Course: DD2427 - Exercise Class 1

Questions with an asterix(*) are a bit more involved and are more to aid understanding as opposed to representing potentional exam questions.

Exercises 1: *Bayes I*

You have written a face detection algorithm. Let a denote the variable that there is a face in the image and b the output of your algorithm.

$$a = \begin{cases} 1 & \text{if there is a face in the image} \\ 0 & \text{there is not a face in the image} \end{cases} \qquad b = \begin{cases} 1 & \text{your algorithm reports there's a face in the image} \\ 0 & \text{your algorithm reports there's not a face in the image} \end{cases}$$

Your face detection algorithm has a false positive rate of .05 and a true positive rate of .85. Your algorithm is examining images that are taken from your front door.

You run your algorithm on an image taken at 10am (the time when the postman usually passes your house) and the result is positive. What is the probability the image contains a face ?

You run your algorithm on an image taken at 2am and the result is positive. What is the probability the image contains a face ?

Solution:

Know:

We are told the false positive rate is .05 therefore

$$p(b = 1 \,|\, a = 0) = .05 \quad p(b = 0 \,|\, a = 0) = .95$$

and the true positive rate is .85 therefore

$$p(b = 1 \,|\, a = 1) = .85 \quad p(b = 0 \,|\, a = 1) = .15$$

Want to calculate:

$$p(a = 1 \,|\, b = 1)$$

From Bayes' Rule:

$$\begin{aligned} p(a = 1 \,|\, b = 1) &= \frac{p(b = 1 \,|\, a = 1)\, p(a = 1)}{p(b = 1)} \\ &= \frac{p(b = 1 \,|\, a = 1)\, p(a = 1)}{p(b = 1 \,|\, a = 0)\, p(a = 0) + p(b = 1 \,|\, a = 1)\, p(a = 1)} \\ &= \frac{.85 \times p(a = 1)}{.05 \times p(a = 0) + .85 \times p(a = 1)} \end{aligned}$$

When you run your algorithm at 10am it is reasonable to assume

$$p(a = 1) > p(a = 0)$$

as this is the time the postman visits your house. Thus let's set $p(a = 1) = .8$ and this implies $p(a = 0) = .2$.

Therefore, at 10am

$$p(a = 1 \,|\, b = 1) = \frac{.85 \times .8}{.05 \times .2 + .85 \times .8} = .9855$$

While if you run your algorithm at 2am it is reasonable to assume

$$p(a = 0) > p(a = 1)$$

as this is the time the postman visits your house. Thus let's set $p(a = 1) = .2$ and this implies $p(a = 0) = .8$.

Therefore, at 2am

$$p(a = 1 \,|\, b = 1) = \frac{.85 \times .2}{.05 \times .8 + .85 \times .2} = .8095$$

Exercises 2: *Bayes Decision Theory*

A binary 2×2 image is generated by some random mechanism. By studying a large number of noise free realizations of the images generated it has been found that

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \text{ has probability } \tfrac{1}{4},$$
$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ has probability } \tfrac{1}{4},$$
$$\begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \text{ has probability } \tfrac{1}{2}$$

(a priori probabilities). One of these images has been distorted by noise in the sense that the value of a pixel has been changed with probability ϵ, that is

$$P(\text{observing } 0 \,|\, \text{the correct value is } 1) = P(\text{observing } 1 \,|\, \text{the correct value is } 0) = \epsilon$$

Assume that the noise in different pixels is independent. Now consider the image

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$$

Using Bayes theorem calculate the MAP (maximum a posterior) estimation of the scene if

1. $\epsilon = 10\%$
2. $\epsilon = 50\%$

Solution:

Know:

There are 3 possible images

$$I_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \text{ and } I_3 = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}.$$

and we observe the image $I = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$

Want to calculate:

$$P(\text{true image } I_i \,|\, \text{observed } I) \text{ for } i = 1, 2, 3.$$

Step 1 - Likelihood Calculations

Calculate the likelihood of generating the observation I given that the true image is I_i for $i = 1, 2, 3$. In these calculations we exploit the fact that the noise in different pixels are independent.

$$\begin{aligned}
P(\text{observe } I \,|\, \text{image } I_1) &= P(\text{observe } \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \,|\, \text{true image } \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}) \\
&= P(\text{observe } 0 \,|\, \text{true value } 0) \times P(\text{observe } 1 \,|\, \text{true value } 1) \times P(\text{observe } 1 \,|\, \text{true value } 1) \\
&\times P(\text{observe } 1 \,|\, \text{true value } 0) \\
&= (1-\epsilon)^3 \, \epsilon
\end{aligned}$$

$$\begin{aligned}
P(\text{observe } I \,|\, \text{image } I_2) &= P(\text{observe } \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \,|\, \text{true image } \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}) \\
&= P(\text{observe } 0 \,|\, \text{true value } 1) \times P(\text{observe } 1 \,|\, \text{true value } 0) \times P(\text{observe } 1 \,|\, \text{true value } 0) \\
&\times P(\text{observe } 1 \,|\, \text{true value } 1) \\
&= (1-\epsilon) \, \epsilon^3
\end{aligned}$$

$$\begin{aligned}
P(\text{observe } I \,|\, \text{image } I_3) &= P(\text{observe } \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \,|\, \text{true image } \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}) \\
&= P(\text{observe } 0 \,|\, \text{true value } 1) \times P(\text{observe } 1 \,|\, \text{true value } 1) \times P(\text{observe } 1 \,|\, \text{true value } 1) \\
&\times P(\text{observe } 1 \,|\, \text{true value } 0) \\
&= (1-\epsilon)^2 \, \epsilon^2
\end{aligned}$$

Step 2 - calculate the posterior probabilities

From Bayes' Rule we get for $i = 1, 2, 3$

$$P(\text{image } I_i \,|\, \text{observe } I) = \frac{P(\text{observe } I \,|\, \text{image } I_i) \, P(\text{image } I_i)}{P(\text{observe } I)} \tag{1}$$

Now

$$\begin{aligned}
P(\text{observe } I) &= \sum_{i=1}^{3} P(\text{observe } I \,|\, \text{image } I_i) \, P(\text{image } I_i) \\
&= \frac{1}{4}(1-\epsilon)^3 \, \epsilon + \frac{1}{4}(1-\epsilon) \, \epsilon^3 + \frac{1}{2}(1-\epsilon)^2 \, \epsilon^2 \\
&= \frac{1}{4}(1-\epsilon) \, \epsilon
\end{aligned}$$

Substitute the appropriate values into equation (**??**) and obtain

$$P(\text{image } I_1 \,|\, \text{observe } I) = \frac{\frac{1}{4}(1-\epsilon)^3 \, \epsilon}{\frac{1}{4}(1-\epsilon) \, \epsilon} = (1-\epsilon)^2$$

$$P(\text{image } I_2 \,|\, \text{observe } I) = \frac{\frac{1}{4}(1-\epsilon) \, \epsilon^3}{\frac{1}{4}(1-\epsilon) \, \epsilon} = \epsilon^2$$

$$P(\text{image } I_3 \,|\, \text{observe } I) = \frac{\frac{1}{2}(1-\epsilon)^2 \, \epsilon^2}{\frac{1}{4}(1-\epsilon) \, \epsilon} = 2(1-\epsilon) \, \epsilon$$

