Recap

Before we begin

Please ask yourself whether this is the right course for you

- This is a first-level course on ML
- This course follows a statistical (or probability-based) approach to ML.
- Course is more ‘theoretical’ (algorithms and analysis)
- This course does NOT teach any Python programming or use of any standard packages.
- This is not a course on deep learning. Many topics other than neural network models would be covered.
- The course E0 270 is an equivalent course.

Recap

- We consider PR as a two step process – Feature measurement/extraction and Classification (Feature extraction may be implicit or explicit)
- Classifier maps feature (pattern) vectors to Class labels.
- Function learning is a closely related problem.
- The main information we have for the design is a training set of examples.
- In both cases we need to learn from (training) examples.
- In general, most of machine learning is about 'fitting' (probability) models to data.

Recap

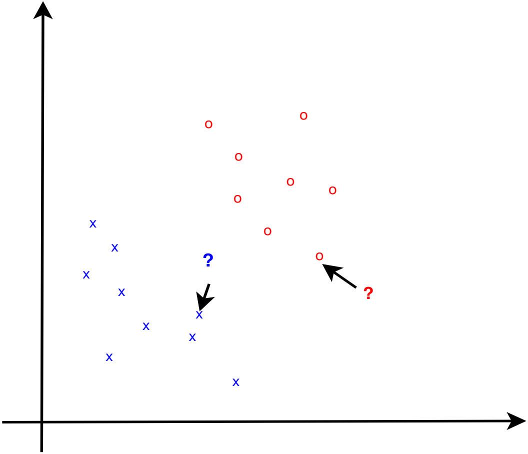
- In statistical pattern recognition, we model variations of feature values through probability distributions.
- The statistical viewpoint gives us ways of looking for 'optimal' classifier.
- For example, Bayes classifier – put the pattern into the class with highest posterior probability.
- This Bayes classifier is optimal in the sense of minimizing probability of misclassification.

- We can have classifiers that do not really make use of any statistical viewpoint.
- For example, the nearest neighbour classifier.

Nearest Neighbour (NN) Classifier (Rule)

- A simple classifier that often performs very well.
- We store some feature vectors from the training set as *prototypes*. (It can be the whole training set).
- Given a new pattern (feature vector) X we find the prototype X' that is closest to X. Then classify X into the same class as X'.
- A variation: k-NN rule.
  Find the k prototypes closest to X. Classify X into the majority class of these prototypes.

Nearest neigbour Classifier



Nearest Neighbour Classifier

- There are two main issues in designing an NN classifier.
  - Selection of Prototypes
  - Distance between feature vectors
- A very simple classifier to design and operate.
- Time and memory needs depend on number of prototypes and complexity of distance function.

- Selection of Prototypes: How many? How to select?
- Distance function: Can use Euclidean distance. $d^2(X, X') = \sum (x_i - x_i')^2$.
- A better method may be $d^2(X, X') = \sum \left(\frac{x_i - x_i'}{\sigma_i}\right)^2$ Here σ_i is the (estimated) variance of i^{th} feature.
- A more general form is:

$$d^2(X, X') = (X - X')^T \Sigma^{-1} (X - X')$$

where Σ is the (estimated) covariance matrix. Called Mahalanobis distance.

- The NN rule does not really use any statistical viewpoint. It is a simple classifier that is often good.
- But one can analyze NN rule from a statistical perspective.
- If we have a sequence of *iid* examples, asymptotically, the probability of error by NN rule is less than twice the Bayes error.
- The NN rule is also related to certain non-parametric methods of estimating class conditional densities

- Now let us go back to Bayes classifier that minimizes probability of misclassification.

Recall notation

- $\mathcal{X}$ – feature space. Usually $\Re^n$.
- $\mathcal{Y}$ – set of class labels.
- $\mathbf{X} = (X_1, \cdots, X_n)^T$ – feature vector.
- A classifier is a function

$$h : \mathcal{X} \rightarrow \mathcal{Y} \; (= \{0, \; 1\})$$

A classifier maps feature vectors to class labels.

Recall Notation

- f_0, f_1 – class conditional densities (these are densities over $\mathcal{X}$)
- p_0, p_1 – prior probabilities.
- q_0, q_1 – posterior probabilities.

Bayes Classifier (2-class case)

- The Bayes classifier

$$\begin{aligned} h_B(\mathbf{x}) &= 0 \ \text{ if } q_0(\mathbf{x}) > q_1(\mathbf{x}) \\ &= 1 \ \text{ otherwise} \end{aligned}$$

- $q_0(\mathbf{x}) > q_1(\mathbf{x})$ is same as $p_0 f_0(\mathbf{x}) > p_1 f_1(\mathbf{x})$.
- Minimizes probability of error in classification.
- Given the underlying probability model, this is the optimal classifier for minimizing probability of error.

