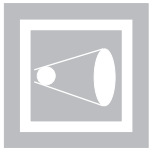
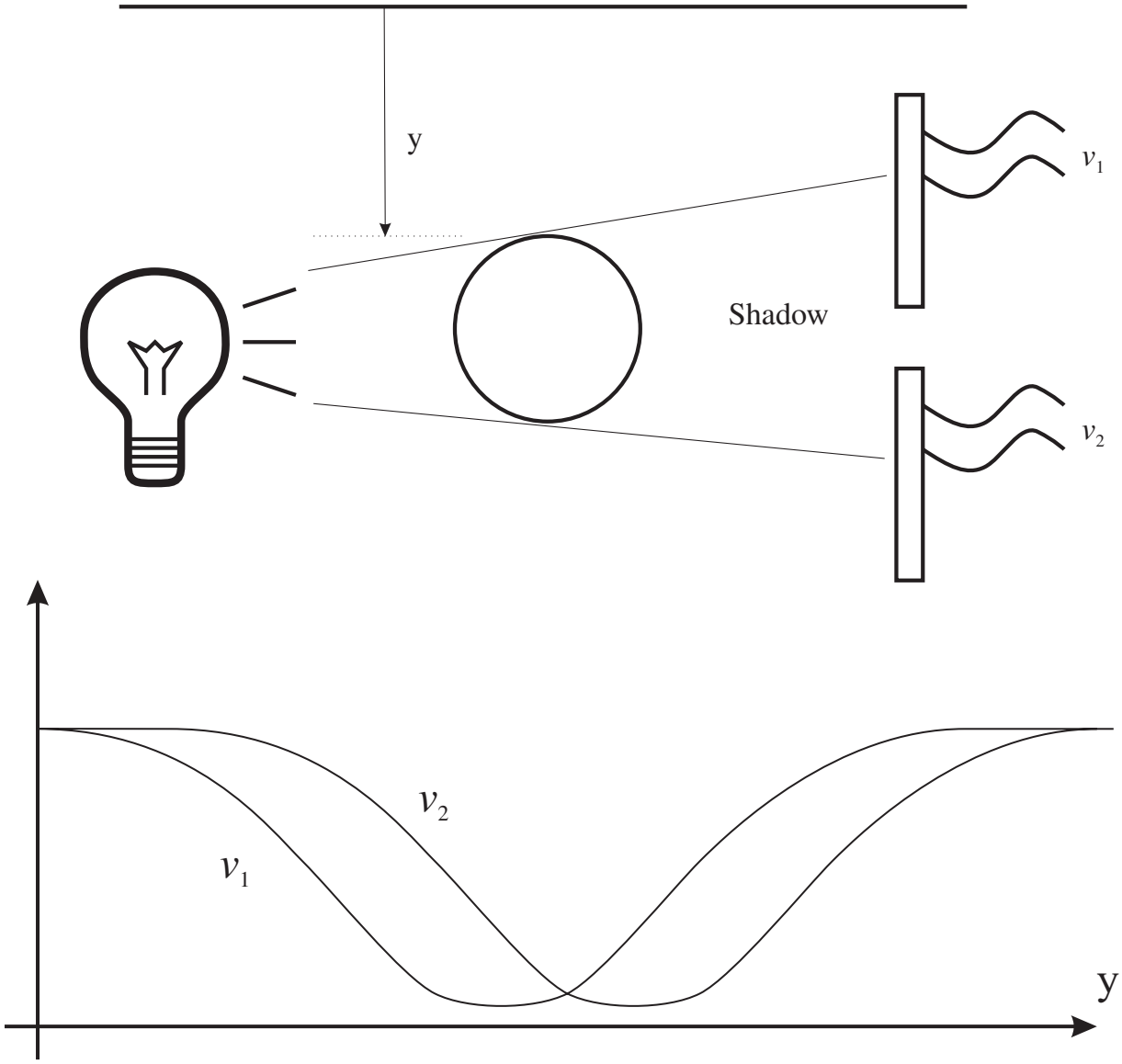