Step 3 - Plug in the value of ϵ

When $\epsilon = .1$ the MAP estimate of the scene is I_1 as

$$P(\text{image } I_1 \,|\, \text{observe } I) = (1-\epsilon)^2 = .81$$

$$P(\text{image } I_2 \,|\, \text{observe } I) = \epsilon^2 = .01$$

$$P(\text{image } I_3 \,|\, \text{observe } I) = 2(1-\epsilon)\,\epsilon = .18$$

While if $\epsilon = \frac{1}{2}$ the MAP estimate of the scene is I_3 as

$$P(\text{image } I_1 \,|\, \text{observe } I) = (1-\epsilon)^2 = \frac{1}{4}$$

$$P(\text{image } I_2 \,|\, \text{observe } I) = \epsilon^2 = \frac{1}{4}$$

$$P(\text{image } I_3 \,|\, \text{observe } I) = 2(1-\epsilon)\,\epsilon = \frac{1}{2}$$

Note in this case the posterior distribution is the same as the prior. This is logical as if $\epsilon = \frac{1}{2}$ then the sensor gives a random response.

Exercises 3:

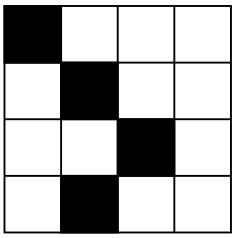
Consider a binary 4×4 image of a scene with a vertical line. In the *correct* image all pixels would be white except one vertical row with black pixels. Unfortunately, the camera used is far from perfect. Errors in different pixels are independent with

$$p(\text{white} \,|\, \text{line}) = p(\text{black} \,|\, \text{not line}) = \epsilon$$

and consequently

$$p(\text{black} \,|\, \text{line}) = p(\text{white} \,|\, \text{not line}) = 1 - \epsilon$$

Assume the a priori probability for the line to be located in column 1 or 4 is 0.3 (each) and the a priori probability that the line is in column 2 or 3 is 0.2 (each). Calculate the maximum a posteriori estimation of the following image when $\epsilon = 0.2$



Solution:

Know:

The 4 possible scenes are

$$I_1 = \begin{bmatrix} x & o & o & o \\ x & o & o & o \\ x & o & o & o \\ x & o & o & o \end{bmatrix}, \; I_2 = \begin{bmatrix} o & x & o & o \\ o & x & o & o \\ o & x & o & o \\ o & x & o & o \end{bmatrix}, \; I_3 = \begin{bmatrix} o & o & x & o \\ o & o & x & o \\ o & o & x & o \\ o & o & x & o \end{bmatrix}, \; I_4 = \begin{bmatrix} o & o & o & x \\ o & o & o & x \\ o & o & o & x \\ o & o & o & x \end{bmatrix}$$

From the camera observe

$$I = \begin{bmatrix} x & o & o & o \\ o & x & o & o \\ o & o & x & o \\ o & x & o & o \end{bmatrix}$$

Want to calculate:

$$p(\text{observe } I \,|\, \text{true scene } I_i) \text{ for } i = 1, 2, 3, 4.$$

Step 1 - Likelihood calculations

$$P(\text{observe } I \,|\, \text{scene } I_1) = P(\text{observe } \begin{bmatrix} x & o & o & o \\ o & x & o & o \\ o & o & x & o \\ o & x & o & o \end{bmatrix} \,|\, \text{scene } \begin{bmatrix} x & o & o & o \\ x & o & o & o \\ x & o & o & o \\ x & o & o & o \end{bmatrix})$$
$$= P(\text{observe } x \,|\, \text{on line})\, P(\text{observe } o \,|\, \text{on line})^3\, P(\text{observe } x \,|\, \text{not on line})^3\, P(\text{observe } o \,|\, \text{not on line})^9$$
$$= (1-\epsilon)\,\epsilon^3\,\epsilon^3\,(1-\epsilon)^9 = (1-\epsilon)^{10}\,\epsilon^6$$

$$P(\text{observe } I \,|\, \text{scene } I_2) = P(\text{observe } \begin{bmatrix} x & o & o & o \\ o & x & o & o \\ o & o & x & o \\ o & x & o & o \end{bmatrix} \,|\, \text{scene } \begin{bmatrix} o & x & o & o \\ o & x & o & o \\ o & x & o & o \\ o & x & o & o \end{bmatrix})$$
$$= P(\text{observe } x \,|\, \text{on line})^2\, P(\text{observe } o \,|\, \text{on line})^2\, P(\text{observe } x \,|\, \text{not on line})^2\, P(\text{observe } o \,|\, \text{not on line})^{10}$$
$$= (1-\epsilon)^2\,\epsilon^2\,\epsilon^2\,(1-\epsilon)^{10} = (1-\epsilon)^{12}\,\epsilon^4$$

$$P(\text{observe } I \,|\, \text{scene } I_3) = P(\text{observe } \begin{bmatrix} x & o & o & o \\ o & x & o & o \\ o & o & x & o \\ o & x & o & o \end{bmatrix} \,|\, \text{scene } \begin{bmatrix} o & o & x & o \\ o & o & x & o \\ o & o & x & o \\ o & o & x & o \end{bmatrix})$$
$$= P(\text{observe } x \,|\, \text{on line})\, P(\text{observe } o \,|\, \text{on line})^3\, P(\text{observe } x \,|\, \text{not on line})^3\, P(\text{observe } o \,|\, \text{not on line})^9$$
$$= (1-\epsilon)\,\epsilon^3\,\epsilon^3\,(1-\epsilon)^9 = (1-\epsilon)^{10}\,\epsilon^6$$

$$P(\text{observe } I \,|\, \text{scene } I_4) = P(\text{observe } \begin{bmatrix} x & o & o & o \\ o & x & o & o \\ o & o & x & o \\ o & x & o & o \end{bmatrix} \,|\, \text{scene } \begin{bmatrix} o & o & o & x \\ o & o & o & x \\ o & o & o & x \\ o & o & o & x \end{bmatrix})$$
$$= P(\text{observe } x \,|\, \text{on line})^0\, P(\text{observe } o \,|\, \text{on line})^4\, P(\text{observe } x \,|\, \text{not on line})^4\, P(\text{observe } o \,|\, \text{not on line})^8$$
$$= (1-\epsilon)^0\,\epsilon^4\,\epsilon^4\,(1-\epsilon)^8 = (1-\epsilon)^8\,\epsilon^8$$

Step 2 - Calculate the posterior probabilities

From Bayes' Rule we get for $i = 1, 2, 3, 4$

$$P(\text{scene } I_i \,|\, \text{observe } I) = \frac{P(\text{observe } I \,|\, \text{scene } I_i)\, P(\text{image } I_i)}{P(\text{observe } I)} \qquad (2)$$