Optimality

- Each classifier h maps $\mathcal{X}$ to $\{0, 1\}$.
- For any classifier h, let
  $R_i(h) = \{\mathbf{x} \in \mathcal{X} \ : \ h(\mathbf{x}) = i\}, i = 0, 1.$
- That is, $R_0(h)$ is the set of all feature vectors that get classified as Class-0 by the classifier h.
- Note that $R_0(h) \cap R_1(h) = \phi$ and
  $R_0(h) \cup R_1(h) = \mathcal{X}, \ \forall h.$
- Let $F(h)$ denote probability of error for h.

Optimality

$$
\begin{aligned}
F(h) &= \mathrm{P}[\mathbf{X} \in R_1(h), \mathbf{X} \in \text{C-0}] + \mathrm{P}[\mathbf{X} \in R_0(h), \mathbf{X} \in \text{C-1}] \\
&= \mathrm{P}[\mathbf{X} \in \text{C-0}]\ \mathrm{P}[\mathbf{X} \in R_1(h) | \mathbf{X} \in \text{C-0}] + \\
&\qquad\qquad \mathrm{P}[\mathbf{X} \in \text{C-1}]\ \mathrm{P}[\mathbf{X} \in R_0(h) | \mathbf{X} \in \text{C-1}] \\
&= p_0\ \mathrm{P}[\mathbf{X} \in R_1(h) | \mathbf{X} \in \text{C-0}] + \\
&\qquad\qquad p_1\ \mathrm{P}[\mathbf{X} \in R_0(h) | \mathbf{X} \in \text{C-1}] \\
&= p_0 \int_{R_1(h)} f_0(\mathbf{x}) d\mathbf{x} + p_1 \int_{R_0(h)} f_1(\mathbf{x}) d\mathbf{x}
\end{aligned}
$$

- Note that for each $\mathbf{x}$ we 'add' either $p_0 f_0(\mathbf{x})$ or $p_1 f_1(\mathbf{x})$ in calculating the error integral.

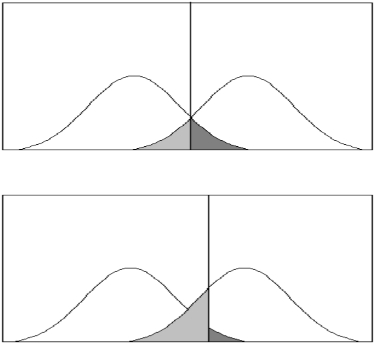
Optimality

- For Bayes Classifier,
  $R_0(h_B) = \{\mathbf{x} \ : \ p_0 f_0(\mathbf{x}) > p_1 f_1(\mathbf{x})\}$ and
  $R_1(h_B) = \{\mathbf{x} \ : \ p_1 f_1(\mathbf{x}) \geq p_0 f_0(\mathbf{x})\}$.
- Then

$$
\begin{aligned}
F(h_B) &= \int_{R_1(h_B)} p_0 f_0(\mathbf{x}) d\mathbf{x} + \int_{R_0(h_B)} p_1 f_1(\mathbf{x}) d\mathbf{x} \\
&= \int_{\mathcal{X}} \min\,(p_0 f_0(\mathbf{x}), p_1 f_1(\mathbf{x})) d\mathbf{x}
\end{aligned}
$$

- Hence Bayes classifier is optimal. (That is, given the knowledge of the probability densities, no other classifier performs better).

Optimality



Bayes Classifier (Contd.)

- Bayes classifier minimizes probability of error (misclassification).
- There are two kind of errors in classification.
- Classifying – C-0 as C-1 or C-1 as C-0
- False Positive or False negative; Type-I or Type-II; False Alarm or Missed detection
- The 'costs' for these errors may be different.
- We may want to trade one type of errors with the other type

Risk of a classifier

- Recall that a more general way to assign figure of merit is to use a **loss function**, $L : \mathcal{Y} \times \mathcal{Y} \rightarrow \Re^+$.
- The risk of the classifier is expectation of loss.

$$F(h) = E[L(h(\mathbf{X}), y(\mathbf{X}))]$$

Bayes Classifier for Minimizing Risk

- We saw Bayes classifier for 0-1 loss and 2-class case.
- Next we consider the Bayes classifier for M classes under a general loss function.
- This can actually be looked at as a special case of a more general problem of decision making under uncertainity.

