Intro to Neural Networks

BA865 – Mohannad Elhamod

BOSTON UNIVERSITY

Boston University Questrom School of Business

CNNs

Convolutional Networks

A Problem of Scalability

- How many parameters in this network?
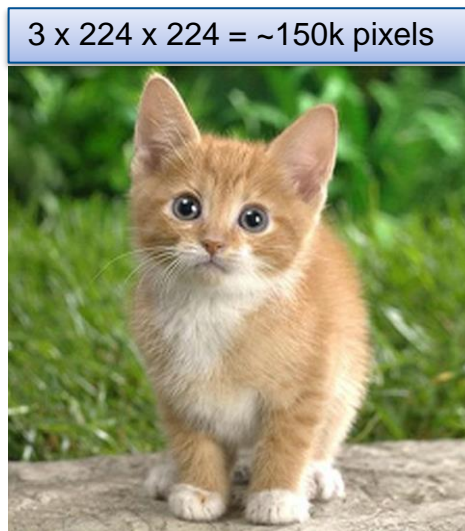- Do we really need to learn all these parameters?

3 x 224 x 224 = ~150k pixels
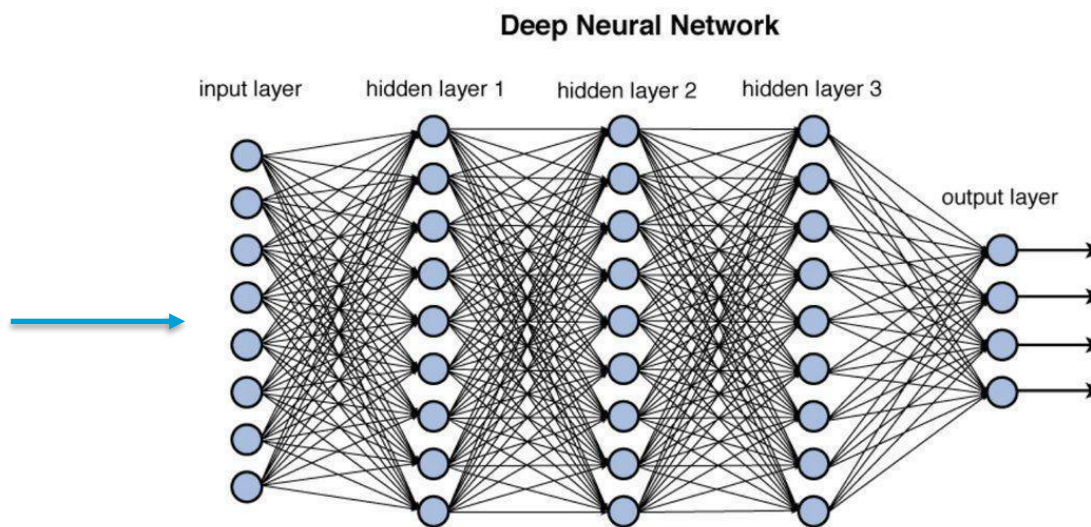

Figure courtesy of Robert Bond

Deep Neural Network

input layer   hidden layer 1   hidden layer 2   hidden layer 3   output layer

Figure 12.2 Deep network architecture with multiple layers.

