EENG582: Artificial Neural Networks

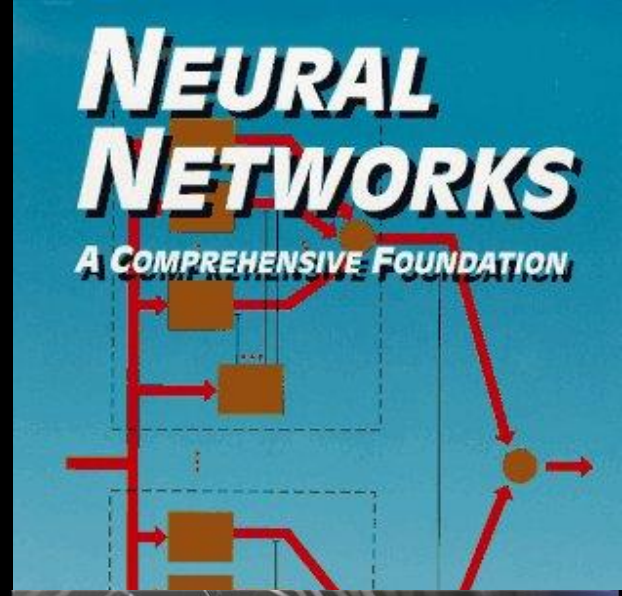
Neural Networks A Comprehensive Foundation by S. Haykin

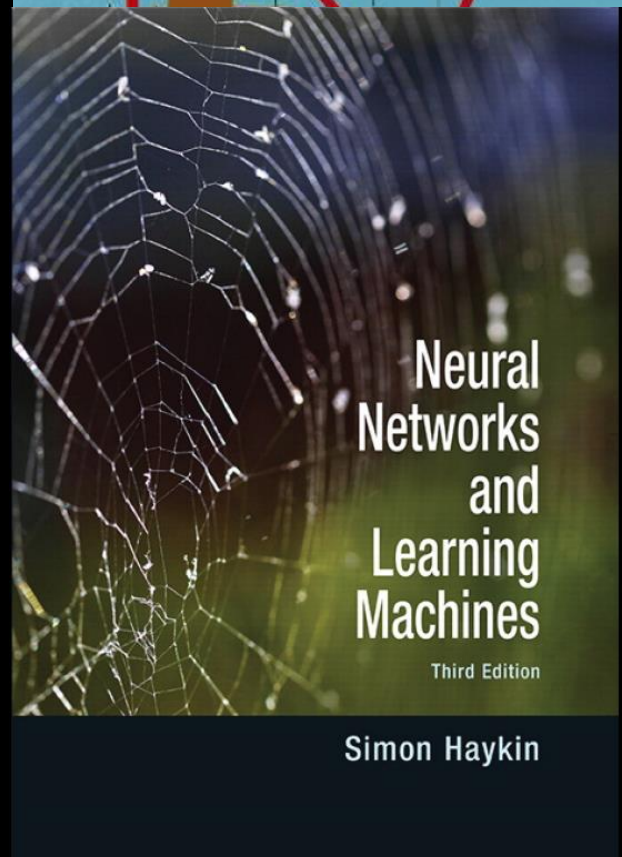
Problems with Solutions

4 Multilayer Perceptrons

Prof. Dr. Hasan AMCA
Electrical and Electronic Engineering Department
(ee.emu.edu.tr)

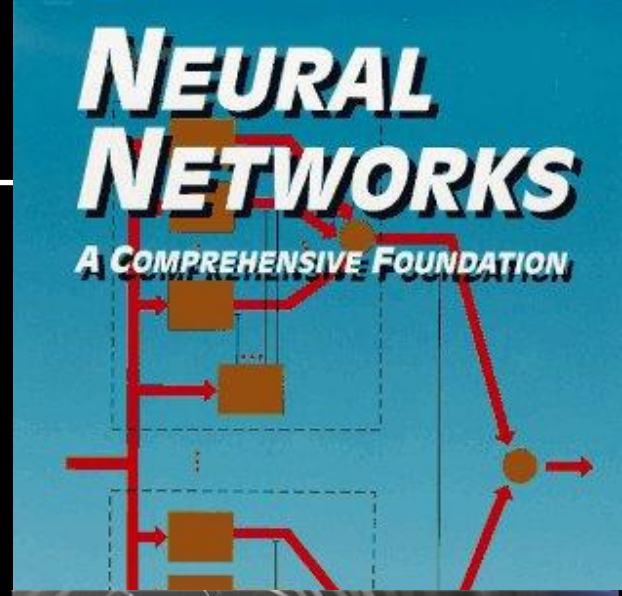
Eastern Mediterranean University
(emu.edu.tr)