Bayesian Decision Making

- The task: Decision making under uncertainity
- We want to decide on one of finitely many 'actions' based on some observation/measurement.
- Our 'payoff' or 'cost' depends on the *unknown* 'state of nature'.
- The measured quantity gives some (stochastic) information on the 'state of nature'.
- A Loss function gives 'costs' for each decision for every 'true' state of nature.
- We want a strategy of decision making that minimizes, e.g., expected loss.

In the context of classifier design

- Observation is the feature vector.
- The ‘state of nature’ is the ‘true’ class label of the feature vector.
- We need to decide on a class label based on the observation.

Bayes Classifier

- We now consider the Bayes classifier with M classes and arbitrary loss function.
- $C_0, C_1, \cdots, C_{M-1}$ – the class labels.
  $y(\mathbf{X}) \in \{C_0, \cdots, C_{M-1}\}$.
  (States of Nature)
- Let $h(\mathbf{X}) \in \{\alpha_0, \alpha_1, \cdots, \alpha_{K-1}\}$.
  The out put of classifier would be α_j's.
  (Actions of decision maker)
- In general, we may have $M \neq K$.
  The classifier output need not always be a class label.
- For example, we can have $K = M + 1$ and α_M may denote the decision of 'rejection'.
  (We take $M = K$ unless specified otherwise)

- $L(\alpha_j, C_k)$ – loss when classifier says α_j and 'true class' is C_k. We assume that loss function is non-negative.
- The risk of a classifier h is

$$R(h) = EL(h(\mathbf{X}), y(\mathbf{X}))$$

- We want the classifier that has the least risk value.

- Given a $\mathbf{X}$, let $R(\alpha_i \mid \mathbf{X})$ denote the expected loss when classifier says α_i and conditioned on $\mathbf{X}$.

$$
\begin{aligned}
R(\alpha_i \mid \mathbf{X}) &= E\left[L(h(\mathbf{X}), y(\mathbf{X})) \mid h(\mathbf{X}) = \alpha_i, \mathbf{X}\right] \\
&= E\left[L(\alpha_i, y(\mathbf{X})) \mid \mathbf{X}\right] \\
&= \sum_{j=0}^{M-1} L(\alpha_i, C_j)\mathrm{Prob}(y(\mathbf{X}) = C_j \mid \mathbf{X}) \\
&= \sum_{j=0}^{M-1} L(\alpha_i, C_j) q_j(\mathbf{X})
\end{aligned}
$$

- We saw

$$\begin{aligned} R(\alpha_i \mid \mathbf{X}) &= E\left[L(h(\mathbf{X}), y(\mathbf{X})) \mid h(\mathbf{X}) = \alpha_i, \mathbf{X}\right] \\ &= \sum_{j=0}^{M-1} L(\alpha_i, C_j) q_j(\mathbf{X}) \end{aligned}$$

- In general, we have

$$R(h(\mathbf{X}) \mid \mathbf{X}) = \sum_{j=0}^{M-1} L(h(\mathbf{X}), C_j) q_j(\mathbf{X})$$

- Let f denote the density of $\mathbf{X}$.
- Now risk of any classifier is

$$\begin{aligned} R(h) &= E[L(h(\mathbf{X}), y(\mathbf{X}))] \\ &= E\,[\ E[L(h(\mathbf{X}), y(\mathbf{X})) \mid \mathbf{X}]\ ] \\ &= \int R(h(\mathbf{X}) \mid \mathbf{X}) f(\mathbf{X})\, d\mathbf{X} \end{aligned}$$

- The optimal classifier (could be):

for each $\mathbf{X}$, $h(\mathbf{X})$ *should be such that*

$$R(h(\mathbf{X}) \mid \mathbf{X}) \leq R(h'(\mathbf{X}) \mid \mathbf{X}),\ \forall h'$$

The Bayes Classifier

- Recall $R(h(\mathbf{X}) \mid \mathbf{X}) = \sum_{j=0}^{M-1} L(h(\mathbf{X}), C_j) q_j(\mathbf{X})$
- The Bayes classifier, h_B, for the M-class case is:

  $h_B(\mathbf{X}) = \alpha_i$ *if*

$$\sum_{j=0}^{M-1} L(\alpha_i, C_j) q_j(\mathbf{X}) \leq \sum_{j=0}^{M-1} L(\alpha_k, C_j) q_j(\mathbf{X}), \; \forall k$$

  (Break ties arbitrarily)

- Thus $R(h_B(\mathbf{X}) \mid \mathbf{X}) \leq R(h(\mathbf{X}) \mid \mathbf{X}), \; \forall h$ and thus Bayes classifier is optimal.