Figure courtesy of Ravindra Parmar

Boston University Questrom School of Business

BOSTON UNIVERSITY

Structure in Images

- Interesting images have:
  - Locality of information.
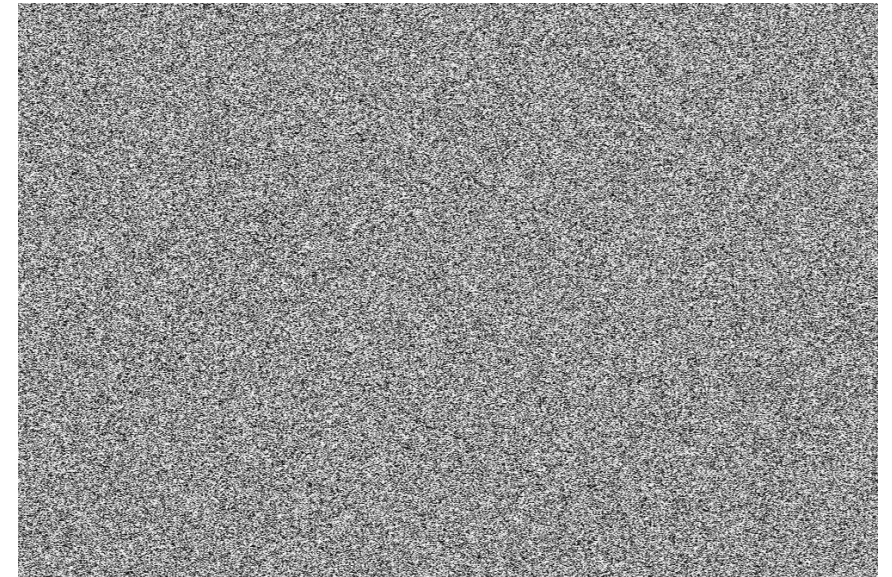 - Spatial invariance.

Figure courtesy of Robert Bond

vs.

