Best Practices for Training ANNs

Heni Ben Amor, Ph.D.
Assistant Professor
Arizona State University

Ira A. Fulton Schools of Engineering
Arizona State University

Training with Back Propagation

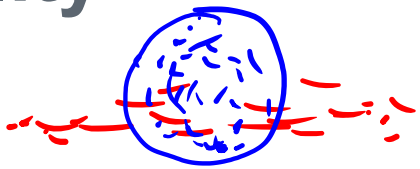
- Given a set of training data
- Each sample a tuple $< \mathbf{x}, \mathbf{y} >$
- Where $\mathbf{x}$ is the input and $\mathbf{y}$ is the desired output
- Train network such that

$$NN(\mathbf{x}) \approx \mathbf{y} \quad \forall \mathbf{x} \in \mathbf{X}$$

- Assumes labeled training data
- Typically labels are provided by human annotation

Input Normalization

- Normalizing input values to the same range helps improve learning quality
- Normalization: for input data, do

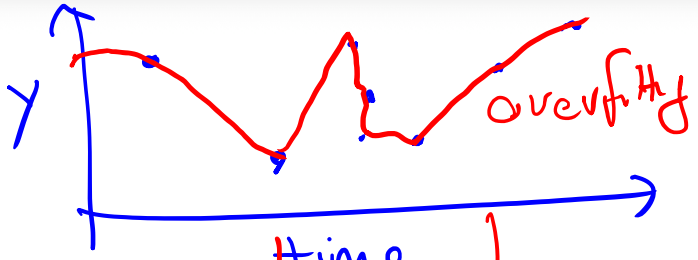


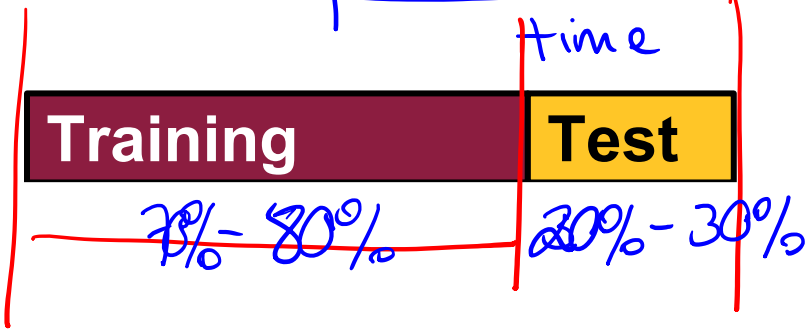
$$\mu = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i \qquad \sigma^2 = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \mu)^2$$

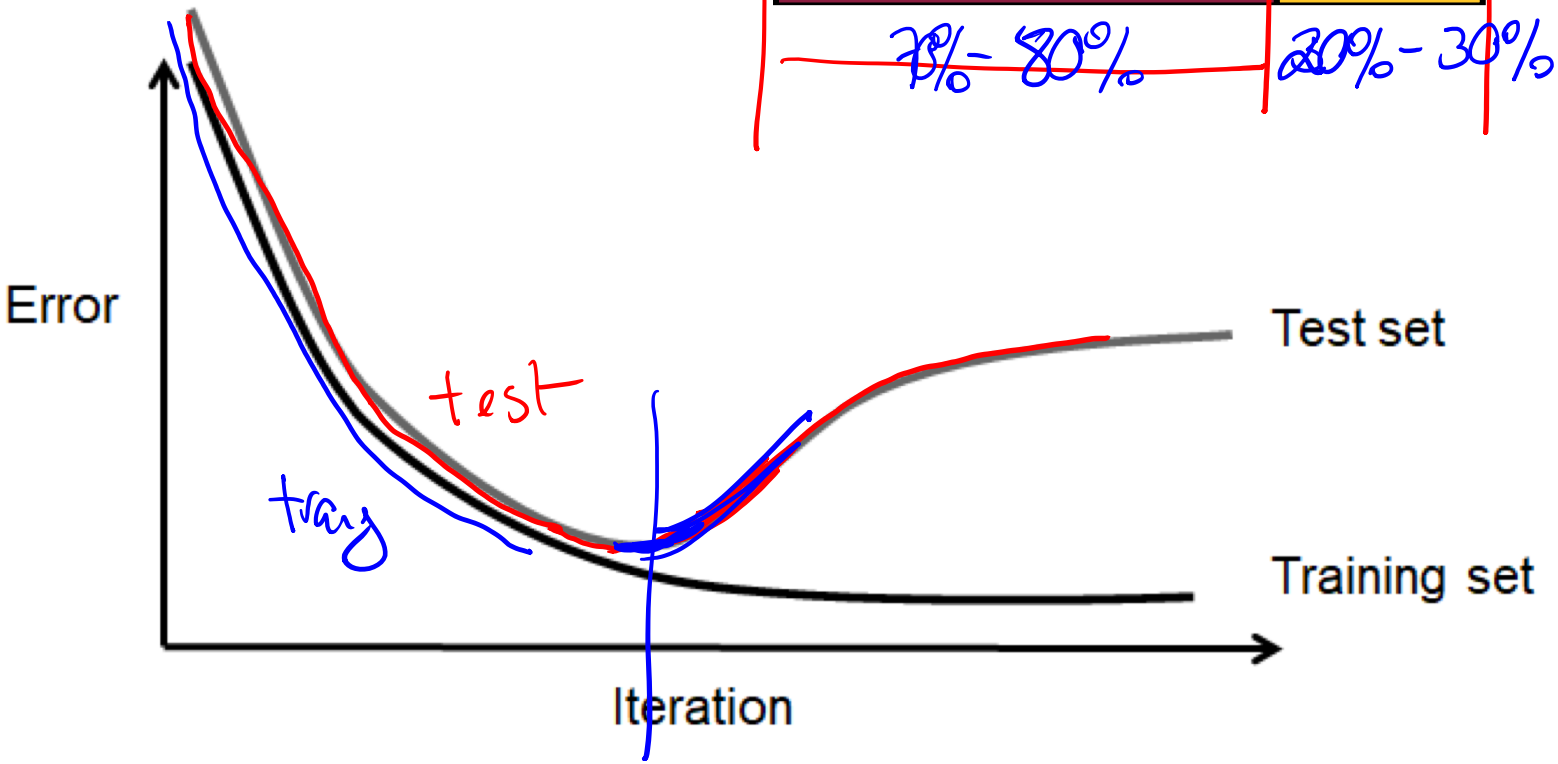
$$\hat{x}_i = \frac{\mathbf{x}_i - \mu}{\sqrt{\sigma^2}}$$

Ensuring Generalization

- **Overfitting** to data is bad
- Early stopping







K-Fold Cross-validation

- Divide data in K-folds
- Train and test on remaining fold

- True error is average of individual errors

Summary

- Prepare your data and experiments
- Normalize your input data
- Make sure you have train and test data
- Use cross-validation to get a better estimate of network performance
- Beware of overfitting, use early stopping