2-Class Case

- Take $M = 2$. Now the Bayes classifier is:

$$h_B(\mathbf{X}) = \alpha_0 \;\; \textit{if}$$

$$L(\alpha_0, C_0)q_0(\mathbf{X}) + L(\alpha_0, C_1)q_1(\mathbf{X}) \leq$$

$$L(\alpha_1, C_0)q_0(\mathbf{X}) + L(\alpha_1, C_1)q_1(\mathbf{X})$$

- Same as

$$\frac{q_0(\mathbf{X})}{q_1(\mathbf{X})} \geq \frac{L(\alpha_0, C_1)}{L(\alpha_1, C_0)} \quad \text{if} \quad L(\alpha_0, C_0) = L(\alpha_1, C_1) = 0.$$

- This tells us how we 'trade' the two types of errors.

M-Class case with 0–1 loss

- For M-class case and 0–1 loss function

$$
\begin{aligned}
R(\alpha_i \mid \mathbf{X}) &= \sum_{j=0}^{M-1} L(\alpha_i, C_j) q_j(\mathbf{X}) \\
&= \sum_{j \neq i} q_j(\mathbf{X}) \;=\; 1 - q_i(\mathbf{X})
\end{aligned}
$$

- Thus the Bayes classifier is: $h_B(\mathbf{X}) = \alpha_i$ if

$$
(1 - q_i(\mathbf{X})) \leq (1 - q_j(\mathbf{X})) \;\text{ or }\; q_i(\mathbf{X}) \geq q_j(\mathbf{X}), \;\; \forall j
$$

- This is the M-class classifier for 0–1 loss function.
  Minimizes probability of misclassification.

Bayes Classifier – General Case

- The Bayes classifier that minimizes risk is:

$$h_B(\mathbf{X}) = \alpha_i \quad \textit{if}$$

$$R(\alpha_i \mid \mathbf{X}) \leq R(\alpha_j \mid \mathbf{X}), \ \forall j.$$

  where

$$R(\alpha_i \mid \mathbf{X}) = \sum_{j=0}^{M-1} L(\alpha_i, C_j) q_j(\mathbf{X})$$

  (Break ties arbitrarily)

- Note that this is the most general case. (Even when $L(\alpha_i, C_i) \neq 0$). This is optimal for minimizing risk.

Some special cases

- Let us consider the two class case and explicitly write down Bayes classifier for some specific class conditional densities.
- For simplicity we write $L(\alpha_i, C_j) = L(i, j)$. We also assume $L(0,0) = L(1,1) = 0$.
- For the 2-class case, we decide on C_0 if

$$\frac{q_0(\mathbf{X})}{q_1(\mathbf{X})} = \frac{f_0(\mathbf{X})p_0}{f_1(\mathbf{X})p_1} \geq \frac{L(0,1)}{L(1,0)}$$

Normal class conditional densities

- We start with the simple case of $\mathbf{X} \in \Re$ (hence use X for $\mathbf{X}$) and both class conditional densities normal.

$$f_i(X) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(\frac{-(X-\mu_i)^2}{2\sigma_i^2} \right), \quad i = 0, 1$$

- $h_B(X) = 0$ if

$$p_0 f_0(X) L(1,0) > p_1 f_1(X) L(0,1)$$

- Same as

$$\ln(p_0 L(1,0)) + \ln(f_0(X)) > \ln(p_1 L(0,1)) + \ln(f_1(X))$$

- That is, $h_B(X) = 0$ if

$$\ln(p_0 L(1,0)) - \ln(\sigma_0) - \frac{1}{2}\ln(2\pi) - \frac{(X-\mu_0)^2}{2\sigma_0^2} >$$
$$\ln(p_1 L(0,1)) - \ln(\sigma_1) - \frac{1}{2}\ln(2\pi) - \frac{(X-\mu_1)^2}{2\sigma_1^2}$$

- That is, $h_B(X) = 0$ if

$$\frac{1}{2}X^2\left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_0^2}\right) + X\left(\frac{\mu_0}{\sigma_0^2} - \frac{\mu_1}{\sigma_1^2}\right)$$
$$+\frac{1}{2}\left(\frac{\mu_1^2}{\sigma_1^2} - \frac{\mu_0^2}{\sigma_0^2}\right) + \ln\left(\frac{\sigma_1}{\sigma_0}\right) + \ln\left(\frac{p_0 L(1,0)}{p_1 L(0,1)}\right) > 0$$

- That is, $h_B(X) = 0$ if

$$\frac{1}{2}X^2\left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_0^2}\right) + X\left(\frac{\mu_0}{\sigma_0^2} - \frac{\mu_1}{\sigma_1^2}\right)$$
$$+\frac{1}{2}\left(\frac{\mu_1^2}{\sigma_1^2} - \frac{\mu_0^2}{\sigma_0^2}\right) + \ln\left(\frac{\sigma_1}{\sigma_0}\right) + \ln\left(\frac{p_0 L(1,0)}{p_1 L(0,1)}\right) > 0$$

- This is of the form

$$h_B(X) = 0 \quad \text{if} \quad aX^2 + bX + c > 0$$

where a, b, c are some constants.