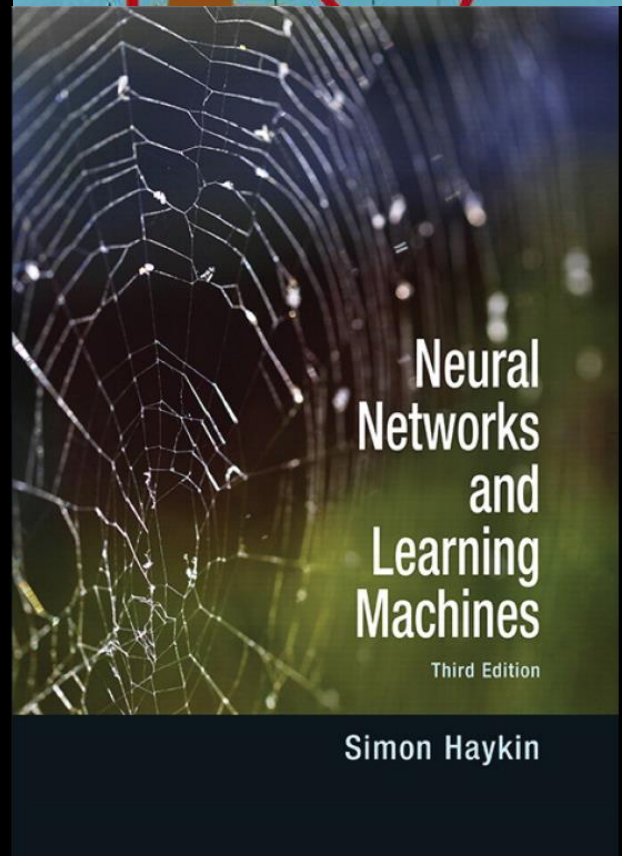




3 Single Layer Perceptrons

4.1 Introduction 178
4.2 Some Preliminaries 181
4.3 Back-Propagation AJgorithm . 183
4.4 Summary of the Back-Propagation Algorithm 195
4.5 XOR Problem 197
4.6 Heuristics for Making the Back -Propagation Algorithm Perfol'm Better 200
4. 7 Output Representation and Decision Rule 206
4.8 Computer Experiment 209
4.9 Feature Detection 221
4.10 Back-Propagation and Differentiation 224
4.11 Hessian Matrix 226
4.12 Generalization 227
4.13 Approximations of Functions 230
4.14 Cross-Validation 235
4.15 Network Pruning Techniques 240
4.16 Virtues and Limitations of Back-Propagation Learning 248
4.17 Accelerated Convergence of Back-Propagation Learning 255
4.18 Supervised Learning Viewed as an Optimization Problem 256
4.19 Convolutional Networks 267
4.20 Summary and Discussion 269
Notes and References 270
Problems 274





Example 1: Bayesian Decision Boundary

- Consider a communication system where the transmitter transmits messages $m = 0$ or $m = 1$, occurring with a priori probabilities of ¾ and ¼ respectively. The message is contaminated by a noise n, which is independent from m and takes on the values -1, 0, 1 with probabilities 1/8, 5/8 and 2/8 respectively. The received signal, or the observation, can be represented as $r = m + n$. From r, we wish to infer what the transmitted message m was (estimated state), denoted using $\hat{m}$, which also takes values on 0 or 1. When $m = \hat{m}$, the detector correctly receives the original message, otherwise an error occurs.

  a) Find the decision rule that achieves the maximum probability of correct decision. Compute the probability of error for this decision rule.

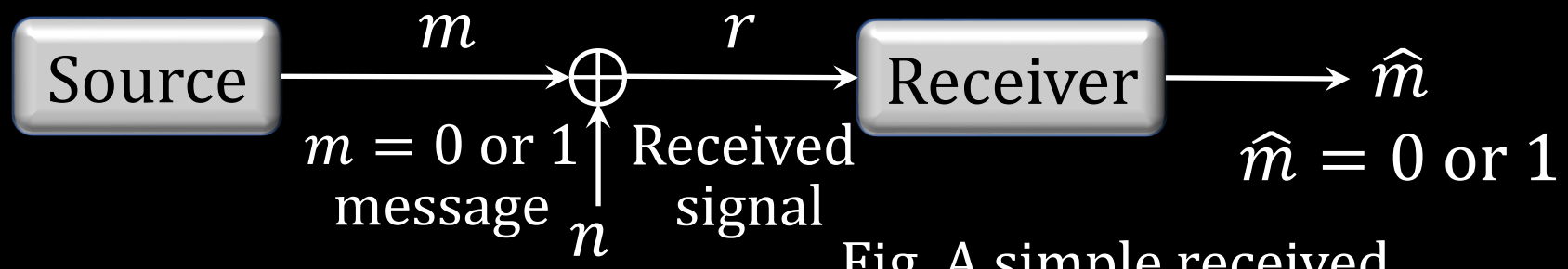


Fig. A simple received

Example 1: Bayesian Decision Boundary

a) Find the decision rule that achieves the maximum probability of correct decision. Compute the probability of error for this decision rule.

Solution: It is equivalent to find the decision rule that achieves the minimum probability of error. The receiver decides the transmitted message is 1, i.e., $\hat{m} = 1$ if

$$P(r|m=1).P_r(m=1) \geq P(r|m=0).P_r(m=0)$$

$$P(r|m=1) \geq P(r|m=0).3 \quad (1)$$

Otherwise, the receiver decides $\hat{m} = 0$. The likelihood functions for these two cases are

$$P(r|m=1) = \begin{cases} 1/8 & \text{if } r=0 \\ 5/8 & \text{if } r=1 \\ 2/8 & \text{if } r=2 \\ 0 & \text{otherwise} \end{cases} \qquad P(r|m=0) = \begin{cases} 1/8 & \text{if } r=-1 \\ 5/8 & \text{if } r=0 \\ 2/8 & \text{if } r=1 \\ 0 & \text{otherwise} \end{cases}$$

Example 1: Bayesian Decision Boundary

- Eq. 1 holds only when r = 2. Therefore, the decision rule can be summarized as

$$\hat{m} = \begin{cases} 1 & \text{if } r = 0 \\ 0 & \text{otherwise} \end{cases}$$

- The probability of error is as follows:

$$\begin{aligned} P_r(e) &= P_r(\hat{m} = 1|m = 0).P_r(m = 0) + P_r(\hat{m} = 0|m = 1).P_r(m = 1) \\ &= P_r(r = 2|m = 0).P_r(m = 0) + P_r(r \neq 2|m = 1).P_r(m = 1) \\ &= 0 + \left(\frac{1}{8} + \frac{5}{8}\right).\frac{1}{4} = \frac{3}{6} = 0.1875 \end{aligned} \quad (1)$$

Example 1: Bayesian Decision Boundary

b) Let's have the noise n be a continuous random variable, uniformly distributed between -3/4 and 2/4, and still statistically independent of m. First, plot the pdf of n. Then, find a decision rule that achieves the minimum probability of error and compute the probability of error.

