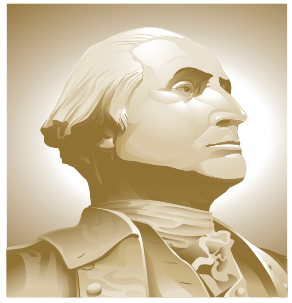
Generalization

Machine Learning II

Lecture 6

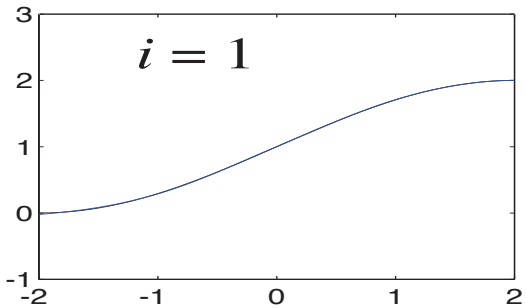


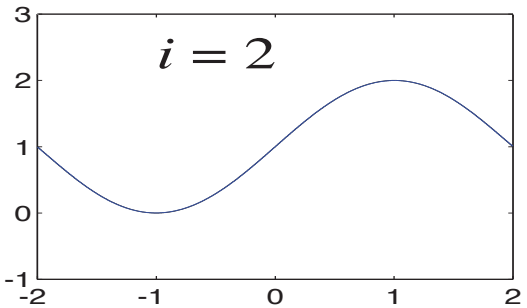
THE GEORGE WASHINGTON UNIVERSITY

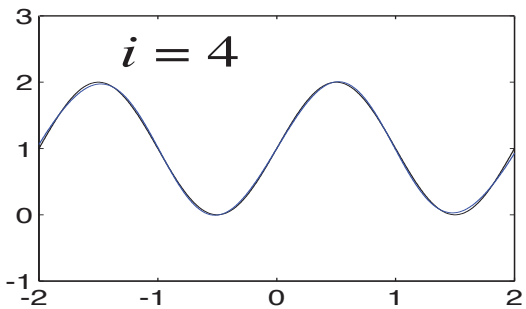
WASHINGTON, DC

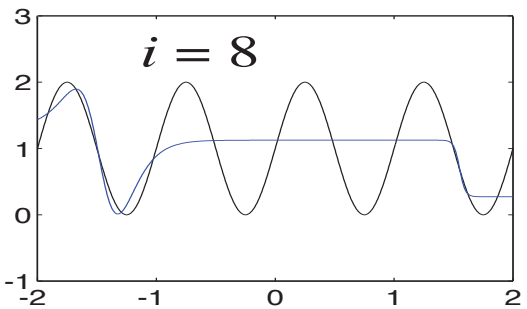
$$g(p) = 1 + \sin\left(\frac{i\pi}{4}p\right)$$