- Thus the Bayes classifier in this case is a quadratic discriminant function.

some special cases

- The Bayes classifier is: $h_B(X) = 0$ if
  $\frac{1}{2}X^2\left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_0^2}\right) + X\left(\frac{\mu_0}{\sigma_0^2} - \frac{\mu_1}{\sigma_1^2}\right)$
  $+\frac{1}{2}\left(\frac{\mu_1^2}{\sigma_1^2} - \frac{\mu_0^2}{\sigma_0^2}\right) + \ln\left(\frac{\sigma_1}{\sigma_0}\right) + \ln\left(\frac{p_0 L(1,0)}{p_1 L(0,1)}\right) > 0$
- Take $\sigma_0 = \sigma_1 = \sigma, \quad p_0 = p_1, \quad L(1,0) = L(0,1).$
  Then $h_B(X) = 0$ if

$$\frac{X}{\sigma^2}(\mu_0 - \mu_1) - \frac{1}{2\sigma^2}(\mu_0^2 - \mu_1^2) > 0$$

- That is, $X > \frac{\mu_0+\mu_1}{2}$, assuming $\mu_0 > \mu_1$.
- Intuitively the classifier is very clear.

some special cases

- The Bayes classifier is: $h_B(X) = 0$ if
  $\frac{1}{2}X^2\left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_0^2}\right) + X\left(\frac{\mu_0}{\sigma_0^2} - \frac{\mu_1}{\sigma_1^2}\right)$
  $+\frac{1}{2}\left(\frac{\mu_1^2}{\sigma_1^2} - \frac{\mu_0^2}{\sigma_0^2}\right) + \ln\left(\frac{\sigma_1}{\sigma_0}\right) + \ln\left(\frac{p_0 L(1,0)}{p_1 L(0,1)}\right) > 0$
- Take $\mu_0 = \mu_1 = 0, \quad p_0 = p_1, \quad L(1,0) = L(0,1)$.
  Then $h_B(X) = 0$ if

$$\frac{1}{2}X^2\left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_0^2}\right) - \ln\left(\frac{\sigma_0}{\sigma_1}\right) > 0$$

- Assuming $\sigma_0 > \sigma_1$, this is same as
  $X^2 > \frac{\sigma_1^2\sigma_0^2 \ln(\sigma_0/\sigma_1)}{(\sigma_0^2 - \sigma_1^2)}$
  (again, intuitively clear).

- Now let us consider the case of $\mathbf{X} \in \Re^n$ and normal class conditional densities.

  $$f_i(\mathbf{X}) = ((2\pi)^n|\Sigma_i|)^{-\frac{1}{2}} \exp(-\frac{1}{2}(\mathbf{X} - \boldsymbol{\mu}_i)^T \Sigma_i^{-1} (\mathbf{X} - \boldsymbol{\mu}_i)), \quad i = 0, 1$$

- The Bayes classifier is: $h_B(\mathbf{X}) = 0$ if

  $$\ln(p_0 L(1,0)) + \ln(f_0(\mathbf{X})) > \ln(p_1 L(0,1)) + \ln(f_1(\mathbf{X}).$$

- That is,

  $$\ln(p_0 L(1,0)) - \frac{1}{2}\ln(|\Sigma_0|) - \frac{1}{2}(\mathbf{X} - \boldsymbol{\mu}_0)^T \Sigma_0^{-1} (\mathbf{X} - \boldsymbol{\mu}_0) > \ln(p_1 L(0,1)) - \frac{1}{2}\ln(|\Sigma_1|) - \frac{1}{2}(\mathbf{X} - \boldsymbol{\mu}_1)^T \Sigma_1^{-1} (\mathbf{X} - \boldsymbol{\mu}_1)$$

- We have $h_B(\mathbf{X}) = 0$ if

$$\ln(p_0 L(1,0)) - \frac{1}{2}\ln(|\Sigma_0|) - \frac{1}{2}(\mathbf{X}-\boldsymbol{\mu}_0)^T \Sigma_0^{-1}(\mathbf{X}-\boldsymbol{\mu}_0) > \ln(p_1 L(0,1)) - \frac{1}{2}\ln(|\Sigma_1|) - \frac{1}{2}(\mathbf{X}-\boldsymbol{\mu}_1)^T \Sigma_1^{-1}(\mathbf{X}-\boldsymbol{\mu}_1)$$