Figure courtesy of Jorge Stolfi

Boston University Questrom School of Business

Convolutional Filters

- Instead of learning a mesh of all possible parameters, let's learn local descriptors *(kernels or filters)* that can be reused across the image!
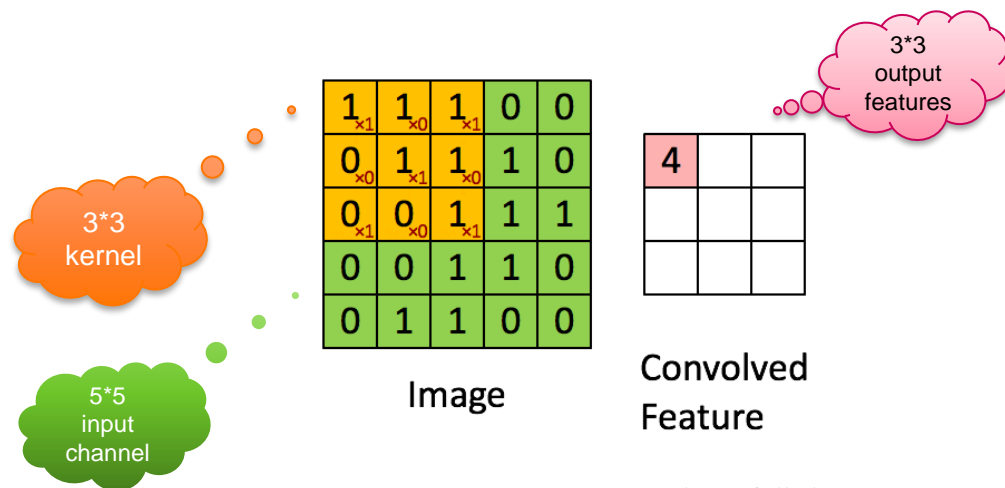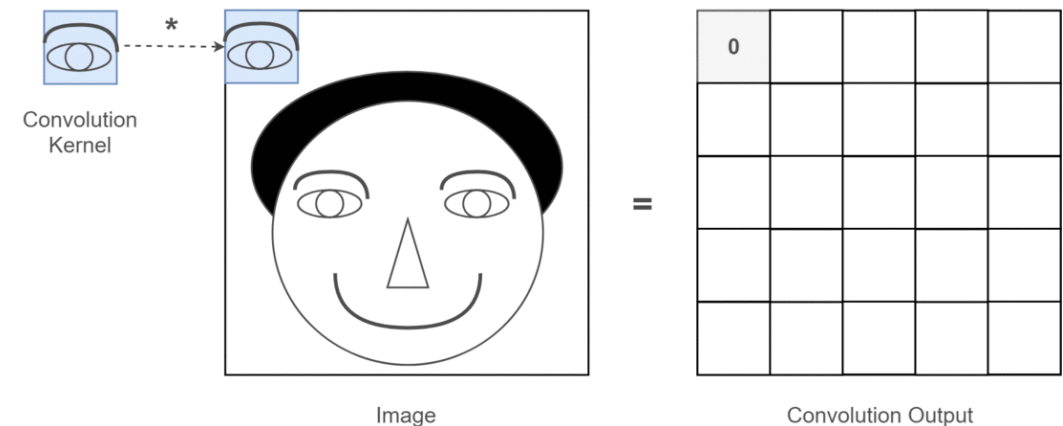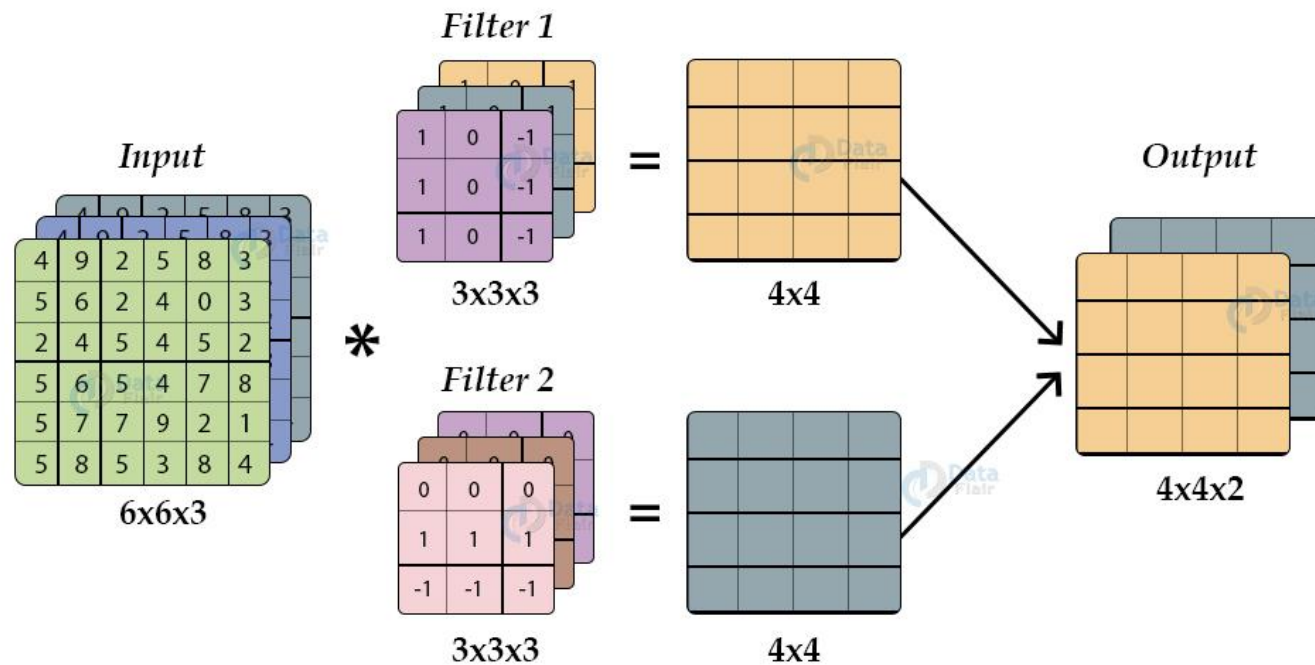