1-3-1 Network



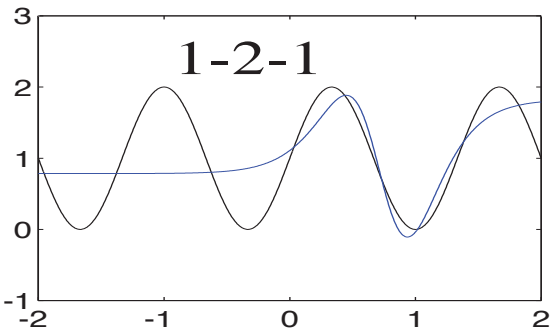


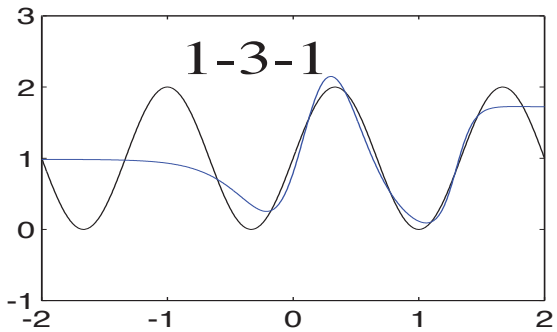


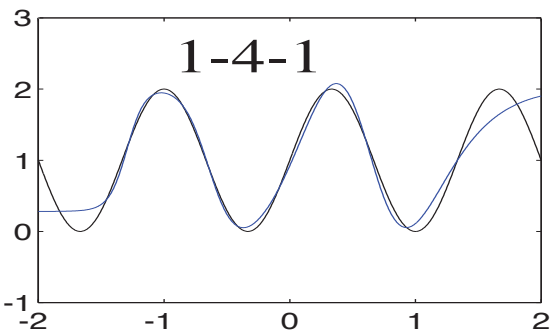


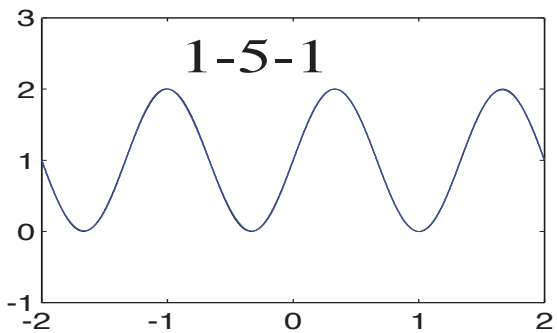
Choice of Network Architecture

$$g(p) = 1 + \sin\left(\frac{6\pi}{4}p\right)$$