- That is

$$\frac{1}{2}\mathbf{X}^T(\Sigma_1^{-1} - \Sigma_0^{-1})\mathbf{X} + \mathbf{X}^T(\Sigma_0^{-1}\boldsymbol{\mu}_o - \Sigma_1^{-1}\boldsymbol{\mu}_1) + \frac{1}{2}(\boldsymbol{\mu}_1^T \Sigma_1^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0^T \Sigma_0^{-1}\boldsymbol{\mu}_0) + \ln\left(\frac{p_0 L(1,0)}{p_1 L(0,1)}\right) + \frac{1}{2}\ln\left(\frac{|\Sigma_1|}{|\Sigma_0|}\right) > 0$$

- Once again, the Bayes classifier is a quadratic discriminant function.

- The Bayes classifier is based on the discriminant function

$$\begin{aligned}&\tfrac{1}{2}\mathbf{X}^T(\Sigma_1^{-1}-\Sigma_0^{-1})\mathbf{X}+\mathbf{X}^T(\Sigma_0^{-1}\boldsymbol{\mu}_o-\Sigma_1^{-1}\boldsymbol{\mu}_1)\\&\quad+\tfrac{1}{2}(\boldsymbol{\mu}_1^T\Sigma_1^{-1}\boldsymbol{\mu}_1-\boldsymbol{\mu}_0^T\Sigma_0^{-1}\boldsymbol{\mu}_0)\\&+\ln\left(\tfrac{p_0L(1,0)}{p_1L(0,1)}\right)+\tfrac{1}{2}\ln\left(\tfrac{|\Sigma_1|}{|\Sigma_0|}\right)>0\end{aligned}$$

- Consider the specisl case $\Sigma_i=\Sigma$.
- Then the quadratic term Vanishes.
- The Bayes classifier now becomes a linear discriminant function.

- In the special case $\Sigma_i = \Sigma$, the Bayes classifier is:
- $h_B(\mathbf{X}) = 0 \;\text{ if }\; g(\mathbf{X}) > 0, \;\text{ where}$

  $$g(\mathbf{X}) = \mathbf{W}^T \mathbf{X} + w_0, \;\text{ with}$$

  $$\mathbf{W} = \Sigma^{-1}(\boldsymbol{\mu}_0 - \boldsymbol{\mu}_1)$$
  $$w_0 = \tfrac{1}{2}(\boldsymbol{\mu}_1^T \Sigma^{-1} \boldsymbol{\mu}_1 - \boldsymbol{\mu}_0^T \Sigma^{-1} \boldsymbol{\mu}_0) + \ln\left(\tfrac{p_0 L(1,0)}{p_1 L(0,1)}\right)$$

Example: Binary Features

- Let $\mathbf{X} \in \{0,\ 1\}^d$
- Let $\text{Prob}[X_i = 1|\mathbf{C_0}] = p_i^0$ and $\text{Prob}[X_i = 1|\mathbf{C_1}] = p_i^1$
- Assume features are independent. Then we have

$$f(\mathbf{x}|\mathbf{C_0}) = \prod_{i=1}^{d} (p_i^0)^{x_i} \ (1 - p_i^0)^{(1-x_i)}$$

  where $\mathbf{x} = (x_1, \cdots, x_d) \in \{0,\ 1\}^d$.
- $f(\mathbf{x}|\mathbf{C_1})$ can be written similarly. .

Document Classification

- Binary features can be used for document classification.
- We can use 'bag of words' representation. The dimension of feature vector is size of dictionary.
- We take $x_i = 1$ if i^{th} word is present in the document.
- Then p_i^0 is the probability i^{th} word is present in a 'positive-class' document.

- Now, $h_B(\mathbf{x}) = 1$ if

$$\prod_{i=1}^{d} (p_i^0)^{x_i} \, (1 - p_i^0)^{(1-x_i)} \leq \prod_{i=1}^{d} (p_i^1)^{x_i} \, (1 - p_i^1)^{(1-x_i)}$$

- Equivalently

$$\sum_{i=1}^{d} \left(x_i \ln \left(\frac{p_i^0}{p_i^1} \right) + (1 - x_i) \ln \left(\frac{1 - p_i^0}{1 - p_i^1} \right) \right) \leq 0$$

This is a linear classifier

- A special case: $p_i^0 = p, p_i^1 = (1-p), \forall i$ and $p > 0.5$.
- Now, $h_B(\mathbf{x}) = 1$ if

$$\prod_{i=1}^{d} (p)^{x_i} \, (1-p)^{(1-x_i)} \leq \prod_{i=1}^{d} (1-p)^{x_i} \, (p)^{(1-x_i)}$$

Same as

$$p^{2\sum x_i - d} \leq (1-p)^{2\sum x_i - d}$$

- Since $0 < (1-p) < p < 1$, this is same as $\sum x_i < d/2$.

