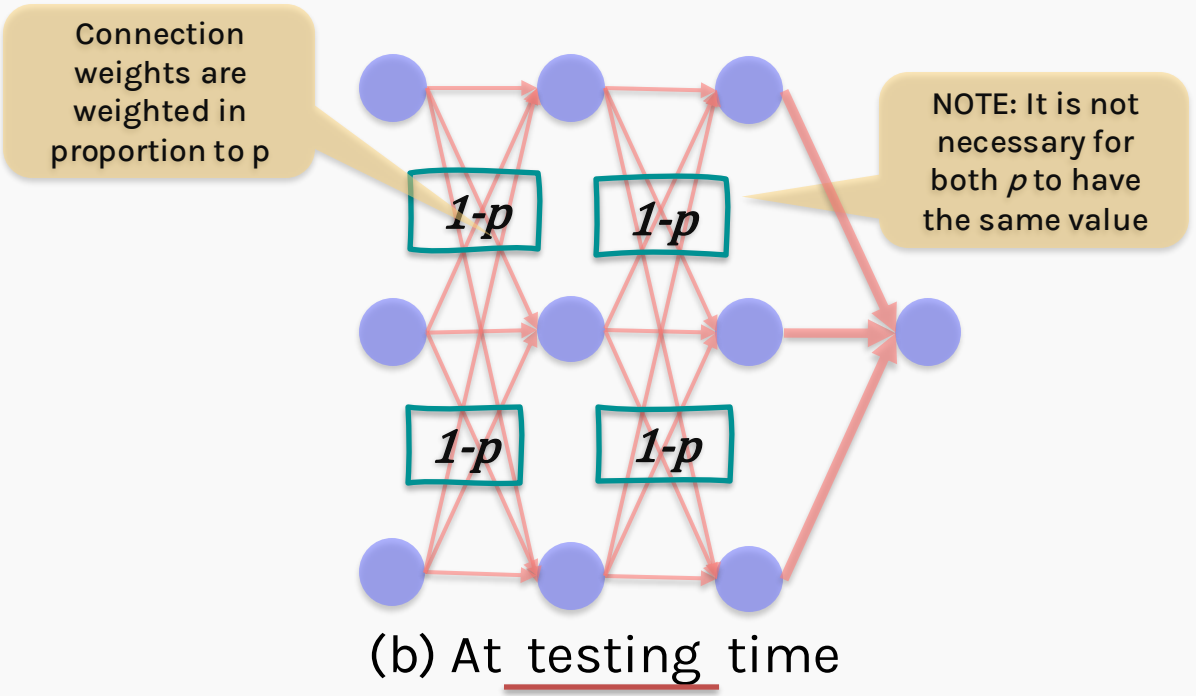
Dropout | Prediction

What do you think occurs at testing time?

Dropout | Prediction

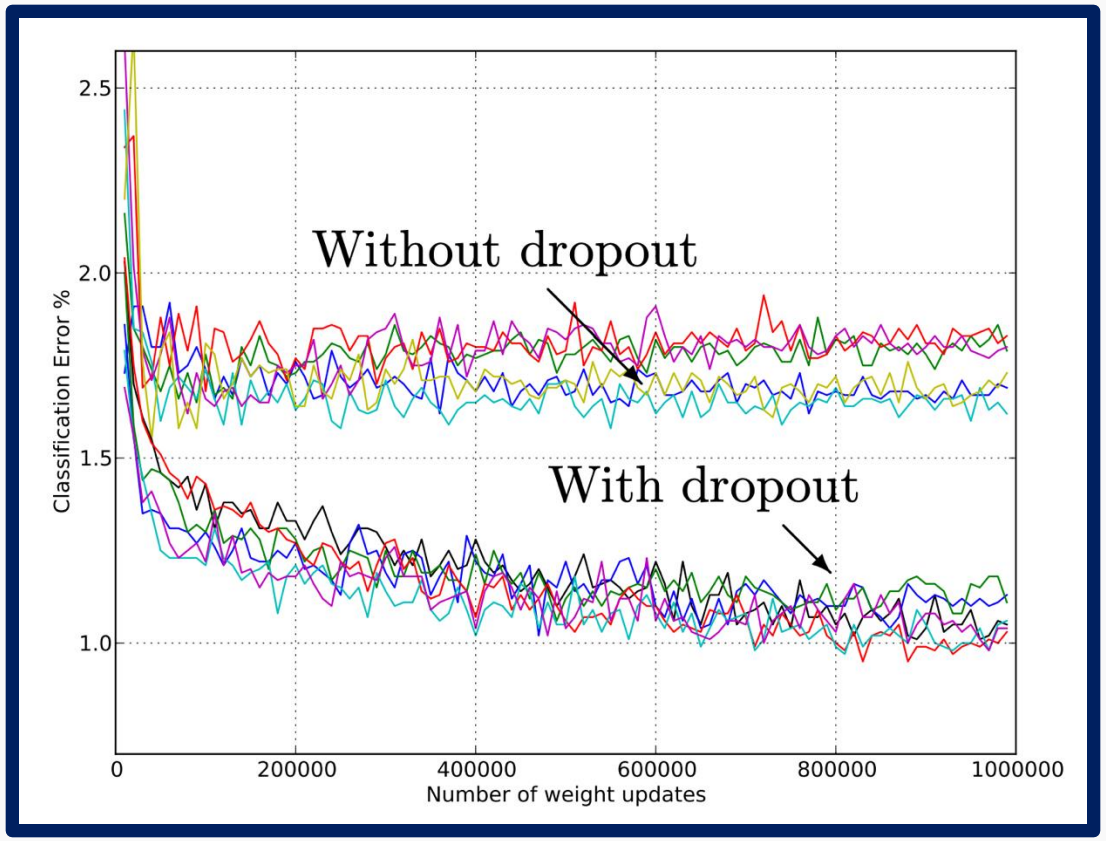
- At **test time**, we can "aggregate" over these sub-networks by **reducing connection weights in proportion to dropout probability, *p***