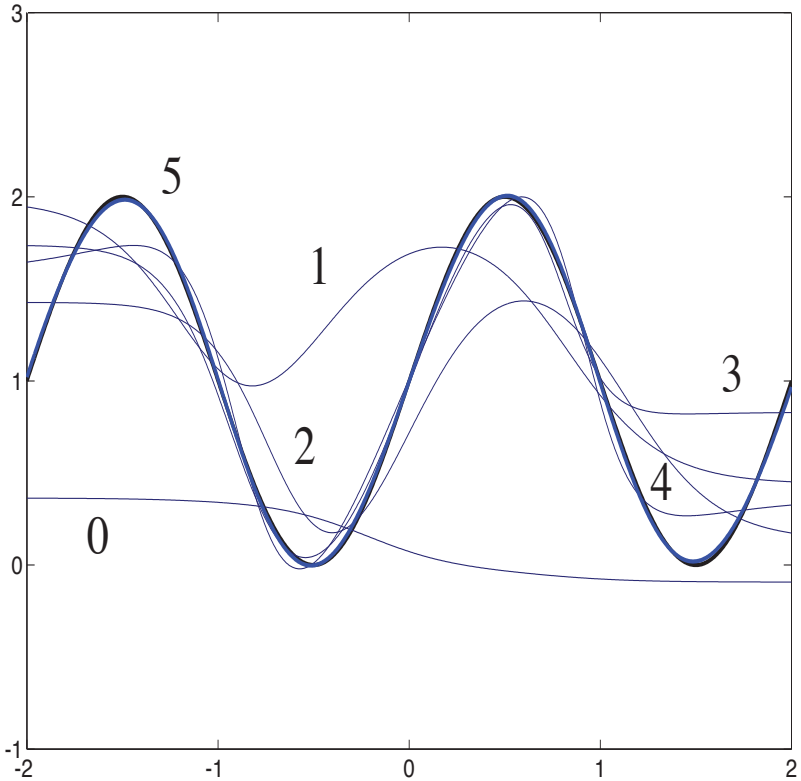


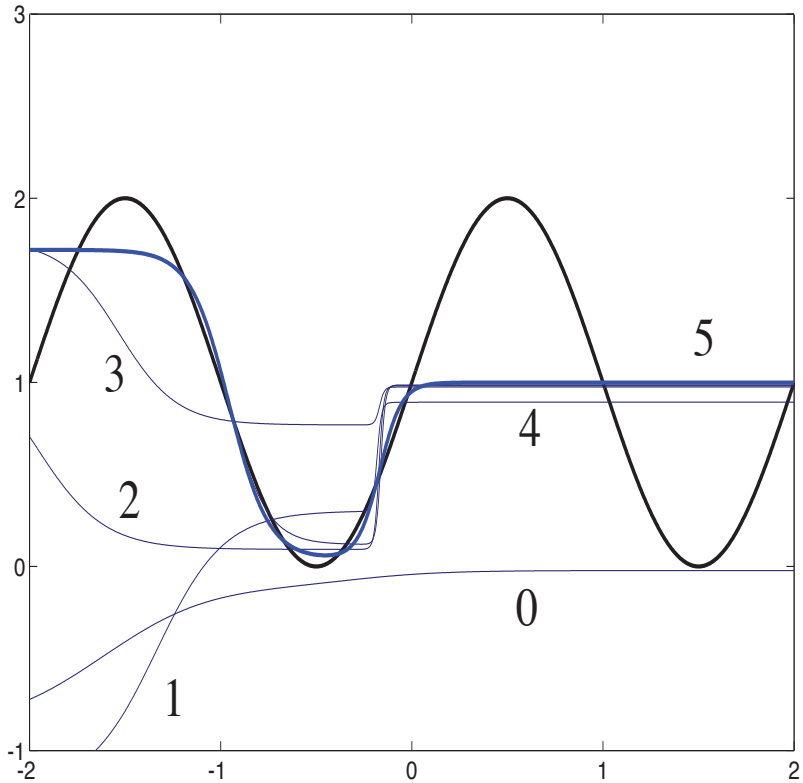




Convergence

$$g(p) = 1 + \sin(\pi p)$$

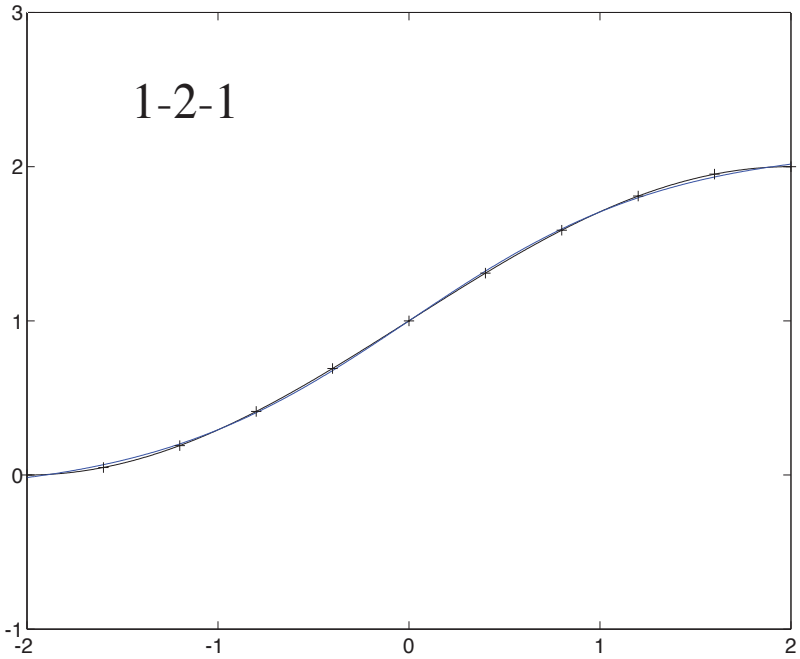


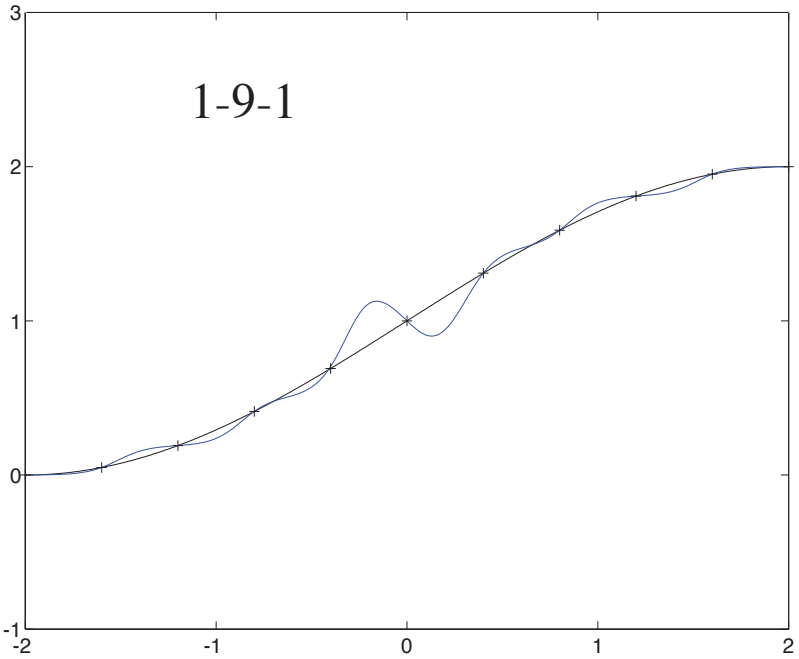


Generalization

$$\{\mathbf{p}_1,\mathbf{t}_1\},\{\mathbf{p}_2,\mathbf{t}_2\},\ \ldots\ ,\{\mathbf{p}_Q,\mathbf{t}_Q\}$$

$$g(p) = 1 + \sin\left(\frac{\pi}{4}p\right) \qquad p = -2, -1.6, -1.2, \ldots, 1.6, 2$$





- A cat that once sat on a hot stove will never again sit on a hot stove or on a cold one either.