Figure courtesy of Daniel Nouri

Figure courtesy of Thushan Ganegedara

Boston University Questrom School of Business
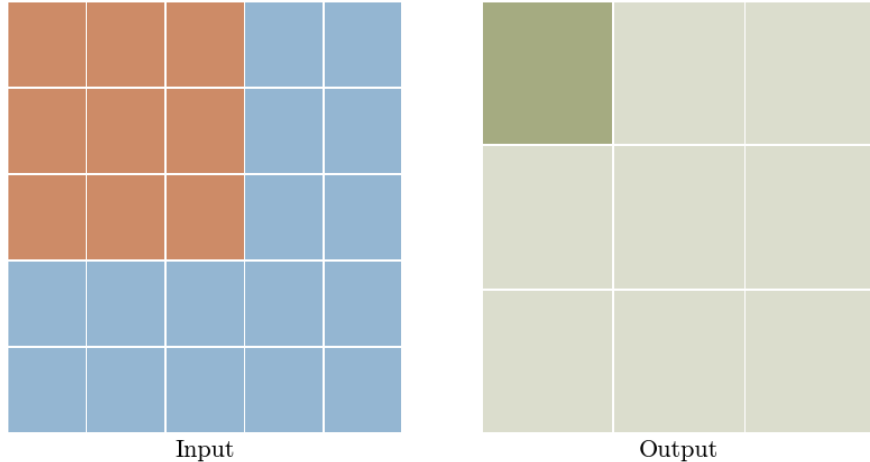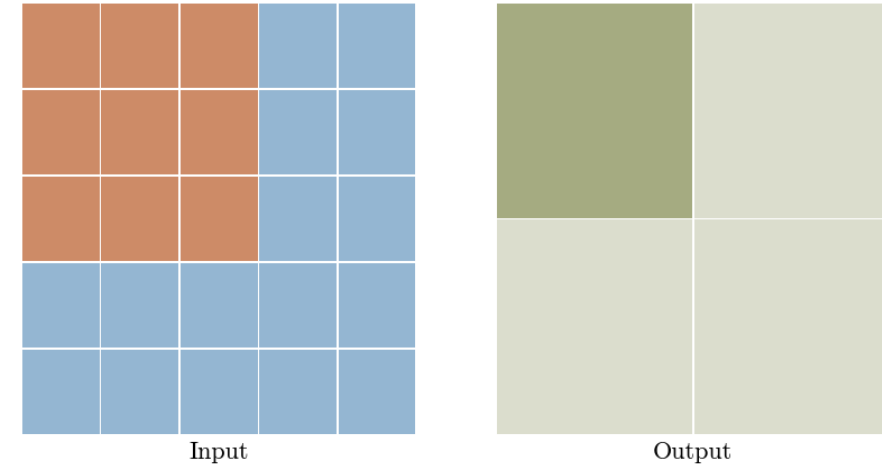
Mathematically Speaking...

Convolution Layer in Keras

Stride and Padding

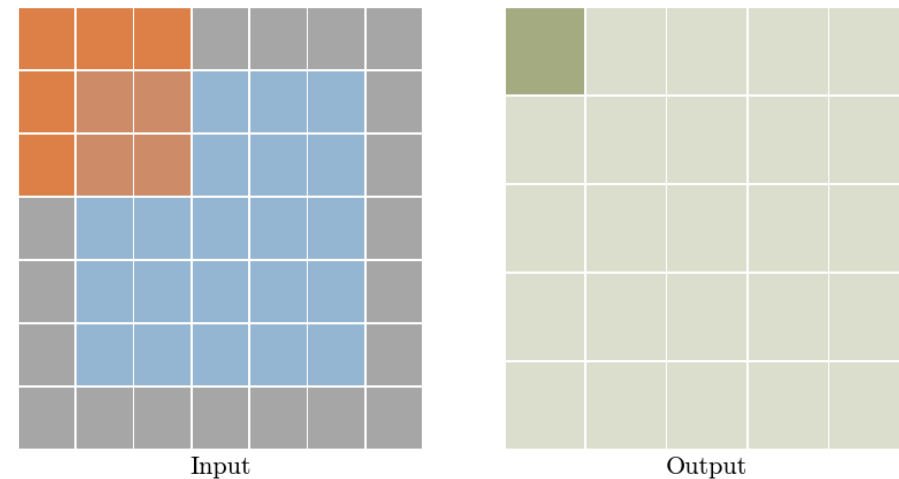- Stride controls feature field overlap.
- Padding controls the down-sampling.