Now

$$P(\text{observe } I) = \sum_{i=1}^{4} P(\text{observe } I \,|\, \text{scene } I_i)\, P(\text{scene } I_i)$$
$$= \frac{3}{10}(1-\epsilon)^{10}\,\epsilon^6 + \frac{2}{10}(1-\epsilon)^{12}\,\epsilon^4 + \frac{2}{10}(1-\epsilon)^{10}\,\epsilon^6 + \frac{3}{10}(1-\epsilon)^8\,\epsilon^8$$
$$= \frac{1}{10}(1-\epsilon)^8\,\epsilon^4\,(1-2\epsilon+2\epsilon^2)\,(5\epsilon^2-4\epsilon+2)$$

Substitute the appropriate values into equation (**??**) to obtain

$$P(\text{scene } I_1 \,|\, \text{observe } I) = \frac{\frac{3}{10}(1-\epsilon)^{10}\,\epsilon^6}{\frac{1}{10}(1-\epsilon)^8\,\epsilon^4\,(1-2\epsilon+2\epsilon^2)\,(5\epsilon^2-4\epsilon+2)} = \frac{3(1-\epsilon)^2\,\epsilon^2}{(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)}$$

$$P(\text{scene } I_2 \,|\, \text{observe } I) = \frac{\frac{2}{10}(1-\epsilon)^{12}\,\epsilon^4}{\frac{1}{10}(1-\epsilon)^8\,\epsilon^4\,(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)} = \frac{2(1-\epsilon)^4}{(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)}$$

$$P(\text{scene } I_3 \,|\, \text{observe } I) = \frac{\frac{2}{10}(1-\epsilon)^{10}\,\epsilon^6}{\frac{1}{10}(1-\epsilon)^8\,\epsilon^4\,(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)} = \frac{2(1-\epsilon)^2\,\epsilon^2}{(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)}$$

$$P(\text{scene } I_4 \,|\, \text{observe } I) = \frac{\frac{3}{10}(1-\epsilon)^{8}\,\epsilon^8}{\frac{1}{10}(1-\epsilon)^8\,\epsilon^4\,(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)} = \frac{3\epsilon^4}{(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)}$$

Step 3 - Plug in the value of $\epsilon = .2$

When $\epsilon = .2$ the MAP estimate of the scene is I_2 as

$$P(\text{scene } I_1 \,|\, \text{observe } I) = \frac{3(1-\epsilon)^2\,\epsilon^2}{(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)} = .0807$$

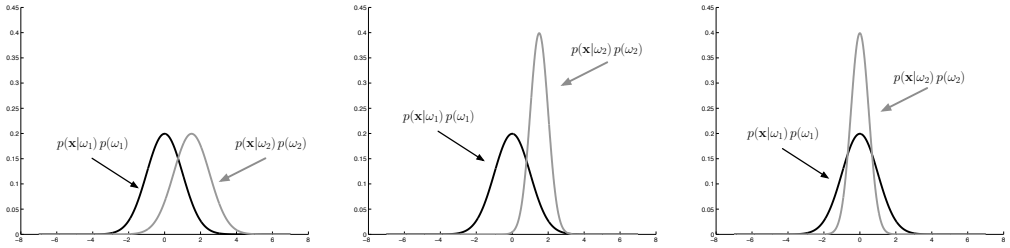
$$P(\text{scene } I_2 \,|\, \text{observe } I) = \frac{2(1-\epsilon)^4}{(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)} = .8605$$

$$P(\text{scene } I_3 \,|\, \text{observe } I) = \frac{2(1-\epsilon)^2\,\epsilon^2}{(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)} = .0538$$

$$P(\text{scene } I_4 \,|\, \text{observe } I) = \frac{3\epsilon^4}{(1-2\epsilon+2\epsilon^2)(5\epsilon^2-4\epsilon+2)} = .005$$

Exercises 4: *Bayes' Decision Theory*

Assume you have a two class classification problem. Each class generates a one dimensional feature vector according to $p(x|\omega_i) = \mathcal{N}(\mu_i, \sigma_i^2)$ for $i = 1, 2$. The prior probabilities for each class are $p(\omega_1) = p(\omega_2) = .5$. In the graphs below $p(x|\omega_i)\, p(\omega_i)$ for $i = 1, 2$ are shown for different values of the μ's and σ's. For each example $\mu_1 = 0, \sigma_1 = 1$ and then **1**) $\mu_2 = 1.5, \sigma_2 = 1$, **2**) $\mu_2 = 1.5, \sigma_2 = .5$ and **3**) $\mu_2 = 0, \sigma_2 = .5$



(**1**) $\mu_1 \neq \mu_2, \sigma_1 = \sigma_2$ (**2**) $\mu_1 \neq \mu_2, \sigma_1 > \sigma_2$ (**3**) $\mu_1 = \mu_2, \sigma_1 > \sigma_2$

i) For the two-class problem how is the *Bayes' Classifier* defined?

ii) In the figure draw the decision boundaries/boundary defined by a Bayes' classifier.

iii) For case (**2**) explicitly calculate the decision boundaries.

iv) For case (**2**) write down the $P(\text{error})$ for the *Bayes' Classifier* and show in a diagram where the errors are being made.

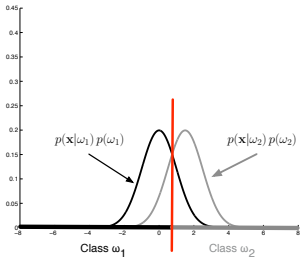
v) What is optimal about the *Bayes' Classifier*?

Solution:

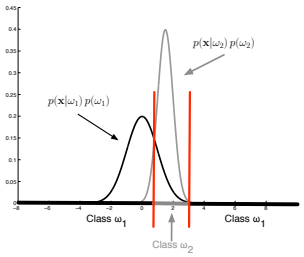
i) The *Bayes' Classifier* for the two class problem

$$\text{Class}\,\{\mathbf{x}\} = \begin{cases} \omega_1 & \text{if } p(\omega_1\,|\,\mathbf{x}) \geq p(\omega_2\,|\,\mathbf{x}) \\ \omega_2 & \text{if } p(\omega_1\,|\,\mathbf{x}) < p(\omega_2\,|\,\mathbf{x}) \end{cases}$$

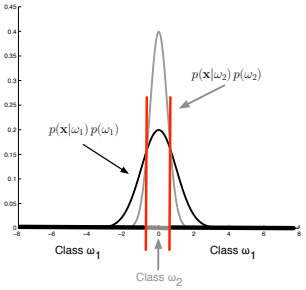
ii) The decision boundaries/boundary defined by the Bayes' classifier:



(1) $\mu_1 \neq \mu_2, \sigma_1 = \sigma_2$



(2) $\mu_1 \neq \mu_2, \sigma_1 > \sigma_2$



(3) $\mu_1 = \mu_2, \sigma_1 > \sigma_2$

iii) The decision boundary is defined by

$$p(\omega_1 \,|\, x) = p(\omega_2 \,|\, x)$$

As $p(\omega_1) = p(\omega_2) = .5$ the above decision boundary is equivalently defined by

$$p(x \,|\, \omega_1) = p(x \,|\, \omega_2)$$

Thus

$$\frac{1}{\sigma_1\sqrt{2\pi}} \exp\left\{-\frac{1}{2\sigma_1^2}(x-\mu_1)^2\right\} = \frac{1}{\sigma_2\sqrt{2\pi}} \exp\left\{-\frac{1}{2\sigma_2^2}(x-\mu_2)^2\right\}$$

$$\implies \exp\left\{-\frac{1}{2}x^2\right\} = 2\exp\left\{-2(x-\frac{3}{2})^2\right\}$$

$$\implies -\frac{1}{2}x^2 = \log 2 - 2(x-\frac{3}{2})^2 = \log 2 - 2x^2 + 6x - \frac{9}{2}$$

$$\implies \frac{3}{2}x^2 - 6x + \frac{9}{2} - \log 2 = 0$$

$$\implies 3x^2 - 12x + 9 - 2\log 2 = 0$$

This quadratic expression is easily solved and

$$x = \frac{12 \pm \sqrt{144 - 12(-2\log 2)}}{6} = .79, 3.2092$$

Therefore the Bayes' classifier for this problem is

$$\text{Class}\,\{\mathbf{x}\} = \begin{cases} \omega_1 & \text{if } \; x \leq .79 \text{ or } x \geq 3.2092 \\ \omega_2 & \text{if } \; .79 < x < 3.2092 \end{cases}$$

iv)

$$P(\text{error}) = \frac{1}{2}\left(\int_{x=-\infty}^{.79} p(x \,|\, \omega_2)\, dx + \int_{x=.79}^{3.2092} p(x \,|\, \omega_1)\, dx + \int_{x=3.2092}^{\infty} p(x \,|\, \omega_2)\, dx\right)$$

v) It is the classifier which has minimal $P(\text{error})$.

Exercises 5: *Bayes' Risk I*

Consider the following 2 class classification problem. The likelihood functions for each class is a Gaussian:

$$P(x|\omega_i) = \frac{1}{\sigma_i\sqrt{2\pi}} \exp\left(-\frac{(x-\mu_i)^2}{2\sigma_i^2}\right)$$

with $\mu_1 = 0, \sigma_1^2 = 5$ and $\mu_2 = 3, \sigma_2^2 = 1$. The priors for each class are $P(\omega_1) = P(\omega_2) = .5$. Define the (mis)classification costs as $C_{11} = C_{22} = 0$, $C_{12} = 1, C_{21} = \sqrt{5}$.

Determine a decision rule minimizing the probability of error.

Solution:

Remember: C_{ij} denotes the cost of choosing class ω_i when ω_j is the true class.

The likelihood ratio is calculated as

$$\Lambda(x) = \frac{p(x|\omega_1)}{p(x|\omega_2)} = \frac{\frac{1}{\sqrt{5}\sqrt{2\pi}} \exp\left(-\frac{x^2}{10}\right)}{\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-3)^2}{2}\right)} = \frac{\exp\left(-\frac{x^2}{10}\right)}{\sqrt{5}\exp\left(-\frac{(x-3)^2}{2}\right)}$$

To minimize the Bayes' Risk choose class ω_1 if

$$\Lambda(x) > \frac{(C_{12} - C_{22})P(\omega_2)}{(C_{21} - C_{11})P(\omega_1)} = \frac{1}{\sqrt{5}}$$

$$\begin{aligned}
&\implies \frac{\exp\left(-\frac{x^2}{10}\right)}{\exp\left(-\frac{(x-3)^2}{2}\right)} > 1 \\
&\implies \exp\left(-\frac{x^2}{10}\right) > \exp\left(-\frac{(x-3)^2}{2}\right) \\
&\implies \frac{x^2}{10} < \frac{(x-3)^2}{2} \\
&\implies x^2 < 5(x-3)^2 \\
&\implies x^2 < 5x^2 - 30x + 45 \\
&\implies 0 < 4x^2 - 30x + 45 \\
&\implies x > \frac{1}{4}(15 + 3\sqrt{5}) \text{ and } x < \frac{1}{4}(15 - 3\sqrt{5})
\end{aligned}$$

Exercises 6: *k Nearest Neighbour classifier*

Remember the distance metric used in a nearest neighbour classifier affects the performance of the classifier. A commonly used distance metric family is the L_p norm where

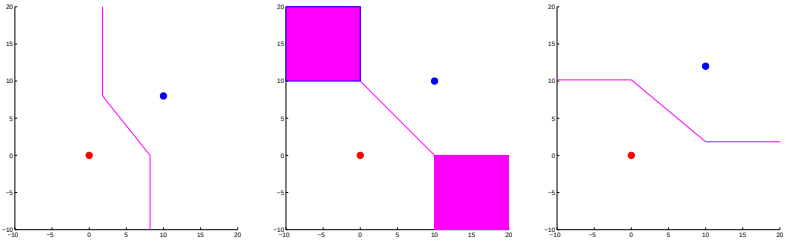
$$\|\mathbf{x}\|_p = \left(\sum_{i=1}^{d} |x_i|^p \right)^{\frac{1}{p}}$$

Consider the case of using a kNN classifier, but with the L_1 norm to measure distances rather than the L_2 (Euclidean) norm. Draw (in two dimensions) a simple case of a binary classification problem for which the L_1 classifier would return a different class for a test point than an L_2 classifier. In particular, draw ≥ 1 training points (one for each class) and a test point that would be classified differently according to the two distance metrics.

What properties of a data set do you imagine would influence whether the L_1 distance would work better or worse than the L_2 distance?

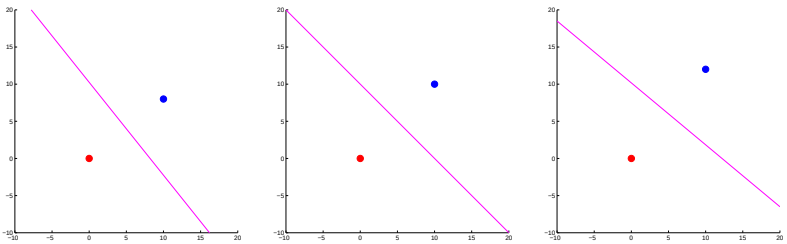
Solution:

In this example there is one point from each class. Below is shown the decision boundary created by the 1-nn classifier using the L_1 norm. The magenta colour signifies when area which are equi-distant from the two points. Note three different decision boundaries are formed based by moving the blue point around. Let (x, y) be the coordinates of the blue point then:



(a) $x > y$ (b) $x = y$ (c) $x < y$

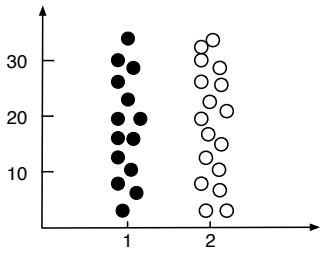
Of course the L_2 norm decision boundaries formed for these different cases are:



(a) $x > y$ (b) $x = y$ (c) $x < y$

Exercises 7: *Nearest neighbour classification*

Carefully examine the data from two classes shown in the figure below.



Answer the following questions about this example.

i) Can you apply a kNN (say with $k = 5$) classifier on this data using a Euclidean distance metric and hope to obtain a sensible decision boundary? Explain your answer.

ii) How must the data be processed before a kNN will produce an accurate decision boundary ?