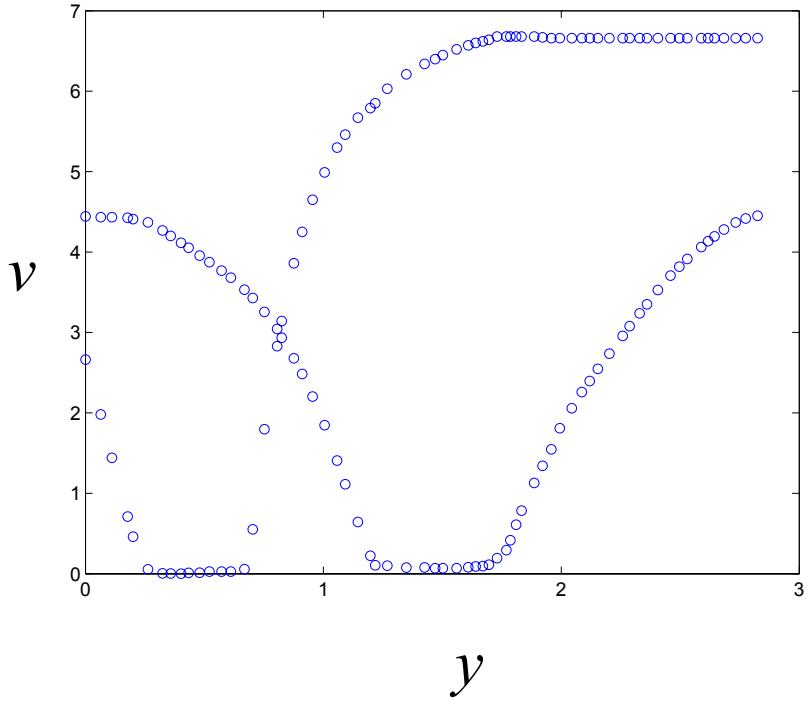
Function Approximation Case Study: Smart Sensor