Bayes Classifier

- Bayes classifier can be similarly derived for any other class conditional densities.
- For example, in a 2-class case with 0-1 loss function, given a $\mathbf{X}$, we decide on the class based on whether or not the inequality $p_0 f_0(\mathbf{X}) > p_1 f_1(\mathbf{X})$ is satisfied.
- Depending on the nature of the densities the final expressions can be complicated.
- Given full statistical information, this is the optimal decision.

Finding Bayes Error

- Given class conditional densities, the Bayes classifier is easily computed.
- We may also want to compute the Bayes error.
- Gives us the expected performance. Also lets us decide whether we need better features.
- For the case of 0-1 loss function, we need to evaluate

$$\int_{\Re^n} \min(p_0 f_0(\mathbf{X}),\ p_1 f_1(\mathbf{X}))\ d\mathbf{X}$$

- In general, a difficult integral to evaluate.

- Let us consider the simplest case:
  2-class problem, $X \in \Re$, normal class conditional densities and 0-1 loss function.
- Assume equal priors. Let $\sigma_0 = \sigma_1 = \sigma$ and $\mu_0 < \mu_1$.
- Then $h_B(X) = 0$ if $X < (\mu_0 + \mu_1)/2$.
- Then, Bayes error is

$$P(\text{error}) = R(h_B) = 0.5 \int_{-\infty}^{\frac{\mu_0+\mu_1}{2}} f_1(X)\, dX + 0.5 \int_{\frac{\mu_0+\mu_1}{2}}^{\infty} f_0(X)\, dX$$

$$
\begin{aligned}
P(\text{error}) &= 0.5 \int_{-\infty}^{\frac{\mu_0+\mu_1}{2}} f_1(X)\, dX + 0.5 \int_{\frac{\mu_0+\mu_1}{2}}^{\infty} f_0(X)\, dX \\
&= 0.5\, \Phi\left(\frac{\mu_0 - \mu_1}{2\sigma}\right) + 0.5\left[1 - \Phi\left(\frac{\mu_1 - \mu_0}{2\sigma}\right)\right]
\end{aligned}
$$

because the substitution, $Z = (X - \mu_i)/\sigma$ makes the two integrals into integrals of the standard normal distribution.

Here, Φ is the distribution function of the Standard Normal random Variable.

The quantity $\frac{|\mu_0 - \mu_1|}{\sigma}$ is called *discriminability*.

- In the general case, we need to evaluate
  $P(\text{error}) = \int_{\Re^n} \min(p_0 f_0(\mathbf{X}),\ p_1 f_1(\mathbf{X}))\ d\mathbf{X}$
- A useful inequality here is
  $\min(a,\ b) \leq a^\beta b^{1-\beta},\ \forall\ a, b \geq 0,\ 0 \leq \beta \leq 1.$
- Easy to prove. Suppose $a < b$
  $a^\beta b^{1-\beta} = a^{-1+\beta} b^{1-\beta} a = \left(\frac{b}{a}\right)^{1-\beta} a \geq a = \min(a,\ b)$
- Hence we have (for 0-1 loss function)

$$P(\text{error})\ \leq\ p_0^\beta p_1^{1-\beta} \int_{\Re^n} f_0^\beta(\mathbf{X}) f_1^{1-\beta}(\mathbf{X})\, d\mathbf{X}$$

- In special cases, we can derive bounds on this.

Other Criteria

- The Bayes classifier is optimal for the criterion of risk minimization.
- There can be other criteria.
- The Bayes classifier depends on both p_i, prior probabilities, and f_i, class conditional densities.
- Suppose we do not want to rely on prior probabilities.
- We may want a classifier that does best against any (or worst) prior probabilities.

- Consider a 2-class case.
- Let $\mathcal{R}_i(h)$ denote the subset of feature space where h classifies into Class-i.
- Then the Risk integral is

$$R(h) = \int_{\mathcal{R}_1(h)} L(1,0) p_0 f_0(\mathbf{X}) d\mathbf{X} + \int_{\mathcal{R}_0(h)} L(0,1) p_1 f_1(\mathbf{X}) d\mathbf{X}$$

- We can simplify this to get rid of dependence on priors.