Solution:

i) No. The range of values in the second dimension are an order of magnitude larger than those in the first. Thus the Euclidean distance computed between any two points will be dominated from the contribution from the second dimension and for this example all the discriminatory information is contained in the first dimension.

ii) The data should be scaled in both dimensions such that the range of values range between 0 and 1.

Exercises 8: *Nearest neighbour classification*

The bias of a classifier at a point $\mathbf{x}$ measures the amount by which the average of our estimate differs from the true class label:

$$\text{Bias} = \text{E}\left[L(y, \text{E}\left[\hat{f}(x)\right])^2\right]$$

while the variance of the classifier is the expected squared deviation of $\text{E}\left[\hat{f}(x)\right]$ around its mean

$$\text{Variance} = \text{E}\left[(\hat{f}(x) - \text{E}\left[\hat{f}(x)\right])^2\right]$$

Say we use the $0, 1$ loss function and a kNN nearest classifier so that

$$\hat{f}(\mathbf{x}) = \text{sgn}\left[\sum_{\mathbf{x}_i \text{ a neighbor of } \mathbf{x}} y_i\right]$$

what effect will the size of k have on the bias and variance of our classifier?

Solution:

For small k, the estimate $\hat{f}(\mathbf{x})$ can potentially adapt itself better to the underlying $f(\mathbf{x})$. Therefore, the bias will be small. On the other hand the variance can be large.

As we increase k, the bias - the squared difference between $f(\mathbf{x})$ and the average of $f(\mathbf{x})$ at the k-nearest neighbours - will typically increase, while the variance decreases.

Exercises 9: *Discriminant Functions I*

Let $p(\mathbf{x}|\omega_i) \sim N(\boldsymbol{\mu}_i, \Sigma)$ for $i = 1, 2$ in a two-class d-dimensional problem with the same covariances but arbitrary means and prior probabilities.

a) Show that the decision boundary between the two classes is a hyper-plane.

b) Need this decision boundary be perpendicular to the line connecting the two means $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$.

c) In terms of the prior probabilities for the two classes $P(\omega_1)$ and $P(\omega_2)$ state the condition that the Bayes decision does not pass between the two means.

Solution:

Assume $\mathbf{x} \in \mathbb{R}^d$ then for $i = 1, 2$:

$$p(\mathbf{x}|\omega_i) = \frac{1}{(2\pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_i)^T\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_i)\right)$$

From Bayes' Rule know that for $i = 1, 2$

$$p(\omega_i|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_i)p(\omega_i)}{p(\mathbf{x})}$$

a) The decision boundary is defined by

$$\begin{aligned}
& p(\omega_1|\mathbf{x}) = p(\omega_2|\mathbf{x}) \\
\implies & p(\mathbf{x}|\omega_1)p(\omega_1) = p(\mathbf{x}|\omega_2)p(\omega_2) \\
\implies & \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_1)^T\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_1)\right)p(\omega_1) = \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_2)^T\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_2)\right)p(\omega_2) \\
\implies & -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_1)^T\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_1) + \log p(\omega_1) = -\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_2)^T\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}_2) + \log p(\omega_2) \\
\implies & \boldsymbol{\mu}_1^T\Sigma^{-1}\mathbf{x} - \frac{1}{2}\boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1 + \log p(\omega_1) = \boldsymbol{\mu}_2^T\Sigma^{-1}\mathbf{x} - \frac{1}{2}\boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2 + \log p(\omega_2) \\
\implies & (\boldsymbol{\mu}_1^T - \boldsymbol{\mu}_2^T)\Sigma^{-1}\mathbf{x} - \frac{1}{2}\left(\boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2\right) + \log\frac{p(\omega_1)}{p(\omega_2)} = 0
\end{aligned}$$

Thus the decision boundary is defined by the hyper-plane:

$$\mathbf{w}^T\mathbf{x} + w_0 = 0$$

where

$$\mathbf{w} = \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \quad w_0 = -\frac{1}{2}\left(\boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2\right) + \log\frac{p(\omega_1)}{p(\omega_2)}$$

b) In general the answer is no. However, if

i) Σ is equal to $\sigma^2 I$ then the statement would be true.

ii) $(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$ is an eigen-vector of Σ then the decision boundary is orthogonal to line joining $(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$.

c) For the decision hyperplane not to intersect the line joining the two means between the two means implies that either $p(\omega_1)/p(\omega_2)$ is large or small. To get an exact value, we do the following calculations. For the condition to hold either:

$$\mathbf{w}^T\boldsymbol{\mu}_1 + w_0 > 0 \quad \textbf{and} \quad \mathbf{w}^T\boldsymbol{\mu}_2 + w_0 > 0$$

or

$$\mathbf{w}^T\boldsymbol{\mu}_1 + w_0 < 0 \quad \textbf{and} \quad \mathbf{w}^T\boldsymbol{\mu}_2 + w_0 < 0$$

Let's look at the first condition.

$$\begin{aligned}
\mathbf{w}^T\boldsymbol{\mu}_1 + w_0 > 0 &\implies \log\frac{p(\omega_1)}{p(\omega_2)} > -(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\Sigma^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\left(\boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2\right) \\
&\implies \frac{p(\omega_1)}{p(\omega_2)} > \exp\left(-(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\Sigma^{-1}\boldsymbol{\mu}_1 + \frac{1}{2}\left(\boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2\right)\right) \\
&\implies \frac{p(\omega_1)}{p(\omega_2)} > \exp\left(-A_1\right)
\end{aligned}$$

where

$$A_1 = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\Sigma^{-1}\boldsymbol{\mu}_1 - \frac{1}{2}\left(\boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2\right)$$

Similarly

$$\begin{aligned}
\mathbf{w}^T\boldsymbol{\mu}_2 + w_0 > 0 &\implies \frac{p(\omega_1)}{p(\omega_2)} > \exp\left(-(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T\Sigma^{-1}\boldsymbol{\mu}_2 + \frac{1}{2}\left(\boldsymbol{\mu}_1^T\Sigma^{-1}\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2^T\Sigma^{-1}\boldsymbol{\mu}_2\right)\right) \\
&\implies \frac{p(\omega_1)}{p(\omega_2)} > \exp\left(-A_2\right)
\end{aligned}$$

Therefore for the decision hyper-plane not to pass between the two means the following must hold:

$$\frac{p(\omega_1)}{p(\omega_2)} > \max\left\{\exp\left(-A_1\right), \exp\left(-A_2\right)\right\}$$

The other set of conditions yields:

$$\frac{p(\omega_1)}{p(\omega_2)} < \min\left\{\exp\left(-A_1\right), \exp\left(-A_2\right)\right\}$$

Exercises 10: *Convergence of the linear perceptron learning rule*

Suppose we have n linearly separable points $\mathbf{x}_i$ in $\mathcal{R}^p$ in general position, with class labels $y_i \in \{-1, 1\}$. Prove that the perceptron learning algorithm converges to a separating hyperplane in a finite number of steps by proving these sub-problems:

1. Denote a hyperplane by $f(\mathbf{x}) = \mathbf{w}_1^t \mathbf{x} + w_0 = 0$, or in more compact notation $\mathbf{w}^t \mathbf{x}^* = 0$, where $\mathbf{x}^* = (\mathbf{x}, 1)$ and $\mathbf{w} = (\mathbf{w}_1, w_0)$. Let $\mathbf{z}_i = \mathbf{x}_i^* / \|\mathbf{x}_i^*\|$. Show that separability implies the existence of a $\mathbf{w}_{\text{sep}}$ such that $y_i \, \mathbf{w}_{\text{sep}}^T \, \mathbf{z}_i \geq 1 \;\; \forall i$

2. Given a current $\mathbf{w}_{\text{old}}$, the perceptron algorithm identifies a point $\mathbf{z}_i$ that is misclassified, and produces the update $\mathbf{w}_{\text{new}} \leftarrow \mathbf{w}_{\text{old}} + y_i \, \mathbf{z}_i$. Show that $\|\mathbf{w}_{\text{new}} - \mathbf{w}_{\text{sep}}\|^2 \leq \|\mathbf{w}_{\text{old}} - \mathbf{w}_{\text{sep}}\|^2 - 1$, and hence that the algorithm converges to a separating hyperplane in no more than $\|\mathbf{w}_{\text{start}} - \mathbf{w}_{\text{sep}}\|^2$ steps.

Solution:

1) Let $\mathbf{w}_S$ be a hyperplane which separates the data then

$$y_i \, \mathbf{w}_S^t \, \mathbf{x}_i^* > 0 \quad \text{for } i = 1, \ldots, n$$

Let $\epsilon > 0$ be defined such that

$$y_i \, \mathbf{w}_S^t \, \mathbf{x}_i^* \geq \epsilon \quad \text{for } i = 1, \ldots, n$$

and also note that as $\mathbf{x}^* = (\mathbf{x}, 1) \implies \|\mathbf{x}^*\| \geq 1$. Define $\gamma \geq 1$ such that

$$\|\mathbf{x}_i^*\| \leq \gamma \quad \text{for } i = 1, \ldots, n$$

Let $\mathbf{w}_{\text{sep}} = \frac{\gamma \, \mathbf{w}_S}{\epsilon}$ and then for $i = 1, \ldots, n$

$$\begin{aligned} y_i \, \mathbf{w}_{\text{sep}}^t \, \mathbf{z}_i &= y_i \, \frac{\gamma \, \mathbf{w}_S^t}{\epsilon} \, \frac{\mathbf{x}_i^*}{\|\mathbf{x}_i^*\|} \\ &= \frac{\gamma}{\|\mathbf{x}_i^*\| \epsilon} (y_i \, \mathbf{w}_S^t \, \mathbf{x}_i^*) \\ &\geq \frac{\gamma}{\|\mathbf{x}_i^*\| \epsilon} \, \epsilon, \quad \text{as } y_i \, \mathbf{w}_S^t \, \mathbf{x}_i^* \geq \epsilon \\ &= \frac{\gamma}{\|\mathbf{x}_i^*\|} \geq 1, \quad \text{as } \gamma \geq \|\mathbf{x}_i^*\| \; \forall i \end{aligned}$$

Thus we have shown that

$$y_i \, \mathbf{w}_{\text{sep}}^t \, \mathbf{z}_i \geq 1 \quad \text{for } i = 1, \ldots, n$$

2) Here we prove the first part of the question - each iteration of the linear perceptron learning algorithm makes the estimate closer to a separating hyperplane:

$$\begin{aligned}
\|\mathbf{w}_{\text{new}} - \mathbf{w}_{\text{sep}}\|^2 &= (\mathbf{w}_{\text{new}} - \mathbf{w}_{\text{sep}})^t(\mathbf{w}_{\text{new}} - \mathbf{w}_{\text{sep}}) \\
&= (\mathbf{w}_{\text{old}} + y_i\mathbf{z}_i - \mathbf{w}_{\text{sep}})^t(\mathbf{w}_{\text{old}} + y_i\mathbf{z}_i - \mathbf{w}_{\text{sep}}), \quad \text{as } \mathbf{w}_{\text{new}} = \mathbf{w}_{\text{old}} + y_i\,\mathbf{z}_i \\
&= (\mathbf{w}_{\text{old}} - \mathbf{w}_{\text{sep}})^t(\mathbf{w}_{\text{old}} - \mathbf{w}_{\text{sep}}) + 2\,y_i(\mathbf{w}_{\text{old}} - \mathbf{w}_{\text{sep}})^t\,\mathbf{z}_i + y_i^2\,\mathbf{z}_i^t\,\mathbf{z}_i \\
&= \|\mathbf{w}_{\text{old}} - \mathbf{w}_{\text{sep}}\|^2 + 2\,\underbrace{(\underbrace{y_i\,\mathbf{w}_{\text{old}}^t\,\mathbf{z}_i}_{\leq 0} - \underbrace{y_i\,\mathbf{w}_{\text{sep}}^t\,\mathbf{z}_i}_{\geq 1})}_{\leq -1} + 1, \quad \text{as } y_i^2 = 1 \text{ and } \mathbf{z}_i^t\,\mathbf{z}_i = 1 \\
&\leq \|\mathbf{w}_{\text{old}} - \mathbf{w}_{\text{sep}}\|^2 + 2(-1) + 1, \quad \text{as shown earlier } y_i\,\mathbf{w}_{\text{sep}}^t\,\mathbf{z}_i \geq 1 \\
&\qquad\qquad \text{and } y_i\,\mathbf{w}_{\text{old}}^t\,\mathbf{z}_i \leq 0 \text{ as point } \mathbf{z}_i \text{ was misclassified} \\
&= \|\mathbf{w}_{\text{old}} - \mathbf{w}_{\text{sep}}\|^2 - 1
\end{aligned}$$

In summary

$$\|\mathbf{w}_{\text{new}} - \mathbf{w}_{\text{sep}}\|^2 \leq \|\mathbf{w}_{\text{old}} - \mathbf{w}_{\text{sep}}\|^2 - 1$$

Let $\mathbf{w}^{(t)}$ be the tth estimate of the seperating hyperplane and $\mathbf{w}^{(0)}$ the initial estimate of this hyperplane. Then using the result just proved:

$$\begin{aligned}
\|\mathbf{w}^{(t)} - \mathbf{w}_{\text{sep}}\|^2 &\leq \|\mathbf{w}^{(t-1)} - \mathbf{w}_{\text{sep}}\|^2 - 1 \\
&\leq \|\mathbf{w}^{(t-2)} - \mathbf{w}_{\text{sep}}\|^2 - 1 - 1 \\
&\vdots \\
&\leq \|\mathbf{w}^{(0)} - \mathbf{w}_{\text{sep}}\|^2 - t
\end{aligned}$$

Now let N be the smallest integer such that $\|\mathbf{w}^{(0)} - \mathbf{w}_{\text{sep}}\|^2 \geq N$ then

$$\|\mathbf{w}^{(t)} - \mathbf{w}_{\text{sep}}\|^2 \leq \|\mathbf{w}^{(0)} - \mathbf{w}_{\text{sep}}\|^2 - t \leq N - t$$

As $\|\mathbf{w}^{(t)} - \mathbf{w}_{\text{sep}}\|^2 \geq 0$ then either $t \leq N$ and algorithm has not converged or else $t > N$ and the algorithm has converged.