Solution:

- The uniform distribution is plotted in Fig. 3.
- The decision rule is still determined by using (1)
- The likelihood functions become continuous, instead of discrete as in (a):

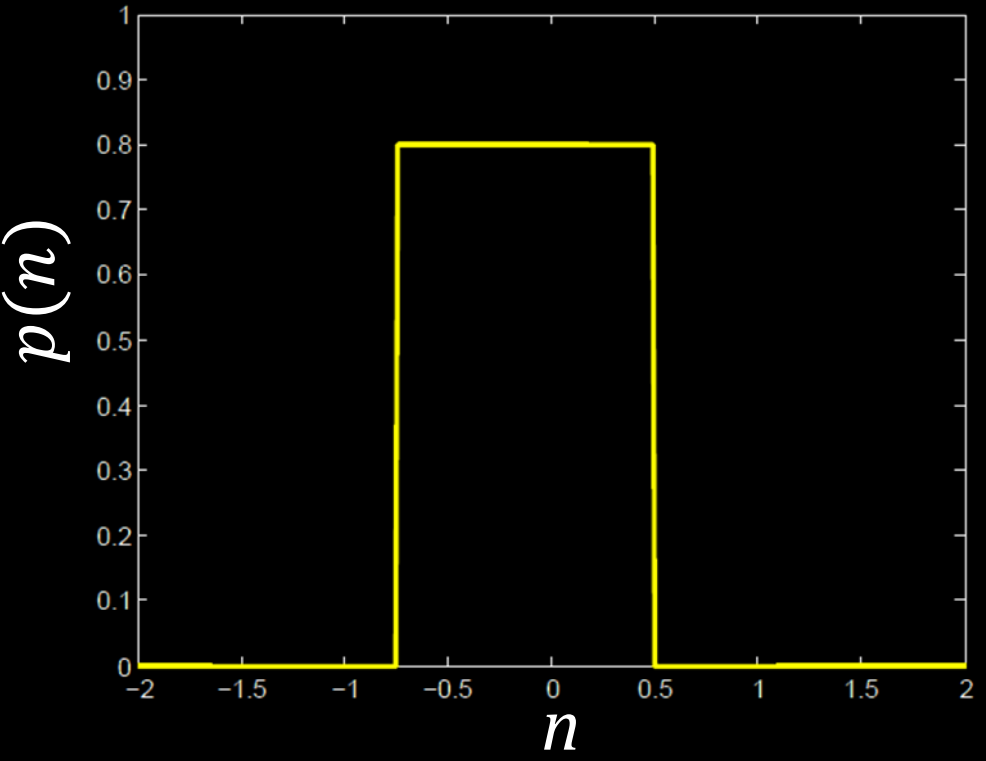


Fig. 3. The pdf of n.

Example 1: Bayesian Decision Boundary

$$P(r|m=1) = \begin{cases} \frac{4}{5} & if \ \frac{1}{4} < r \leq \frac{6}{4} \\ 0 & \text{otherwise} \end{cases}$$

$$P(r|m=0) = \begin{cases} \frac{4}{5} & if \ \frac{-3}{4} < r \leq \frac{2}{4} \\ 0 & \text{otherwise} \end{cases}$$

- The interesting region is where the two pdf's overlap with each other,
- namely when $1/4 < r \leq 2/4$.
- From (1), we know we should decide $\hat{m} = 0$ for this range.

Example 1: Bayesian Decision Boundary

- The decision rule can be summarized as

$$\hat{m} = \begin{cases} 0 & \text{if } -\frac{3}{4} < r \leq \frac{1}{4} \\ 0 & \text{if } +\frac{1}{4} < r \leq \frac{1}{4} \\ 1 & \text{if } \frac{1}{4} < r \leq \frac{2}{4} \end{cases}$$

- Note that at the decision boundaries, there is ambiguity on which decision we should make. Again, either decision won't change the probability of error, so it is acceptable to decide both ways.
- The probability of error is then,

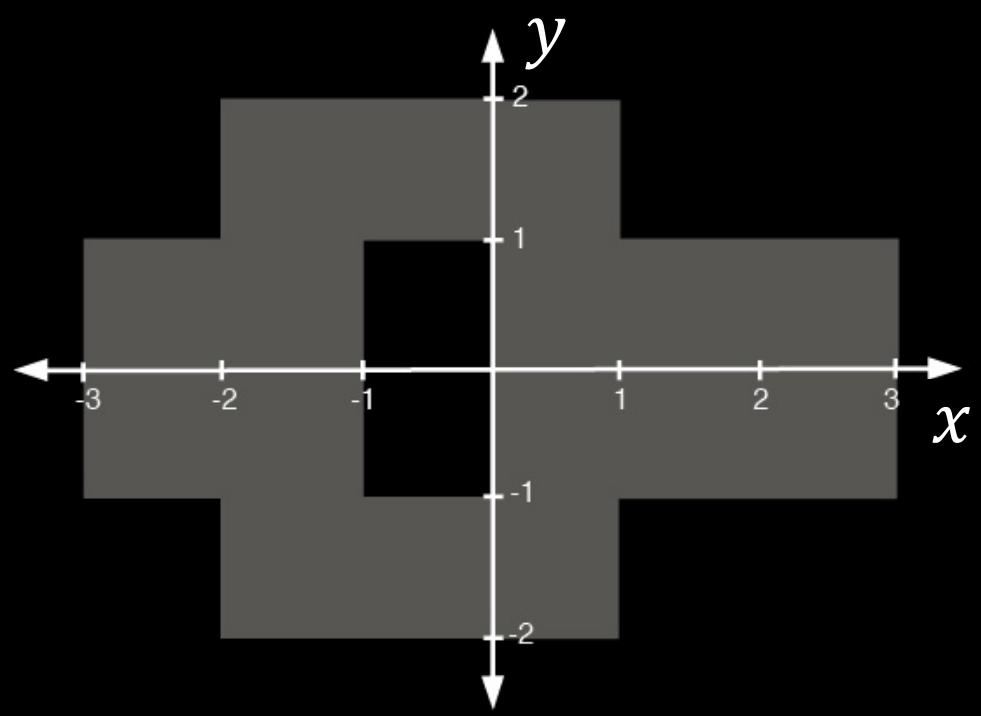
$$P_r(e) = P_r(\hat{m} = 1|m = 0).P_r(m = 0) + P_r(\hat{m} = 0|m = 1).P_r(m = 1)$$

$$P_r\left(\frac{1}{4} < r \leq \frac{2}{4}\middle|m = 1\right).P_r(m = 1) = \left(\frac{2}{4} - \frac{1}{4}\right).\frac{4}{5}.\frac{1}{4} = \frac{1}{20}$$