Smart Sensor Diagram

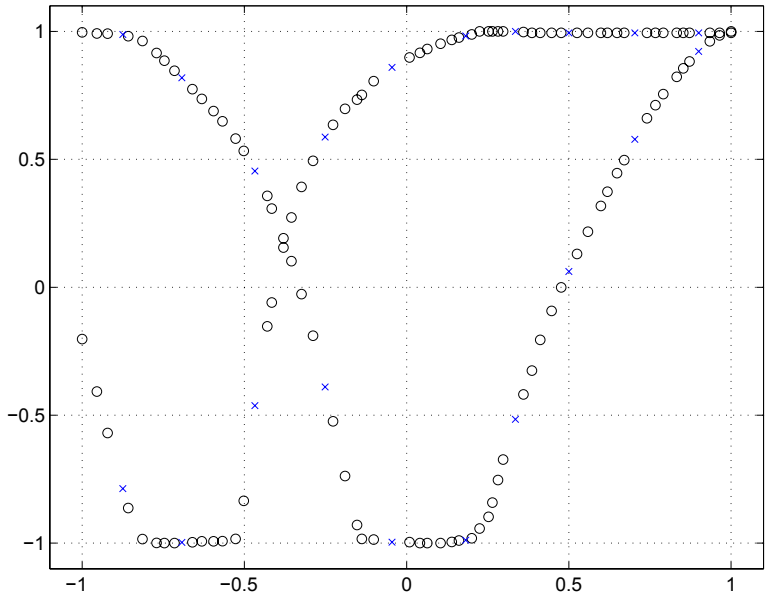


Collected Data

Raw Data

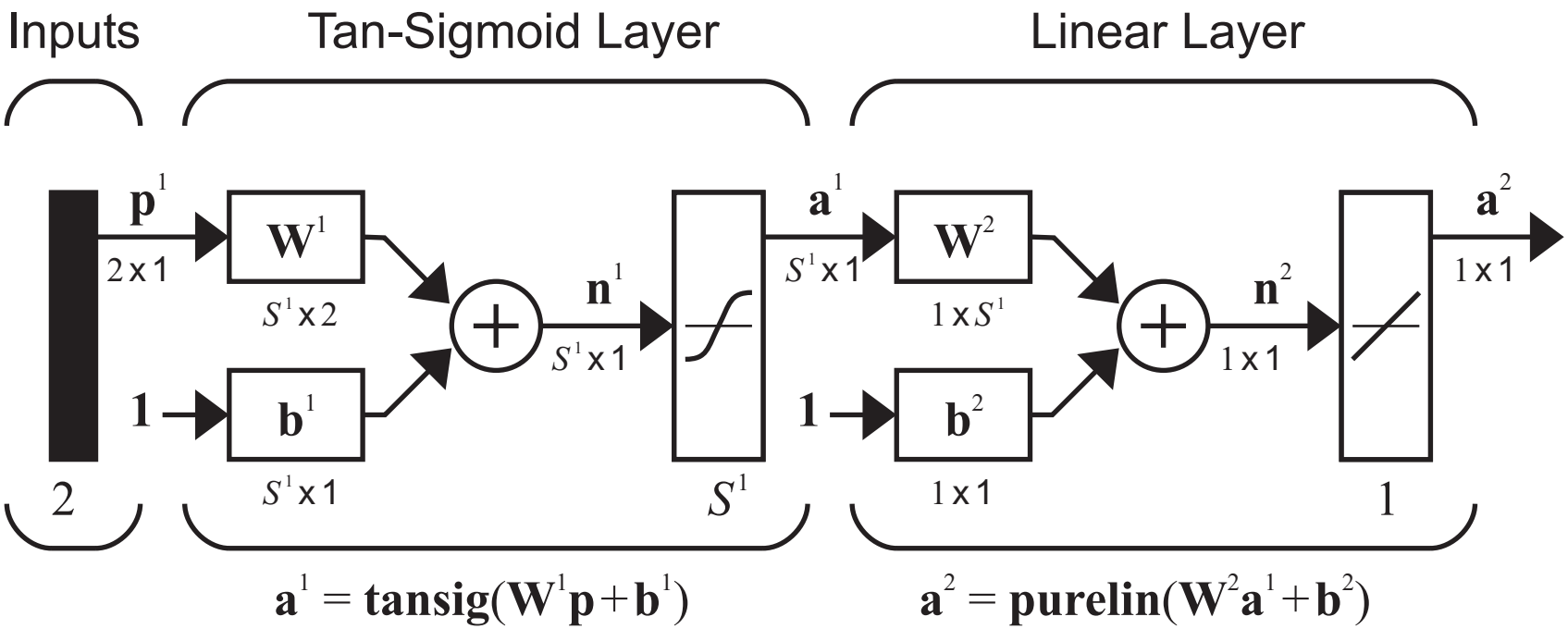


Normalized Data

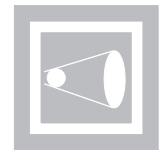


