Hyper-Parameters: control parameters (W and b)

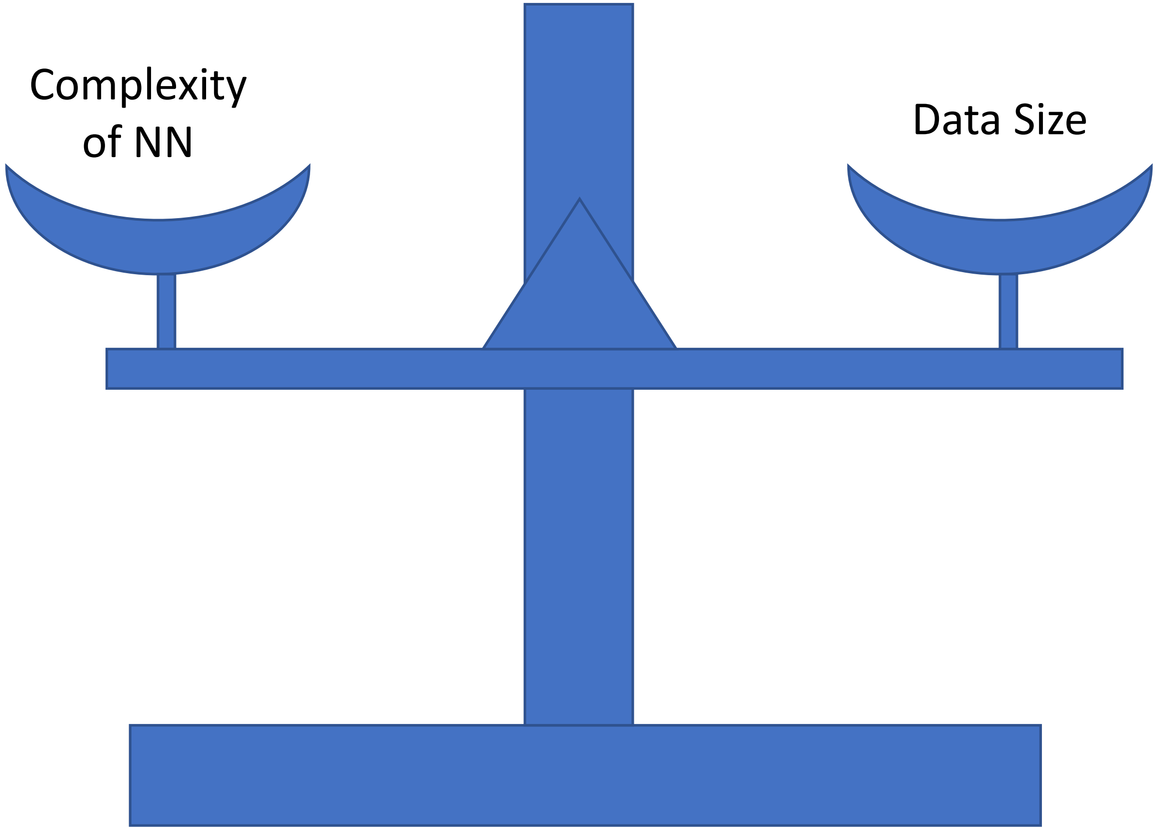
- # hidden layers
- # hidden units
- Learning rates
- Activation functions
- Train / Dev / Test Distribution
- A λ: Regularization