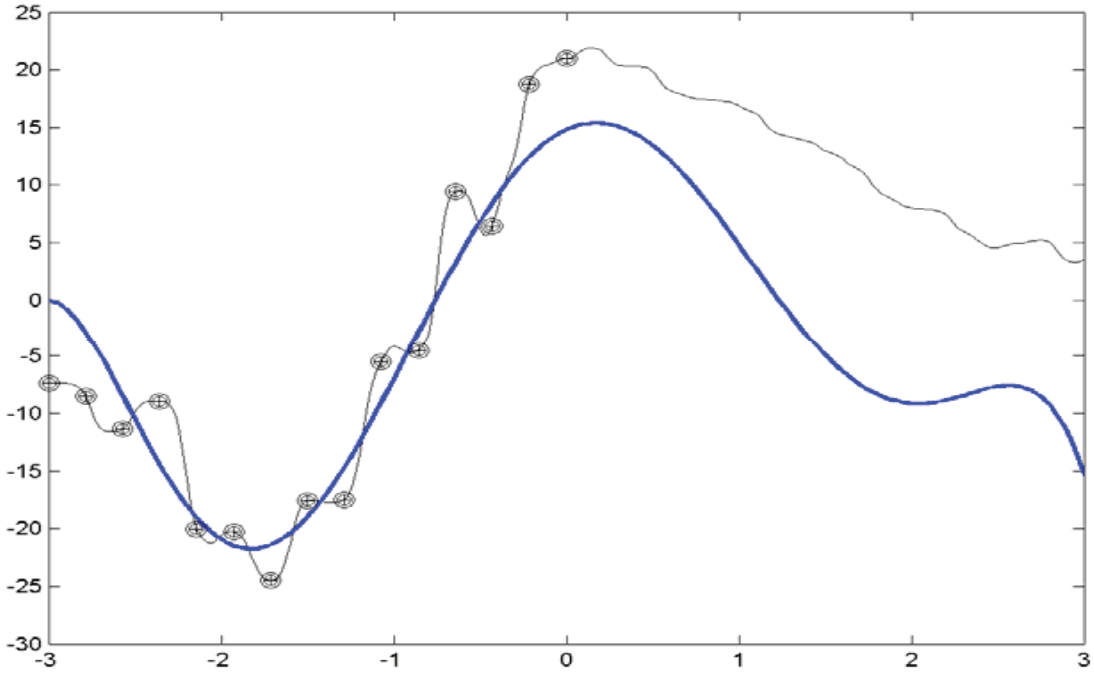
Mark Twain

Cause of Overfitting

- The network input-output mapping is accurate for the training data and for test data never seen before.
- The network interpolates well.
- Poor generalization is caused by using a network that is too complex (too many neurons/parameters). To have the best performance we need to find the least complex network that can represent the data (Ockham's Razor).
- Find the simplest model that explains the data.

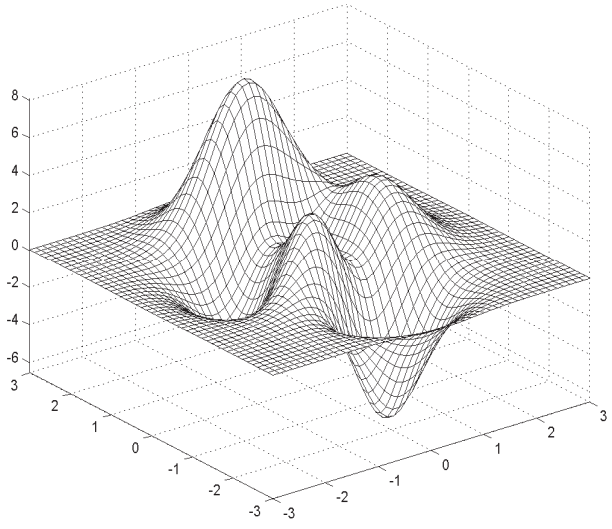
Good Generalization

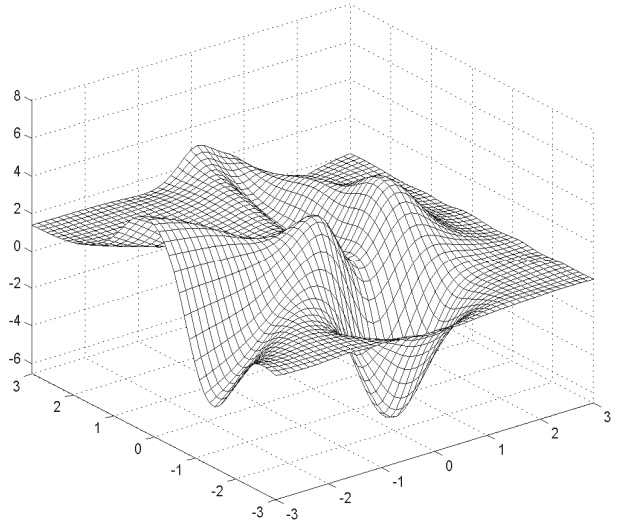
Overfitting Extrapolation



Interpolation

Extrapolation in 3-D





Measuring Generalization

- Part of the available data is set aside during the training process.
- After training, the network error on the test set is used as a measure of generalization ability.
- The test set must never be used in any way to train the network, or even to select one network from a group of candidate networks.
- The test set must be representative of all situations for which the network will be used.

Methods for Improving Generalization

- Pruning (removing neurons) until the performance is degraded.
- Growing (adding neurons) until the performance is adequate.
- Validation Methods
- Regularization

Early Stopping

- Break up data into training, validation, and test sets.
- Use only the training set to compute gradients and determine weight updates.
- Compute the performance on the validation set at each iteration of training.
- Stop training when the performance on the validation set goes up for a specified number of iterations.
- Use the weights which achieved the lowest error on the validation set.

Early Stopping Example