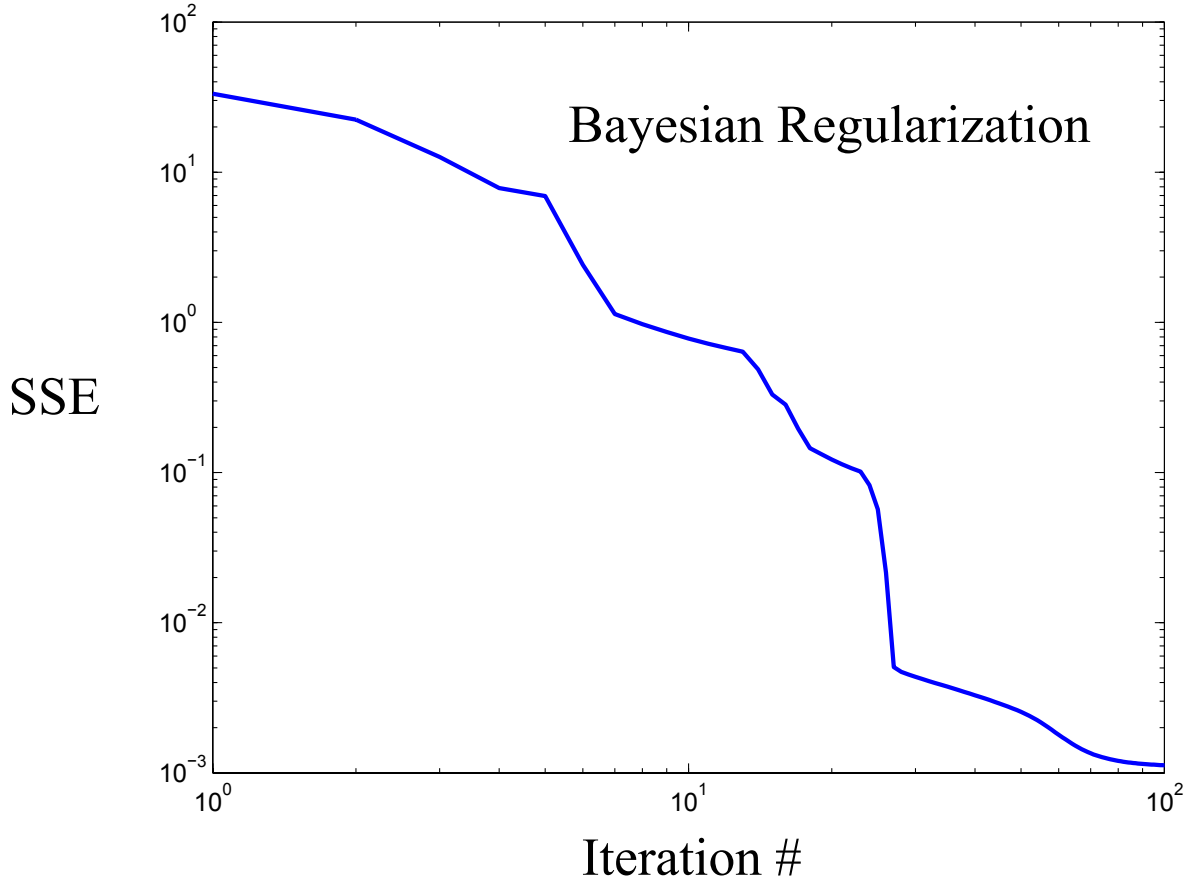
$$\mathbf{p} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \qquad t = y$$

Network Architecture



Training Performance (S^1=10)