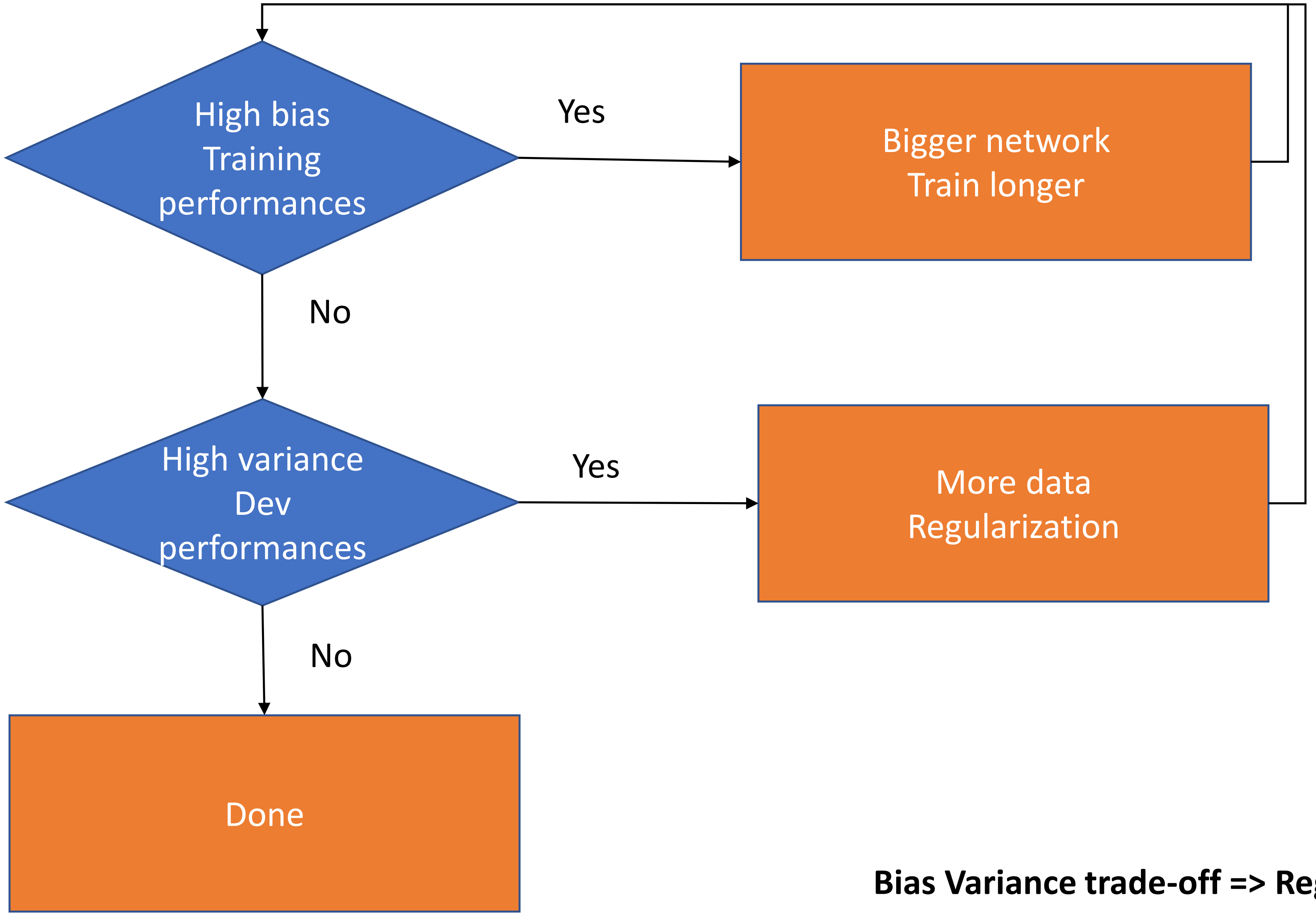
- For big datasets:
  - 98% train – 1% Dev – 1% test

- For small datasets:
  - 70% train – 30% Test

Simpler Neural Network – Higher Data size

- L2 – regularization
- Dropout regularization
- Data augmentation
- Early stopping


Complexity
of NN
Data Size



Bias Variance trade-off => Regularization

Regularization

- Definition of cost function $J(W,b) = \sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)})$
- L1 - Regularization $J(W,b) = \sum_{i=1}^{m} L\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m} \|w\|_1$
- L2 - Regularization $J(W,b) = \sum_{i=1}^{m} L\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m} \|w\|_2^2$
- With $\|w\|_1 = \sum_{i=1}^{n_x} w_j$
- With $\|w\|_2 = \sum_{i=1}^{n_x} w_j^2 = w^T w$

For case of Deeper neural network

- L2 - Regularization $J(W,b) = \sum_{i=1}^{m} L\left(\hat{y}^{(i)}, y^{(i)}\right) + \frac{\lambda}{2m} \sum_{l=1}^{L} \left\| w^{[l]} \right\|_F^2$
- Frobenius norm:

$$\left\| w^{[l]} \right\|_F^2 = \sum_{i=1}^{n^{[l]}} \sum_{j=1}^{n^{[l-1]}} \left(w_{ij}^{[l]} \right)^2$$

$$w^{[l]} : \left(n^{[l]}, n^{[l-1]} \right)$$

The λ coefficient should be integrated just on dev set

- $w^{[l]} \coloneqq w^{[l]} - \ \alpha \, dw^{[l]} - (\alpha\lambda/2\text{m})w^{[l]}$
- If λ is too big so $w^{[l]} \rightarrow 0$.
- Many of hidden unit will be null => much simple neural network
- $z^{[l]} = w^{[l]}a^{[l-1]} + b^{[l]}$ will be linear => Faster computation.

Dropout regularization

- Keep neurones on a layer just that probability is less than keep-prob
- Permit to reduce complexity of neural network
- It shouldn’t be done in test time

Data augmentation

- Flip in different senses for images for example
- Take random crops of the image
- Add some distortions to image

Early stopping

- Stopping training when cost function is min on Dev set