From this we conclude that the linear perceptron algorithm must converge in a finite number of steps and the maximum number of steps is defined by $\|\mathbf{w}^{(0)} - \mathbf{w}_{\text{sep}}\|^2$.

Exercises 11: *Cross Validation*

For k-fold cross-validation what are disdavantages and advantages for small and large k values? Why? Let N be the number of training examples.

Solution:

What value should we choose for k? With $k = N$, the cross-validation estimator is approximately unbiased for the true (expected) prediction error, but can have high variance because the N "training sets" are so similar to one another. The computational burden is also considerable, requiring N applications of the learning method.

While on the other hand with $k = 5$ cross-validation has lower variance. But bias could be a problem, depending on how the performance of the learning method varies with the size of the training set.

Overall, five- or tenfold cross-validation are recommended as a good compromise by the experts!

Remeber bias measures the amount by which the average of our estimate differs from the true mean while the variance is squared deviation of the predictor around its mean.

Exercises 12: *LDA*

Assume we have a two class problem. The feature vectors extracted from each class are two dimensional and the class conditional densities are:

$$p(\mathbf{x}|\omega_1) \sim N(\boldsymbol{\mu}_1, \Sigma) \quad \text{and} \quad p(\mathbf{x}|\omega_2) \sim N(\boldsymbol{\mu}_2, \Sigma)$$

where

$$\boldsymbol{\mu}_1 = (1,1)^T, \quad \boldsymbol{\mu}_2 = (3,2)^T \quad \text{and} \quad \Sigma = \begin{pmatrix} \sigma^2 & 0 \\ 0 & 0.2^2 \end{pmatrix}$$

In LDA you project each feature vector generated by these class conditionals onto a line via $\mathbf{w}^T\mathbf{x}$ to obtain a scalar value.

For $\sigma = 1$ sketch the class conditional densities and indicate why in this case it is better to project feature vectors from these two classes onto the y-axis as opposed to the x-axis for performing discrimination on the resulting scalar values.

For what values of σ will it be better to project onto the x-axis ?

For what values of σ will the y-axis tend towards the optimal projection line with respect to the Fisher criterion?

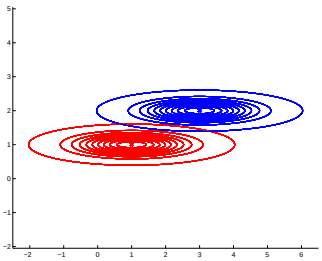
Solution:

The mean of points projected from the two class is defined as

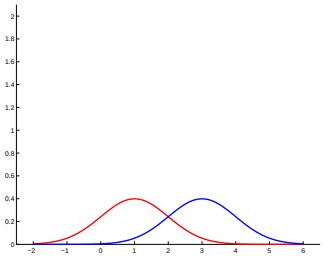
$$\tilde{\mu}_1 = \mathbf{w}^T\boldsymbol{\mu}_1 = \begin{cases} 1 & \text{if } \mathbf{w}^T = (1,0) \\ 1 & \text{if } \mathbf{w}^T = (0,1) \end{cases}, \quad \tilde{\mu}_2 = \mathbf{w}^T\boldsymbol{\mu}_2 = \begin{cases} 3 & \text{if } \mathbf{w}^T = (1,0) \\ 2 & \text{if } \mathbf{w}^T = (0,1) \end{cases}$$

The variance of points projected from the two class is defined as

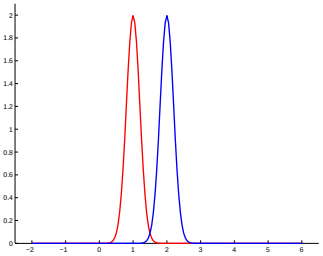
$$\sigma_1^2 = \sigma_2^2 = \mathbf{w}^T\Sigma\mathbf{w} = \begin{cases} \sigma^2 & \text{if } \mathbf{w}^T = (1,0) \\ .2^2 & \text{if } \mathbf{w}^T = (0,1) \end{cases}$$



class conditionals when $\sigma = 1$



distributions when projected onto x-axis



distributions when projected onto y-axis

Obviously projecting onto the y-axis induces a much smaller amount of confusion between the two classes then projecting onto the x-axis. Thus to perform discrimination it would be better to project onto the y-axis.

For what values of σ will it be better to project onto the x-axis ?

We will use the **Fisher score** to decide which projection is better. Remember this score is proportional to

$$\frac{(\tilde{\mu_1} - \tilde{\mu_2})^2}{\sigma_1^2 + \sigma_2^2} = \begin{cases} \frac{4}{2\sigma^2} = \frac{2}{\sigma^2} & \text{if } \mathbf{w}^T = (1,0) \\ \frac{1}{2\times.2^2} = \frac{25}{2} & \text{if } \mathbf{w}^T = (0,1) \end{cases}$$

If we want the **Fisher score** to be larger for projecting onto the x-axis then

$$\frac{2}{\sigma^2} > \frac{25}{2} \quad \textbf{when} \quad \boxed{\sigma < \frac{2}{5}}$$

For what values of σ will the y-axis tend towards the optimal projection line with respect to the Fisher criterion?

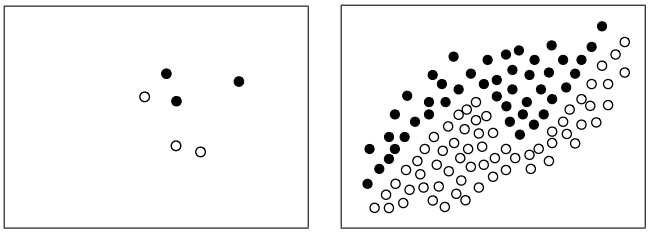
The optimal projection line according to the Fisher criterion is

$$\mathbf{w}^* \propto (\Sigma + \Sigma)^{-1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_2) = \frac{1}{2} \times \begin{pmatrix} \frac{1}{\sigma^2} & 0 \\ 0 & 25 \end{pmatrix} \begin{pmatrix} 2 \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{\sigma^2} \\ \frac{25}{2} \end{pmatrix}$$

The optimal projection w^* tends to $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ when $\sigma \to \infty$. Thus for large values of σ you should project onto the y-axis.

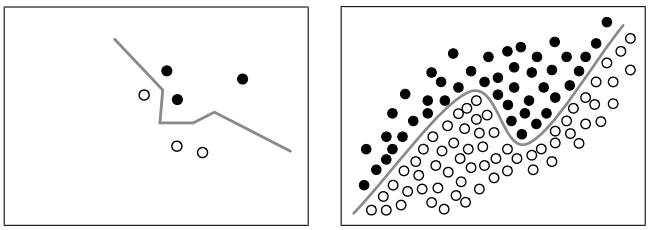
Exercises 13:

Draw the decision boundary formed by a 1-nearest neighbour classifier in the two different figures below. Not that the sparse set of points is a subset of the dense ones. What lesson should be learned from these examples?



(**a**) (**b**)

Solution:



(**a**) (**b**)

Drawing conclusions and learning decision boundaries from a small amount of training data is, generally, not a very good idea! Also it should be noted that one cannot usually extrapolate information far away from the training data.