Dropout

- Widely used and highly effective

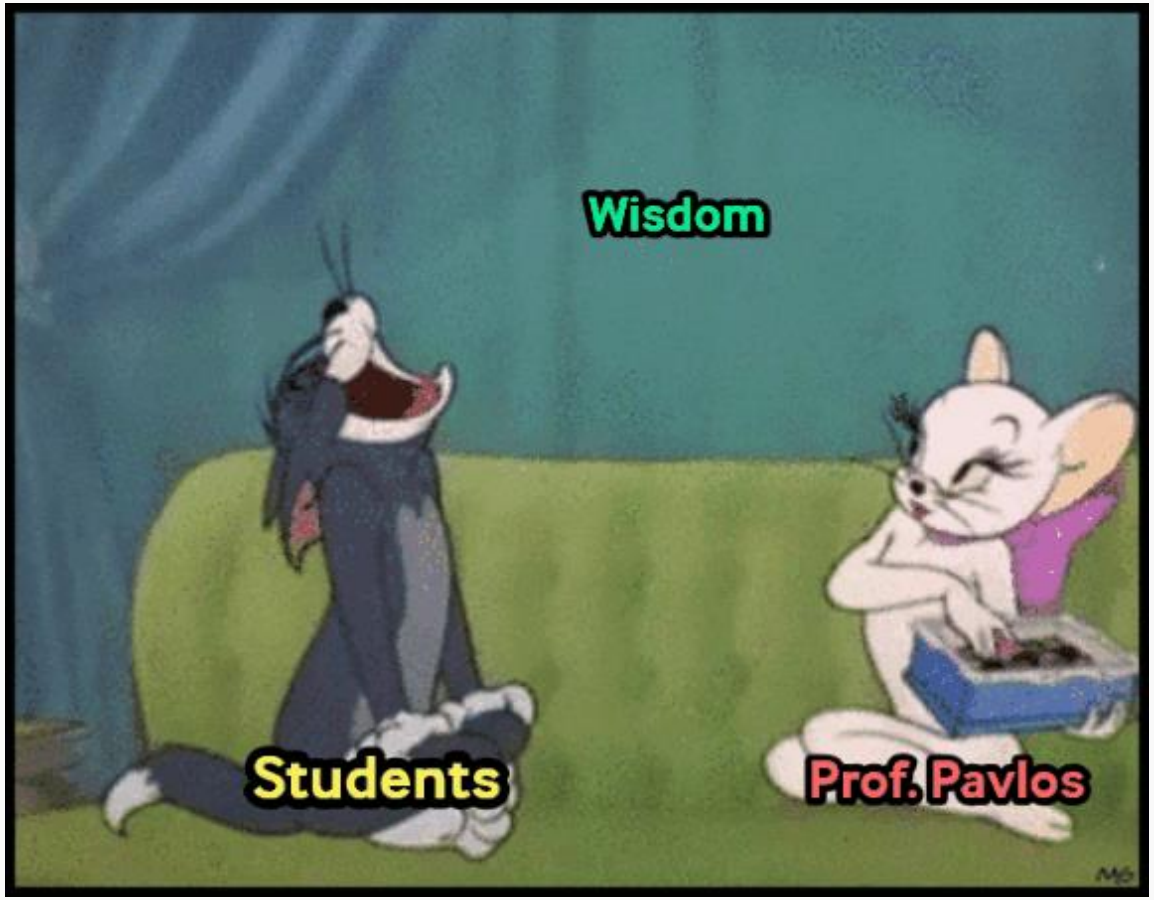


Test error for different architectures with and without dropout.

The networks have 2 to 4 hidden layers each with 1024 to 2048 units.

- Proposed as an alternative to ensemble methods, which is too expensive for neural nets

Srivastava, N., Hinton, G., et al . (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929-1958.



Thank you