Example 2: Bayesian Decision Boundary

- Suppose x and y are random variables. Their joint density, depicted below, is constant in the shaded area and 0 elsewhere,
  a) Let w_1 be the case when $x \leq 0$, and w_2 be the case when $x > 0$. Determine the *a priori* probabilities of the two classes $P(w_1)$ and $P(w_1)$. Let y be the observation from which we infer whether w_1 or w_2 happens. Find the likelihood functions, namely, the two conditional distributions $p(y)|w_1)$ and $p(y)|w_2)$.

Fig. 5: The joint distribution of x and y.



Example 2: Bayesian Decision Boundary

Solution a)

- By simply counting the number of unit squares in the shaded areas on the left and right sides of the line $x = 0$; we can directly find out that there are 8 unit squares on each side. Thus, the two cases are equally likely, i.e. $\mathrm{P}(w_1) = \mathrm{P}(w_2) = 0.5$.
- It's straight-forward to obtain likelihood functions $p(y|w_1)$ and $p(y|w_2)$ by counting the number of unit squares for different ranges of y. We need to be careful with normalizing the integral of the distribution 1. See Figure 6.

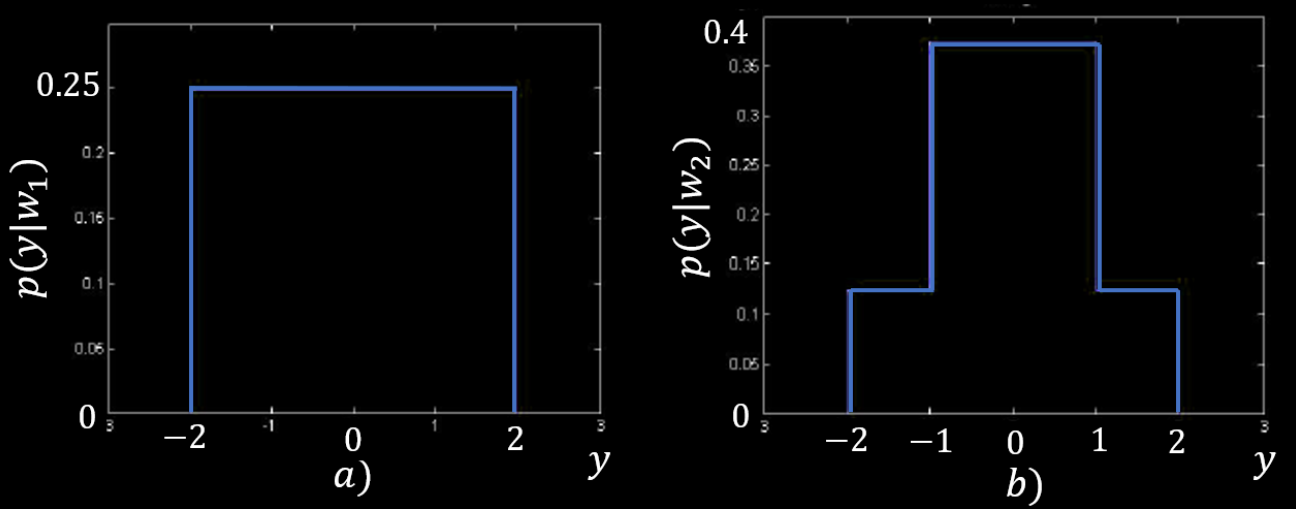


Fig. 6: Likelihood functions: (a) $p(y|w_1)$ left, (b) $p(y|w_2)$ right.