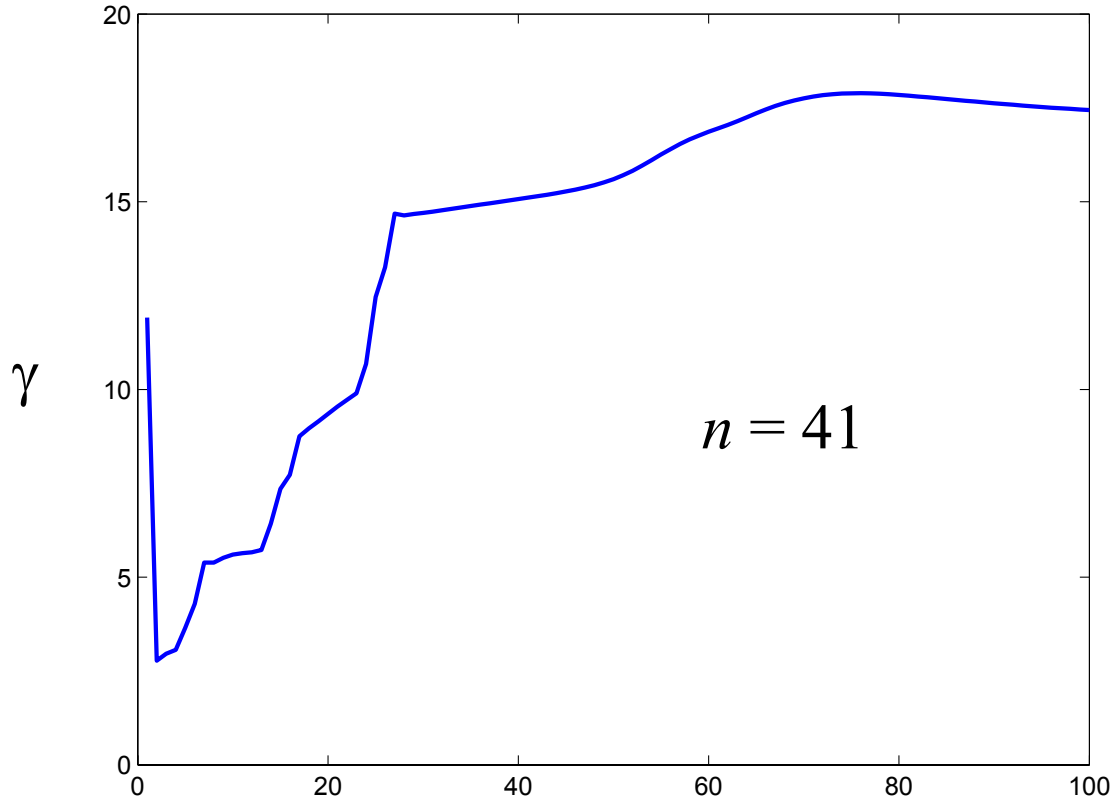


Final performance using five random initial weights.

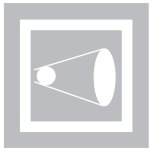
1.121e-003	8.313e-004	1.068e-003	8.672e-004	8.271e-004

23

Effective Number of Parameters



23 Number of Hidden Layer Neurons



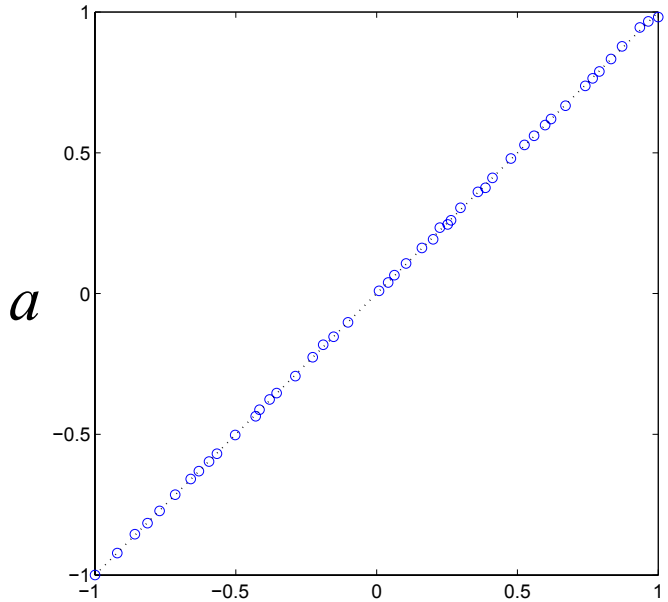
Sum Squared Error

$S^1 = 3$	$S^1 = 5$	$S^1 = 8$	$S^1 = 10$	$S^1 = 20$
4.406e-003	9.227e-004	8.088e-004	8.672e-004	8.096e-004