- Using $p_0 = 1 - p_1$, we get (writing R for $R(h)$ and so on)

$$
\begin{aligned}
R &= \int_{\mathcal{R}_1} L(1,0) p_0 f_0(\mathbf{X})\, d\mathbf{X} + \int_{\mathcal{R}_0} L(0,1) p_1 f_1(\mathbf{X})\, d\mathbf{X} \\
&= L(1,0) p_0 \int_{\mathcal{R}_1} f_0(\mathbf{X})\, d\mathbf{X} + L(0,1)(1 - p_0) \int_{\mathcal{R}_0} f_1(\mathbf{X})\, d\mathbf{X} \\
&= L(0,1) \int_{\mathcal{R}_0} f_1(\mathbf{X})\, d\mathbf{X} + \\
&\qquad p_0 \left[L(1,0) \int_{\mathcal{R}_1} f_0(\mathbf{X})\, d\mathbf{X} - L(0,1) \int_{\mathcal{R}_0} f_1(\mathbf{X})\, d\mathbf{X} \right]
\end{aligned}
$$

For a fixed classifier, risk varies linearly with the prior probability.

Minmax Classifier

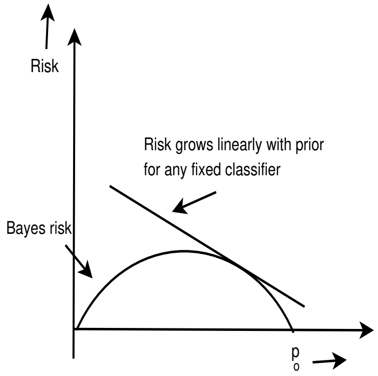
- Consider a classifier such that

$$L(1,0)\int_{\mathcal{R}_1} f_0(\mathbf{X})\,d\mathbf{X} = L(0,1)\int_{\mathcal{R}_0} f_1(\mathbf{X})\,d\mathbf{X}$$

- For this classifier the risk would be $L(0,1)\int_{\mathcal{R}_0} f_1(\mathbf{X})\,d\mathbf{X}$. Also, risk would be independent of priors.
- Called the **minmax** classifier
- We are minimizing the maximum possible (over all priors) risk.
- In general, finding the minmax classifier can be analytically complicated.

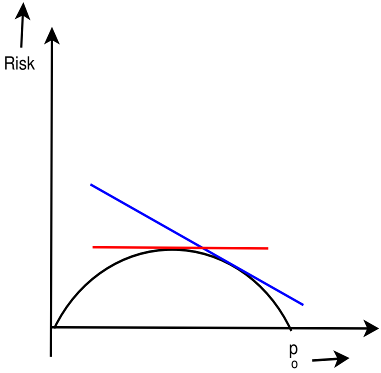
minmax classifier

- We can see it graphically as follows.



minmax classifier

- So, the minmax classifier would be



- Suppose $L(0, 1) = L(1, 0)$.
- Then Minimax classifier is one that achieves

$$\int_{\mathcal{R}_1} f_0(\mathbf{X})\, d\mathbf{X} = \int_{\mathcal{R}_0} f_1(\mathbf{X})\, d\mathbf{X}$$

- For the simple case of one dimensional features and normal class conditional densities, we can easily derive the minimax classifier.

- Let f_i be normal with mean μ_i and variance σ_i^2. Assume $\mu_0 < \mu_1$.
- Consider the classifier

$$h(X) = 0 \text{ iff } X < a$$

- We will fix the threshold, a, to satisfy the minimax criterion.

- We need

$$\int_a^\infty f_0(\mathbf{X})\, d\mathbf{X} = \int_{-\infty}^a f_1(\mathbf{X})\, d\mathbf{X}$$

- These become integrals of standard normal by using $Z = (X - \mu_0)/\sigma_0$ in the first one and $Z = (X - \mu_1)/\sigma_1$ in the second one.
- Thus the threshold a should satisfy

$$1 - \Phi\left(\frac{a - \mu_0}{\sigma_0}\right) = \Phi\left(\frac{a - \mu_1}{\sigma_1}\right)$$

- Using $1 - \Phi(x) = \Phi(-x)$, the condition on a is

$$\Phi\left(\frac{\mu_0 - a}{\sigma_0}\right) = \Phi\left(\frac{a - \mu_1}{\sigma_1}\right) \Rightarrow \frac{\mu_0 - a}{\sigma_0} = \frac{a - \mu_1}{\sigma_1}$$

- Thus we get a as

$$a = \frac{\mu_0\sigma_1 + \mu_1\sigma_0}{\sigma_0 + \sigma_1}$$

- Minimax classifier here is linear while Bayes classifier for this case is quadratic.
- If $\sigma_0 = \sigma_1$ then minimax is same as Bayes in this special case.