Type: conv - Stride: 2 Padding: 0
Input   Output
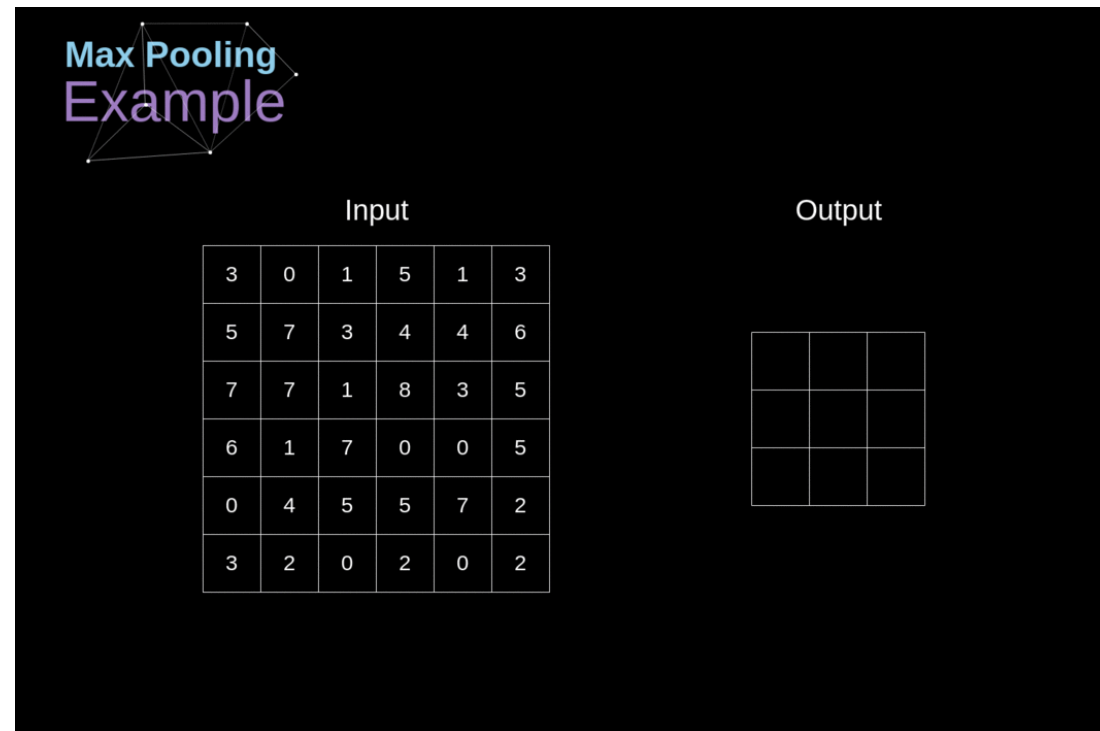