Example 2: Bayesian Decision Boundary

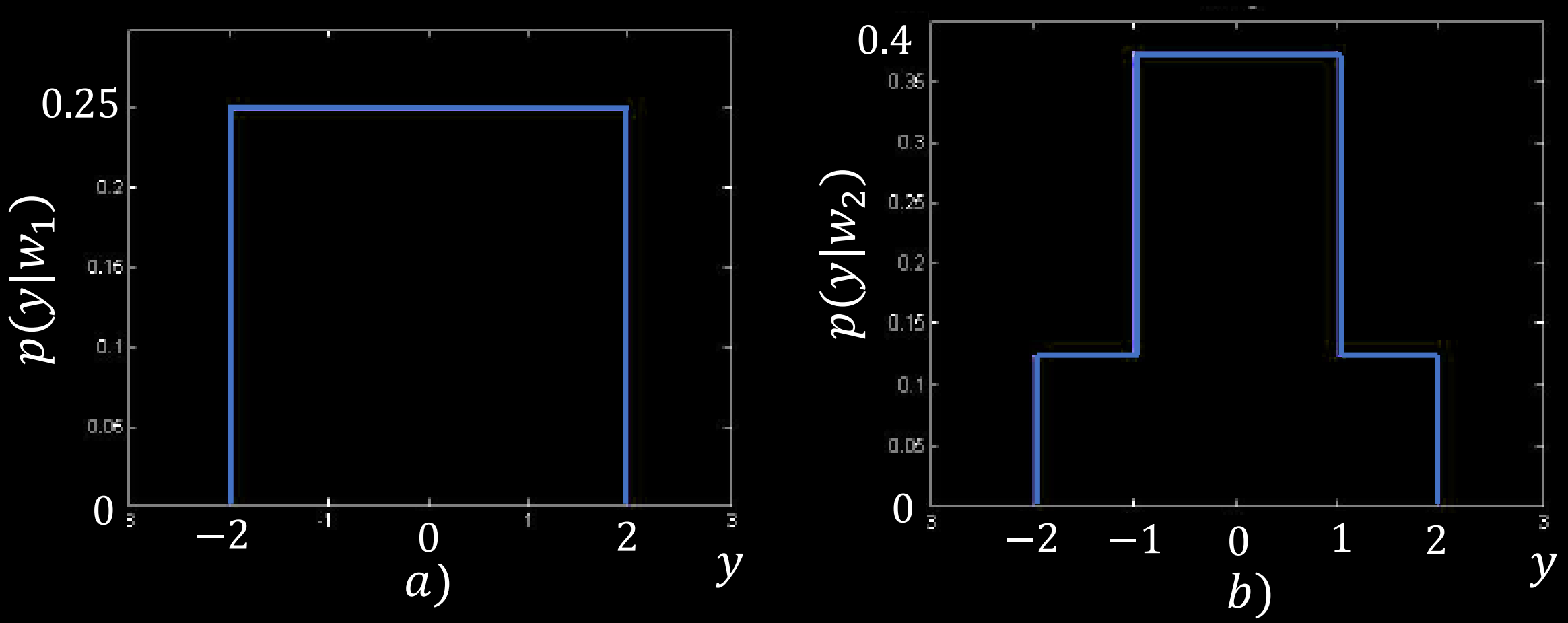


Fig. 6: Likelihood functions: (a) $p(y|w_1)$ left, (b) $p(y|w_2)$ right.

Example 2: Bayesian Decision Boundary

b) Find the decision rule that minimizes the probability of error and calculate what the probability of error is. Please note that there will be ambiguities at decision boundaries, but how you classify when y falls on the decision boundary doesn't affect the probability of error.

Solution b)

- As shown in (a), the *a priori* probabilities of w_1 and w_2 are identical. The minimum probability of error decision rule simply relies on the comparison of the two likelihood functions. In other words, it becomes a ML decision rule.
- The decision rule can be summarized as:

$$\widehat{w} = \begin{cases} w_1 & \text{if } -2 < y \leq -1 \quad \text{or} \quad 1 < y \leq 2 \\ w_2 & \text{if } -1 < y \leq 1 \end{cases}$$

Example 2: Bayesian Decision Boundary

- The probability of error is thus:

$$P_r(e) = P_r(\hat{w} = w_1|w_2)P_r(w_1) + P_r(w_1|w_2)P_r(w_2)$$
$$= \frac{1}{2}P_r(-1 < y \leq 1|w_1) + \frac{1}{2}P_r(-2 < y \leq -1, 1 < y \leq 2|w_2)$$
$$= \frac{1}{2} * \frac{1}{2} + \frac{1}{2} * \frac{1}{4} = \frac{3}{8}$$

Bayesian Decision Boundary

- Assuming that for a two-class problem, (1) the classes $\mathcal{C}_1$ and $\mathcal{C}_2$ are equiprobable, (2) the costs for correct classifications are zero and (3) the costs for misclassifications are equal.
- we find that the optimum decision boundary is found by applying the likelihood ratio test:

$$\Lambda(\mathbf{x}) \underset{\mathcal{C}_2}{\overset{\mathcal{C}_1}{\lessgtr}} \xi \tag{4.58}$$

where $\Lambda(\mathbf{x})$ is the likelihood ratio, defined by

$$\Lambda(\mathbf{x}) = \frac{f_{\mathbf{x}}(\mathbf{x}|\mathcal{C}_1)}{f_{\mathbf{x}}(\mathbf{x}|\mathcal{C}_2)} \tag{4.59}$$

and ξ is the threshold of the test, defined by

$$\xi = p_2/p_1 = 1 \tag{4.60}$$

Bayesian Decision Boundary

- For the example being considered, we have

$$\Lambda(\mathbf{x}) = \frac{\sigma_2^2}{\sigma_1^2} \exp\left(-\frac{1}{2\sigma_1^2} \|\mathbf{x} - \boldsymbol{\mu}_1\|^2 + \frac{1}{2\sigma_2^2} \|\mathbf{x} - \boldsymbol{\mu}_2\|^2 \right)$$

- The optimum (Bayesian) decision boundary is therefore defined by

$$\frac{\sigma_2^2}{\sigma_1^2} \exp\left(-\frac{1}{2\sigma_1^2} \|\mathbf{x} - \boldsymbol{\mu}_1\|^2 + \frac{1}{2\sigma_2^2} \|\mathbf{x} - \boldsymbol{\mu}_2\|^2 \right) = 1$$

or equivalently,

$$\frac{1}{\sigma_2^2} \|\mathbf{x} - \boldsymbol{\mu}_2\|^2 - \frac{1}{\sigma_1^2} \|\mathbf{x} - \boldsymbol{\mu}_1\|^2 = 4 \log\left(\frac{\sigma_1}{\sigma_2}\right) \qquad (4.61)$$

Bayesian Decision Boundary

- We may redefine the optimum decision boundary of (4.61) as

$$\|\mathbf{x} - \mathbf{x}_c\|^2 = \mathbf{r}^2 \tag{4.62}$$

- where,

$$\mathbf{x}_c = \frac{\sigma_2^2 \boldsymbol{\mu}_1 - \sigma_1^2 \boldsymbol{\mu}_2}{\sigma_2^2 - \sigma_1^2} \tag{4.63}$$

and

$$\mathbf{r}^2 = \frac{\sigma_1^2 \sigma_2^2}{\sigma_2^2 - \sigma_1^2} \left[\frac{\|\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2\|^2}{\sigma_2^2 - \sigma_1^2} + 4 \log\left(\frac{\sigma_2}{\sigma_1}\right) \right] \tag{4.64}$$

- (4.62) represents a circle with center $\mathbf{x}_c$ and radius r. Let Ω_1 define the region lying inside this circle. The Bayesian classification rule for the problem at hand is stated as follows:

Classify the observation vector $\mathbf{x}$ *as belonging to class* $\mathcal{C}_1$ *if the likelihood ratio* $\Lambda(\mathbf{x})$ *is greater than the threshold* ξ *and to class* $\mathcal{C}_1$ *otherwise.*

Bayesian Decision Boundary

- Now, we have a circular decision boundary whose center is located at

$$\mathbf{x}_c = \begin{bmatrix} -\frac{2}{3} \\ 0 \end{bmatrix} \text{ and whose radius is } r \cong 2.34.$$

- Let c denote the set of correct classification outcomes, and and e the set of erroneous classification outcomes.
- The *conditional probability of error* P_e (misclassification), given that the classifier input vector was drawn from the distribution of class $\mathcal{C}_1$, according to the Bayesian decision rule is,

$$P_e = p_1 P(e|\mathcal{C}_1) + p_2 P(e|\mathcal{C}_2) \qquad (4.65)$$

- For our problem, $P(e|\mathcal{C}_1) \approx 0.1056,\ P(e|\mathcal{C}_2) \approx 0.1849$, with $p_1 = p_2 = 1/2$.
- Equivalently, the probability of correct classification is $P_c = 1 - P_e \approx 0.8159$.

Experimental Determination of Optimal Multilayer Perceptron

- Table 4.1 lists the variable parameters of a multilayer perceptron (MLP) that involves a single layer of hidden neurons, and that is trained with the back-propagation algorithm operating in the sequential mode.

- Since the ultimate objective of a pattern classifier is to achieve an acceptable rate of correct classification, this criterion is used to judge when the variable parameters of the MLP (used as a pattern classifier) are optimal.

TABLE 4.1 Variable Parameters of Multilayer Perceptron

Parameter	Symbol	Typical Range
Number of hidden neurons	m_1	$(2,\infty)$
Learning-rate parameter	μ	$(0,1)$
Momentum constant	α	$(0,1)$

Optimal Number of Hidden Neurons

- Reflecting practical approaches to the problem of determining the optimal number of hidden neurons, m_1, the criterion used is the smallest number of hidden neurons that yields a performance "close" to the Bayesian classifier usually within 1 percent.
- Thus, the experimental study begins with two hidden neurons as the starting point for the simulation results summarized in Table 4.2.

TABLE 4.2 Simulation Results for Two Hidden Neurons

Run Number	Training Set Size	Number of Epochs	Mean-Square Error	Probability of Correct Classification, P_c
1	500	320	0.2375	80.36%
2	2000	80	0.2341	80.33%
3	8000	20	0.2244	80.47%

Optimal Number of Hidden Neurons

- After convergence of a network trained with a total number of N patterns, the probability of correct classification can in theory be calculated as follows:

$$P(c, N) = p_1 P(c, N|\mathcal{C}_1) + p_2 P(c, N|\mathcal{C}_2) \quad (4.66)$$

where $p_1 = p_2 = 1/2$, and

$$P(c, N|\mathcal{C}_1) = \int_{\Omega_1(N)} f_{\mathbf{x}}(\mathbf{x}|\, \mathcal{C}_1) d\mathbf{x} \quad (4.67)$$

$$P(c, N|\mathcal{C}_2) = \int_{\Omega_1(N)} f_{\mathbf{x}}(\mathbf{x}|\, \mathcal{C}_2) d\mathbf{x} \quad (4.68)$$

- and $\Omega_1(N)$ is the region in decision space over which the multilayer perceptrons (trained with N patterns) classifies the vector $\mathbf{x}$ (representing a realization of the random vector $\mathbf{X}$) as belonging to class $\mathcal{C}_1$.

Optimal Number of Hidden Neurons

- Let A be a random variable that counts the number of patterns out of the N test patterns that are classified correctly. Then the ratio

$$P_N = \frac{A}{N}$$

- is a random variable that provides the maximum-likelihood unbiased estimate of the actual classification performance p of the network.
- Assuming that p is constant over the N input-output pairs, we may apply the Chernoff bound to the estimator p_N of p, obtaining

$$P(|p_N - p| > \epsilon) < 2\exp(-2\epsilon^2 N) = \delta$$

- Application of the Chernoff bound yields $N \approx 26{,}500$ for $\epsilon = 0.01$ and $\delta = 0.01$ (i.e., 99 percent certainty that the estimate p has the given tolerance).
- The last column of Table 4.2 presents the probability of correct classification for a test size of $N = 32{,}000$.

Optimal Number of Hidden Neurons

- Table 4.3 presents the results of simulations repeated for the case of four hidden neurons, with all other parameters held constant.
- Although the mean square error is slightly lower than that in Table 4.2 for two hidden neurons, average rate of correct classification is slightly worse.
- The two hidden neurons choice is therefore preferred to four hidden neurons.

TABLE 4.3 Simulation Results for Four Hidden Neurons

Run Number	Training Set Size	Number of Epochs	Mean-Square Error	Probability of Correct Classification, P_c
1	500	320	0.2199	80.80%
2	2000	80	0.2108	80.81%
3	8000	20	0.2142	80.19%

Optimal Learning and Momentum Constants

- For the "optimal" values of the learning rate parameter η and momentum constant α, we may use any one of three definitions:

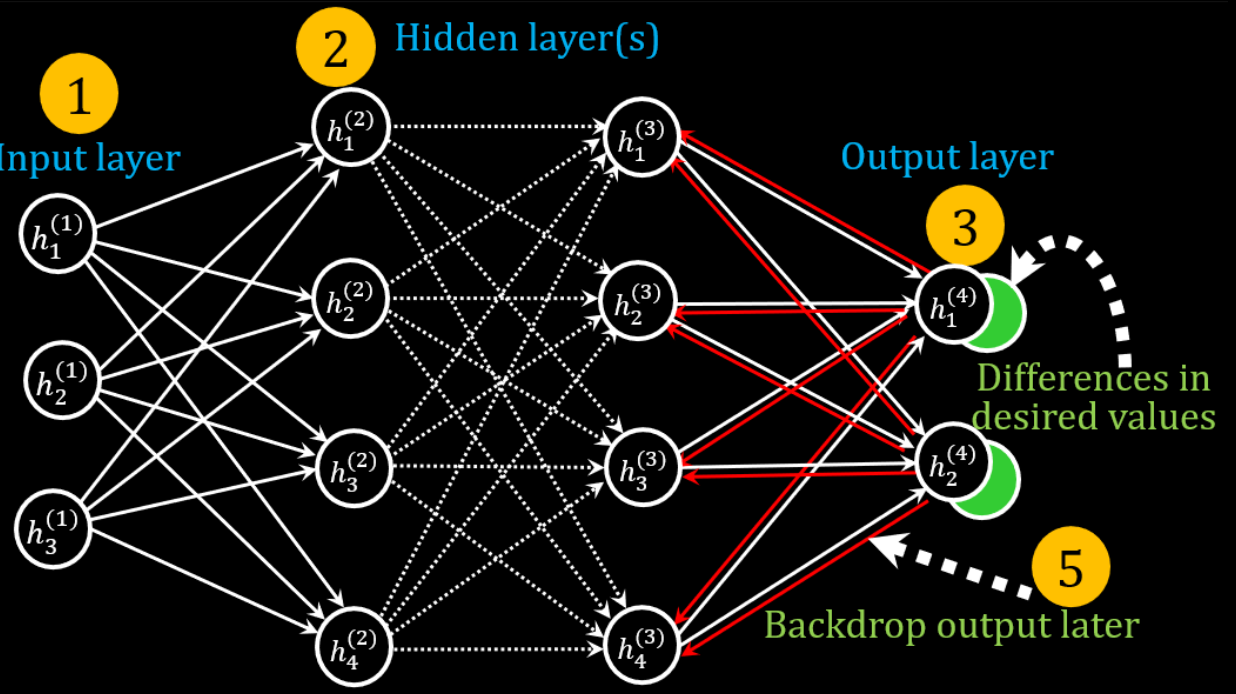
  1. The η and α that on average yield convergence to a local minimum in the error surface of the network with the least number of epochs.

  2. The η and α that, for either the worst-case or on average, yield convergence to the global minimum in the error surface with the least number of epochs.

  3. The η and α that on average yield convergence to the network configuration that has the best generalization over the entire input space, with the least number of epochs.

Optimal Learning and Momentum Constants

- Using a multilayer perceptron with two hidden neurons, combinations of learning rate parameter $\eta \in \{0.01, 0.1, 0.5, 0.9\}$ and momentum constant $\alpha \in \{0.01, 0.1, 0.5, 0.9\}$ are simulated to observe their effect on network convergence.
- See the Matlab code MLP_NN.m for the NN with two hidden layers and number of neurons in hidden layer, leaning rate η and momentum constant α are variables.



Evaluation of Optimal Network Design

- Given the "optimized" multilayer perceptrons having the parameters summarized in Table 4.4, the final network is evaluated to determine its decision boundary, ensemble-averaged learning curve, and probability of correct classification averaged over 20 independently trained networks.

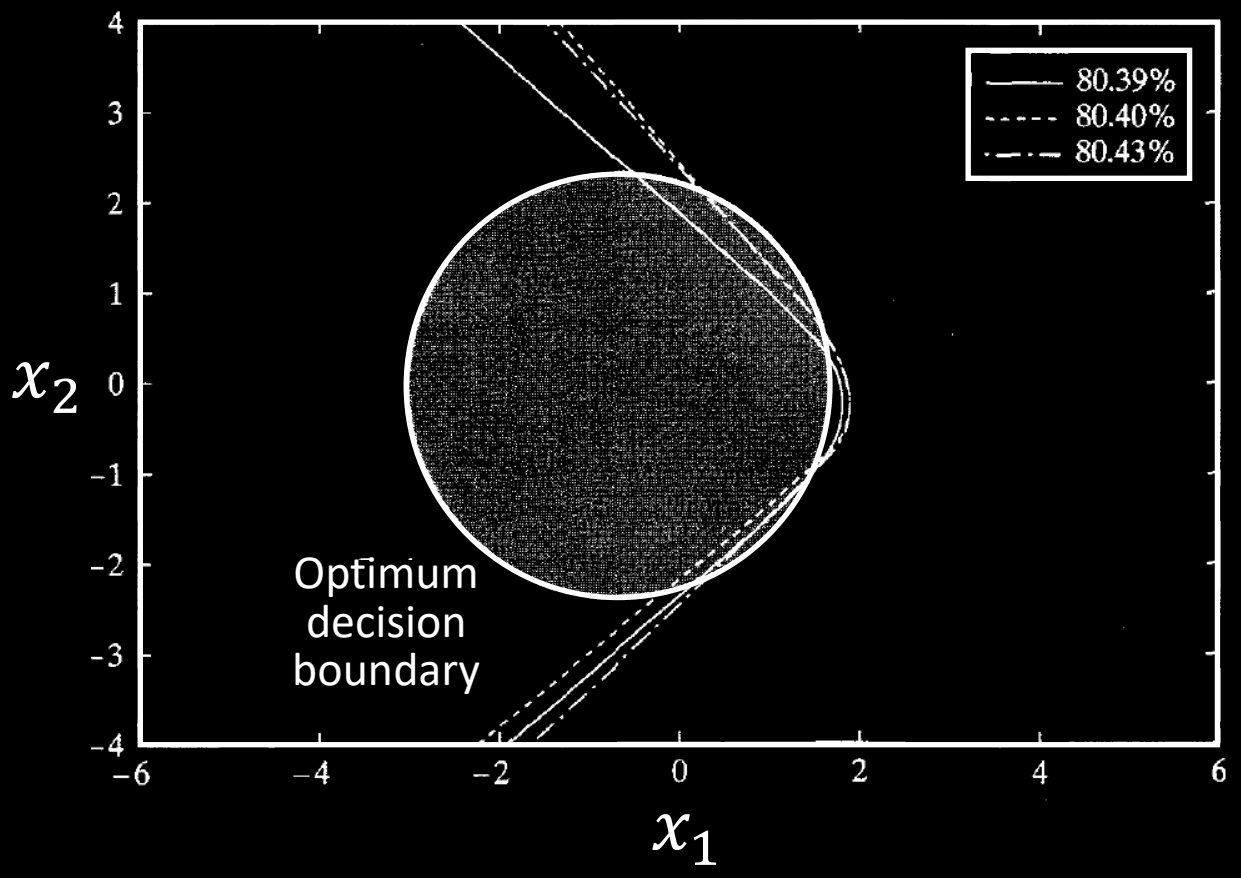
TABLE 4.4 Configuration of Optimized Multilayer Perceptron

