

Data Splits and Neural Network Performance Evaluation

Course 3, Module 3, Lesson 4

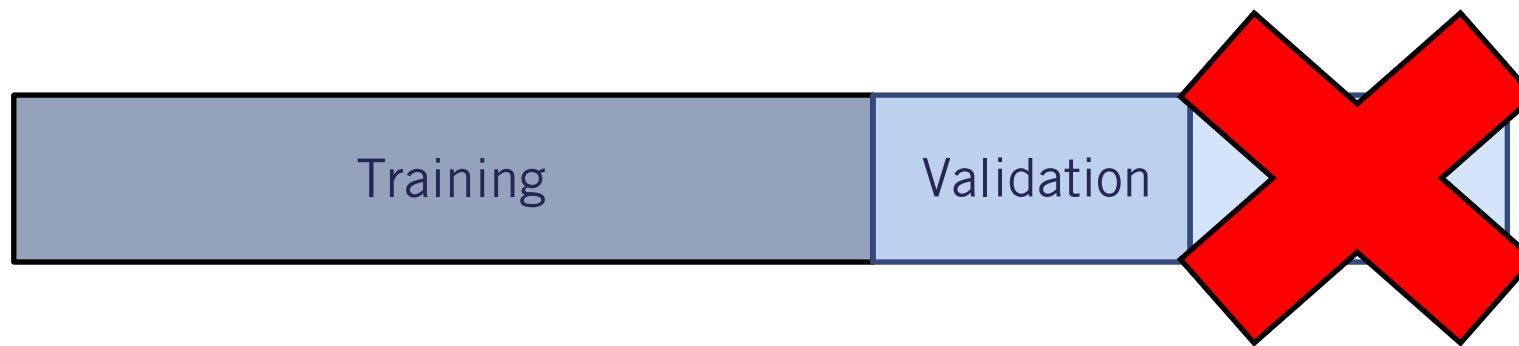


UNIVERSITY OF TORONTO
FACULTY OF APPLIED SCIENCE & ENGINEERING

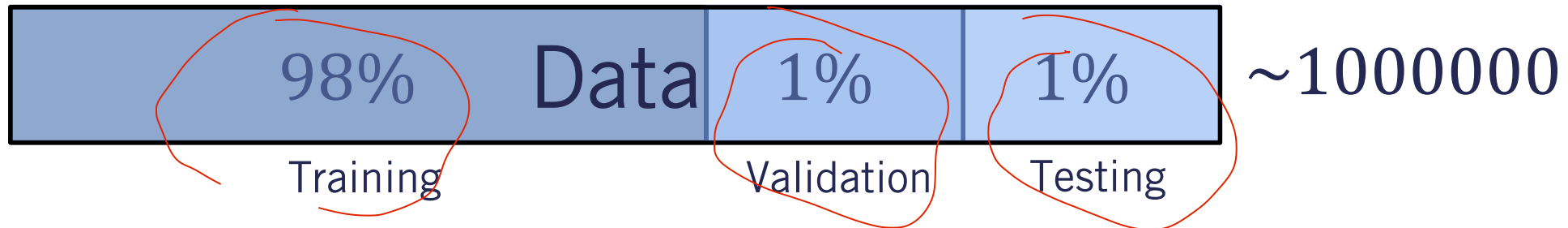
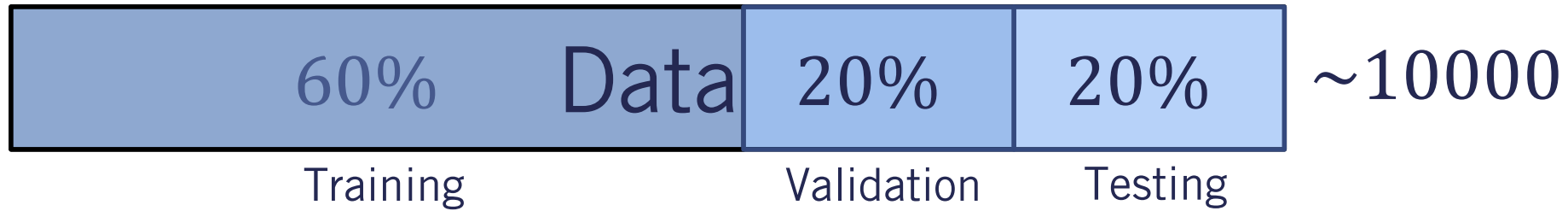
Data Splits

- **Training Split:** used to minimize the Loss Function
- **Validation Split:** used to choose best **hyperparameters**, such as the learning rate, number of layers, etc.
- **Test Split:** the neural network **never observes** this set. The developer **never** uses this set in the design process

*not learned
during
training.*



Data Splits




Behavior of Split Specific Loss Functions

| | 6000 | 2000 | 2000 | |
|----------------|---------------------|-------------------|--------------------|-----------------------|
| | Training | Validation | Testing | ~10000 |
| | $J(\theta)_{train}$ | $J(\theta)_{val}$ | $J(\theta)_{test}$ | $J(\theta)_{Minimum}$ |
| Good Estimator | 0.21 | 0.25 | 0.30 | 0.18 |
| Underfitting | 1.9 | 1.9 | 2.1 | |
| Overfitting | 0.21 | 2.05 | 2.1 | |

loss function

generalization gap

Reducing the Effect of Underfitting/Overfitting

- **Underfitting:** (Training loss is high)
 - Train longer
 - More layers or more parameters per layer
 - Change architecture
-  **Overfitting:** (Generalization gap is large)
 - More training data
 - Regularization
 - Change architecture

Summary

- A dataset should be split to a training, a validation and a test split
- Observing the performance on each of these splits helps in determining why a neural network is not performing well in the real world
- **Underfitting:** Train longer or use a larger neural network
- **Overfitting: Regularization**