Type: conv - Stride: 1 Padding: 0
Input   Output

Type: conv - Stride: 1 Padding: 1
Input   Output

Aqeel Anwar

Boston University Questrom School of Business

Non-Linearity: MaxPooling

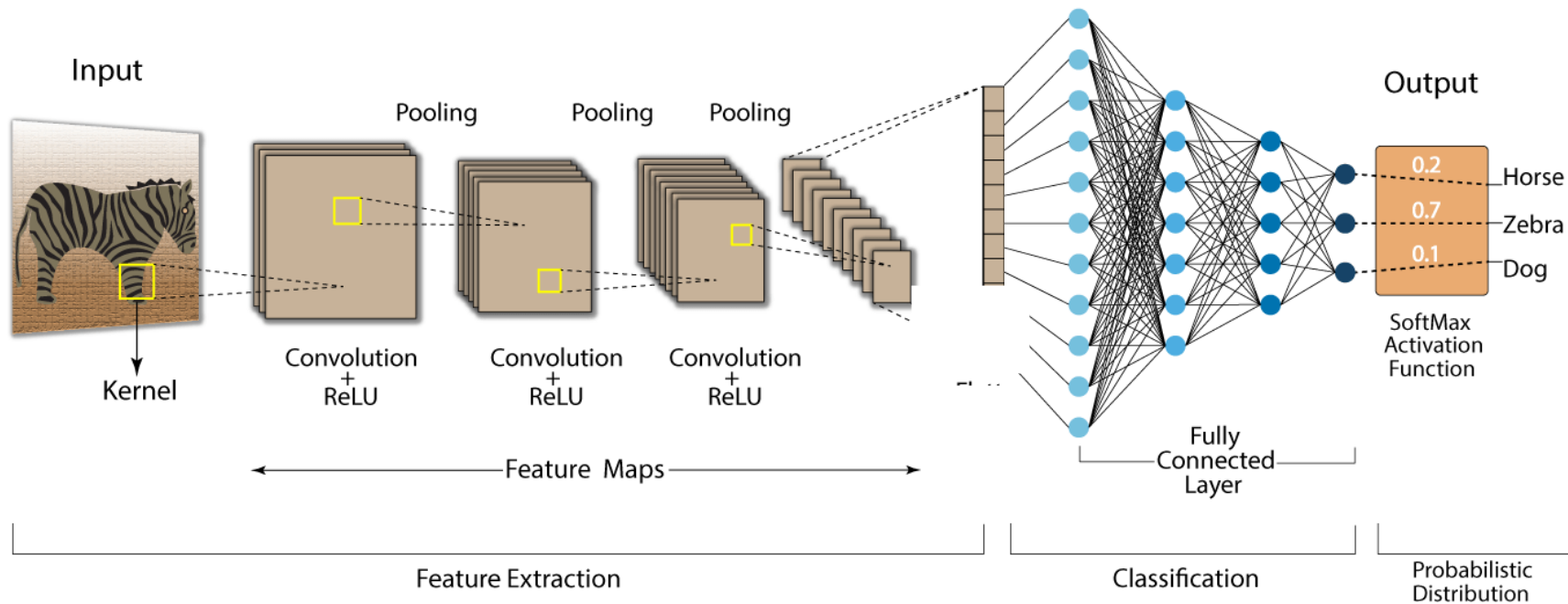- In addition to being a non-linearity…
  - it helps down-sample the image.
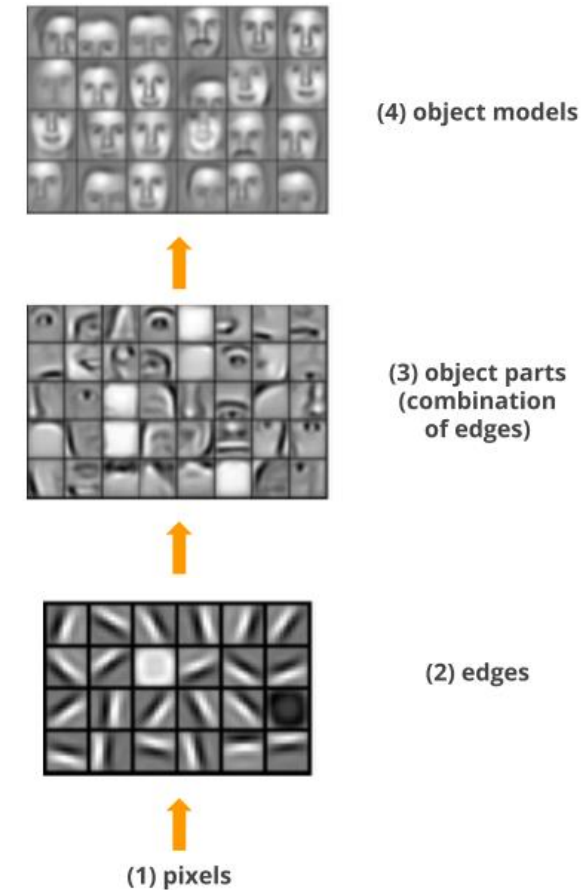 - It helps summarize information in terms of larger blocks.

Boston University Questrom School of Business

Putting It All Together

- Deeper layers generally have more kernels that are smaller.

Boston University Questrom School of Business

Learned Features

- Early layers learn low-level features.