Parameter	Symbol	Value
Optimum number of hidden neurons	m_{opt}	2.0
Optimum learning-rate parameter	η_{opt}	0.1
Optimum momentum constant	α_{opt}	0.5

Evaluation of Optimal Network Design

- Figure 4.17a shows three of the "best" decision boundaries for three networks in the ensemble of 20.
- Figure 4.17b shows three of the "worst" decision boundaries for three other networks in the same ensemble.
- The shaded (circular) Bayesian decision boundary is included in both figures for reference.
- The decision boundaries constructed by the back-propagation algorithm as shown in Fig. 4.17. are convex with respect to the region where they classify the observation vector $\mathbf{x}$ as belonging to class $\mathcal{C}_1$ or class $\mathcal{C}_2$.

Evaluation of Optimal Network Design



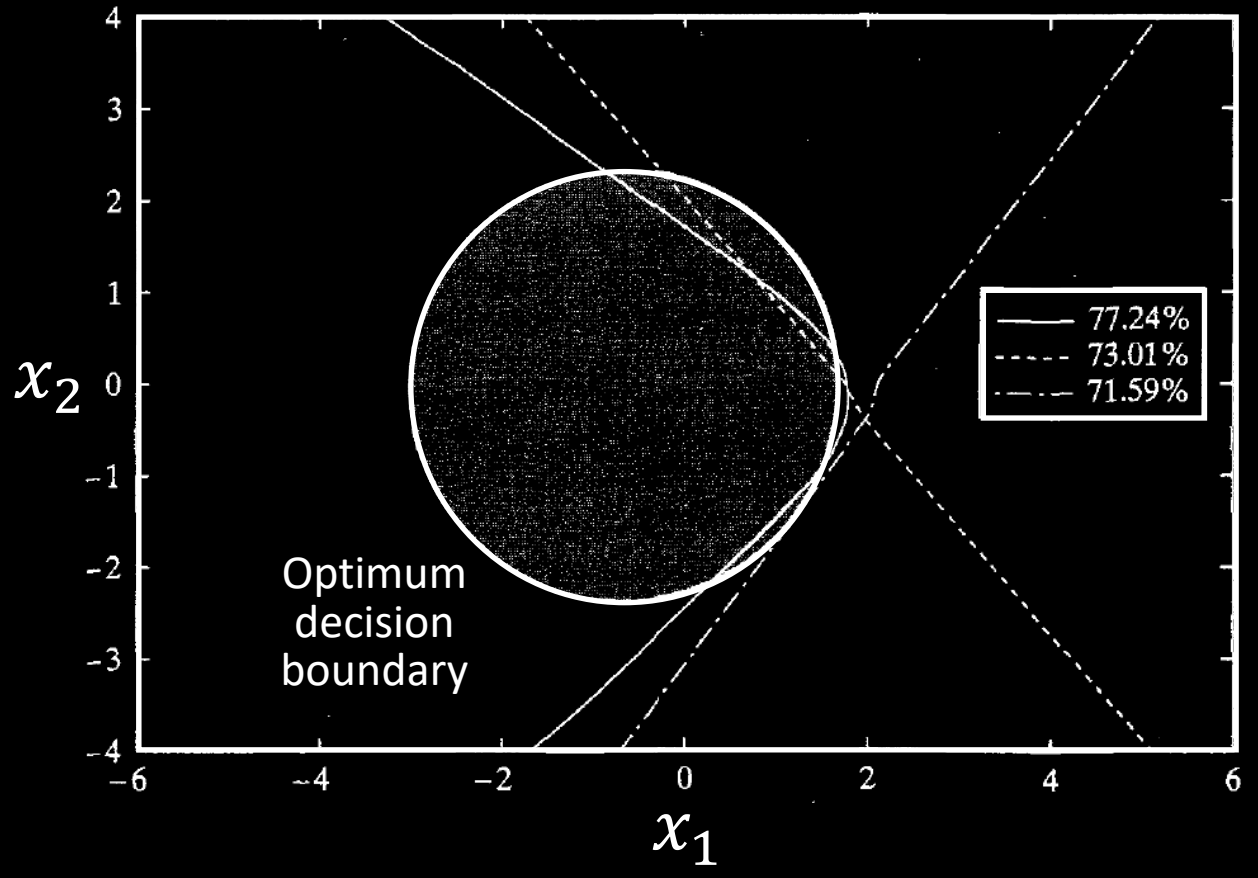


Fig. 4.17 a) Plot of three "best" decision boundaries for the classification accuracies: 80.39, 80.40, and 80.43%. b) Plot of three "poorest" decision boundaries for the following classification accuracies: 77.24, 73.01, and 71.59%.

Evaluation of Optimal Network Design

- The ensemble statistics of the performance measures, probability of correct classification and final mean-squared error, computed over the training sample are listed in Table 4.5.
- The probability of correct classification for the optimum Bayes classifier is 81.51%.

TABLE 4.5 Ensemble Statistics of Performance Measures (Sample Size= 20)

Performance Measure	Mean	Standard Deviation
Probability of correct classification	79.70%	0.44%
Final mean-square error	0.2277	0.0118

End of Section 4.5 - 4.8