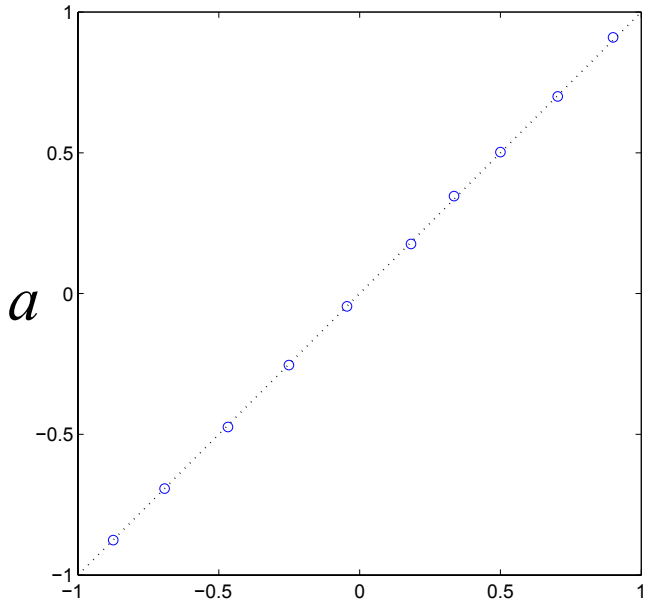
After a sufficient number of neurons is reached (~5) the error does not go down, if Bayesian regularization is used.

Scatter Plots

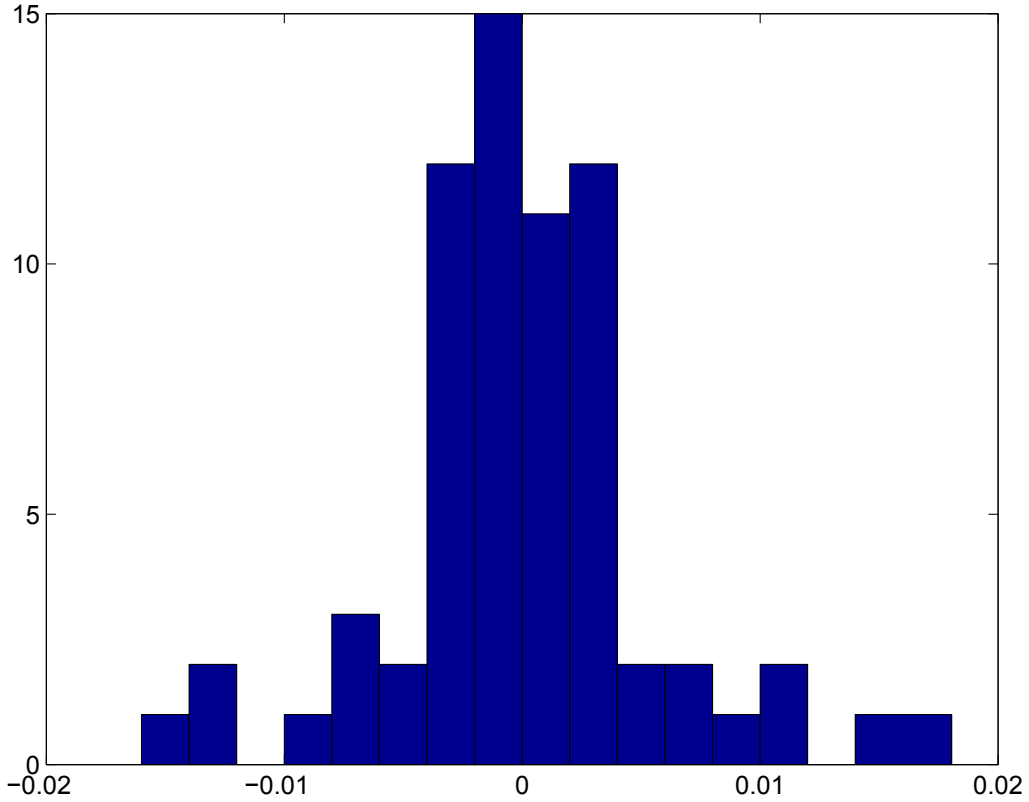
Training



Testing



Error Histogram



$$\mathbf{a} = (\mathbf{a}^n + 1).*\frac{(\mathbf{t}^{max} - \mathbf{t}^{min})}{2} + \mathbf{t}^{min}$$

Trained Network Response