  - spots, edges, etc.

- Later layers learn to detect high-level features as a combination of low-level features.

  - Eyes, ears, hair, etc.

- Demo

(4) object models

(3) object parts (combination of edges)

(2) edges

(1) pixels

https://micro-dimensions.com

Hyper-Parameters

Continued…

Boston University Questrom School of Business

Batch Normalization

- Even if input data is properly normalized, the gradient in subsequent layers may vanish or explode.

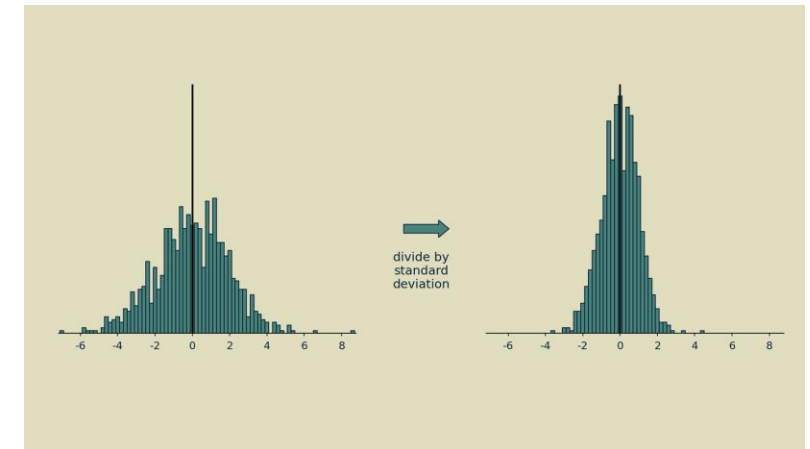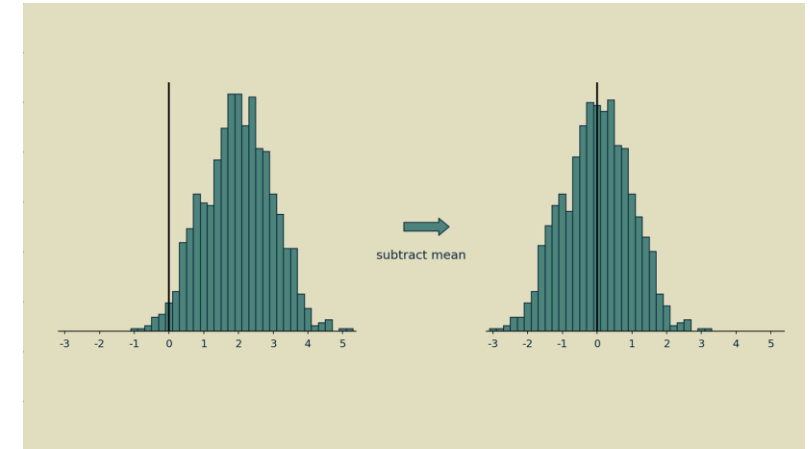- For that, you may add "*batch normalization*" at every layer.

Figure courtesy of Brandon Rohrer

Boston University Questrom School of Business

Learning Rate: Schedulers

- Since larger learning rates may converge faster but smaller ones are more stable, you could adjust the learning rate in phases to get the best of both worlds!

  - This way, you still converge but faster.
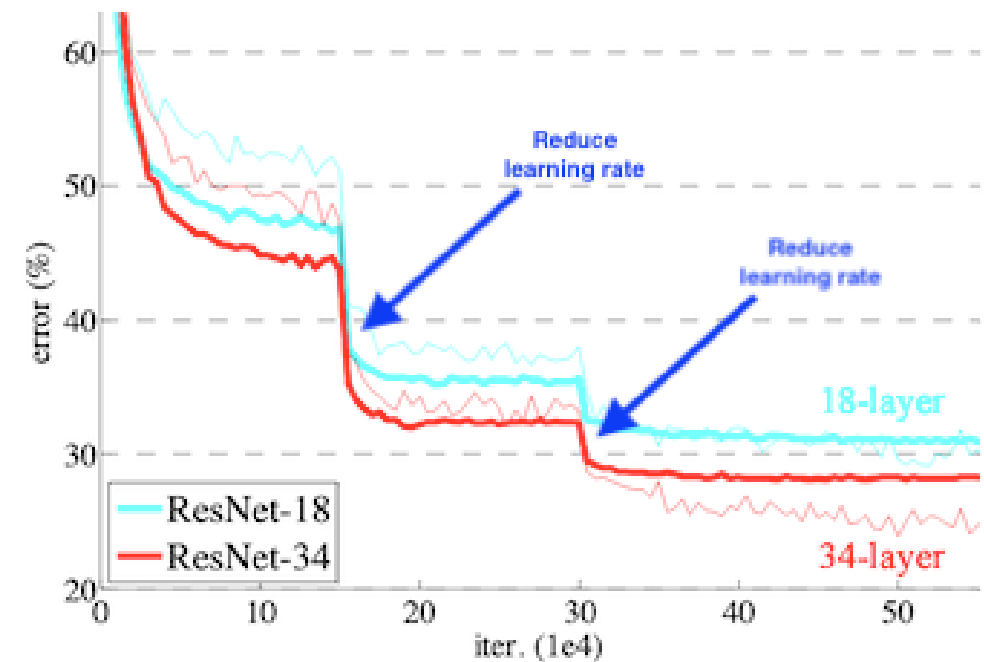
- Using a scheduler is a common practice.

Figure courtesy of B. D. Hammel

Boston University Questrom School of Business

Hyper-Parameter Tuning

Boston University Questrom School of Business

Be Smart About It

- ## It is expensive!

  - 1 hyper-parameter with 3 values → 3 experiments

  - 2 hyper-parameter with 3 values each → 9 experiments

  - 3 hyper-parameter with 3 values each → 27 experiments

  - … exponential growth!
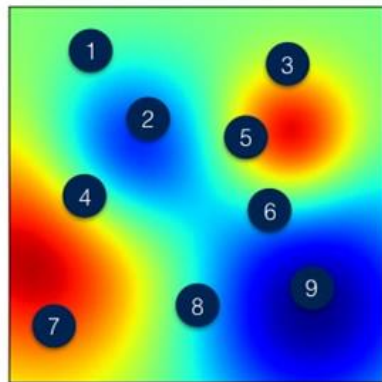
Be Smart About It

- It is expensive!
- Start with generally accepted wisdom:
  - Start with good initial guesses.
  - Different settings work better for different models/problems (e.g., SGD + momentum for computer vision vs. Adam otherwise)
- Be picky about what to fine-tune.
  - Use early stopping.
  - Learning rate is the most important parameter!
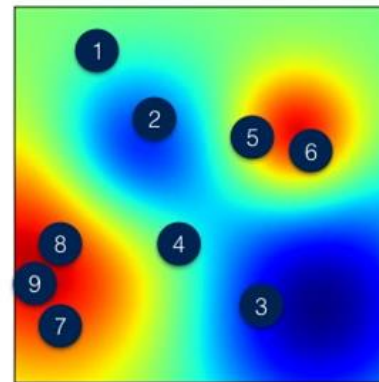
Hyper-Parameter Tuning Methods

- Generally, use log-scale for numerical hyper-parameters.
- Random and Adaptive searches generally find optimal values faster than grid searches.
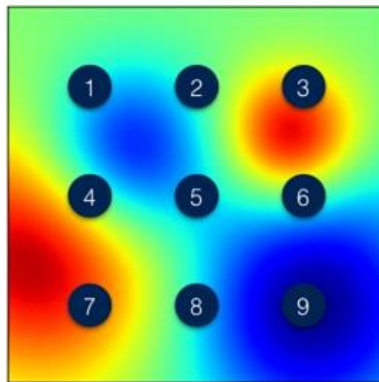
Grid Search   Random Search   Adaptive Selection

Figure courtesy of Liam Li

Boston University Questrom School of Business