Exercises 14: *Naïve Bayes and Logistic regression**

Assume we have a two class problem. The feature vector $\mathbf{x} = (x_1, \ldots, x_d)$ extracted from each class is d dimensional and each $x_i \in \{0, 1\}$. Let

$$p(x_i = 1 \,|\, \omega = 1) = \theta_{i1} \quad p(x_i = 0 \,|\, \omega = 1) = 1 - \theta_{i1}$$
$$p(x_i = 1 \,|\, \omega = 0) = \theta_{i0} \quad p(x_i = 0 \,|\, \omega = 0) = 1 - \theta_{i0}$$

Show that

i) The above likelihoods can be written as $p(x_i \,|\, \omega = j) = \theta_{ij}^{x_i} \, (1 - \theta_{ij})^{1-x_i}$ for $j = 0, 1$.

ii) Assuming a independence between the features write down the expression for $p(\mathbf{x} \,|\, \omega = 0)$.

iii) If $P(\omega = 0) = p_0$ write down an expression for $p(\omega = 0 \,|\, \mathbf{x})$ (this corresponds to the Naive Bayes' model).

iv) Show how $p(\omega = 0 \,|\, \mathbf{x})$ can be written in the form

$$p(\omega = 0 \,|\, \mathbf{x}) = \frac{1}{1 + \exp\left(w_0 + \mathbf{w}^T \, \mathbf{x}\right)}$$

v) Then what are the expressions for $p(\omega = 0 \,|\, \mathbf{x})$ and

$$\log \frac{p(\omega = 1 \,|\, \mathbf{x})}{p(\omega = 0 \,|\, \mathbf{x})}$$

This is the same form as which discriminative model?

Solution:

i) Want to show

$$p(x_i \,|\, \omega = j) = \theta_{ij}^{x_i} \, (1 - \theta_{ij})^{1-x_i} \quad \text{for } j = 0, 1$$

If $j = 0$ then

$$p(x_i{=}0 \,|\, \omega{=}0){=}\theta_{i0}^{0} \, (1{-}\theta_{i0})^{1-0}{=}1{-}\theta_{i0} \quad \textbf{and} \quad p(x_i{=}1 \,|\, \omega{=}0){=}\theta_{i0}^{1} \, (1{-}\theta_{i0})^{1-1}{=}\theta_{i0}$$

and if $j = 1$ then

$$p(x_i{=}0 \,|\, \omega{=}1){=}\theta_{i1}^{0} \, (1{-}\theta_{i1})^{1-0}{=}1{-}\theta_{i1} \quad \textbf{and} \quad p(x_i{=}1 \,|\, \omega{=}1){=}\theta_{i1}^{1} \, (1{-}\theta_{i1})^{1-1}{=}\theta_{i1}$$

Both of these are indeed equal to our original definition of $p(x_i = 0 \,|\, \omega = 0)$ etc..

ii)

$$p(\mathbf{x}\,|\,\omega=0)=\prod_{i=1}^{d}p(x_i\,|\,\omega=0)=\prod_{i=1}^{d}\theta_{i0}^{x_i}\,(1-\theta_{i0})^{1-x_i}$$

iii)

$$\begin{aligned}p(\omega=0\,|\,\mathbf{x}) &= \frac{p(\mathbf{x}\,|\,\omega=0)\,P(\omega=0)}{p(\mathbf{x}\,|\,\omega=0)\,P(\omega=0)+p(\mathbf{x}\,|\,\omega=1)\,P(\omega=1)}\\ &= \frac{p_0\,\prod_{i=1}^{d}\,\theta_{i0}^{x_i}\,(1-\theta_{i0})^{1-x_i}}{p_0\,\prod_{i=1}^{d}\,\theta_{i0}^{x_i}\,(1-\theta_{i0})^{1-x_i}+(1-p_0)\,\prod_{i=1}^{d}\,\theta_{i1}^{x_i}\,(1-\theta_{i1})^{1-x_i}}\end{aligned}$$

iv)

$$\begin{aligned}p(\omega=0\,|\,\mathbf{x}) &= \frac{p_0\,\prod_{i=1}^{d}\,\theta_{i0}^{x_i}\,(1-\theta_{i0})^{1-x_i}}{p_0\,\prod_{i=1}^{d}\,\theta_{i0}^{x_i}\,(1-\theta_{i0})^{1-x_i}+(1-p_0)\,\prod_{i=1}^{d}\,\theta_{i1}^{x_i}\,(1-\theta_{i1})^{1-x_i}}\\ &= \frac{1}{1+\frac{(1-p_0)\,\prod_{i=1}^{d}\,\theta_{i1}^{x_i}\,(1-\theta_{i1})^{1-x_i}}{p_0\,\prod_{i=1}^{d}\,\theta_{i0}^{x_i}\,(1-\theta_{i0})^{1-x_i}}}\\ &= \frac{1}{1+\exp\left(\log\left\{\frac{(1-p_0)\,\prod_{i=1}^{d}\,\theta_{i1}^{x_i}\,(1-\theta_{i1})^{1-x_i}}{p_0\,\prod_{i=1}^{d}\,\theta_{i0}^{x_i}\,(1-\theta_{i0})^{1-x_i}}\right\}\right)}\\ &= \frac{1}{1+\exp(\sum x_i[\log\theta_{i1}-\log(1-\theta_{i1})-\log\theta_{i0}+\log(1-\theta_{i0})]+\log(1-p_0)-\log(p_0)+\sum[\log(1-\theta_{i1})-\log(1-\theta_{i0})])}\\ &= \frac{1}{1+\exp(\,\mathbf{w}^T\mathbf{x}+w_0)}\end{aligned}$$

where

$$w_0=\log(1-p_0)-\log(p_0)+\sum\left[\log(1-\theta_{i1})-\log(1-\theta_{i0})\right]$$

$$w_i=\log\theta_{i1}-\log(1-\theta_{i1})-\log\theta_{i0}+\log(1-\theta_{i0})\quad\text{for } i=1,2,\ldots,d.$$

v) $p(\omega=1\,|\,\mathbf{x})$ can be expressed as:

$$p(\omega=1\,|\,\mathbf{x})=1-p(\omega=0\,|\,\mathbf{x})=1-\frac{1}{1+\exp(\,\mathbf{w}^T\mathbf{x}+w_0)}=\frac{\exp\left(\,\mathbf{w}^T\mathbf{x}+w_0\right)}{1+\exp(\,\mathbf{w}^T\mathbf{x}+w_0)}$$

Therefore

$$\boxed{\log\frac{p(\omega=1\,|\,\mathbf{x})}{p(\omega=0\,|\,\mathbf{x})}}=\log\left(\frac{\exp\left(\,\mathbf{w}^T\mathbf{x}+w_0\right)}{1}\right)=\boxed{\mathbf{w}^T\mathbf{x}+w_0}$$

This is the same form as in logistic regression. Thus if one had lots of training data and no noise then the classifier found via logistic regression and the generative modelling of the question should be exactly the same. However, this is unlikely to the case. In the generative modelling case we have $2d + 2$ parameters to estimate from the training data while in the logistic regression case there are only $d + 1$ parameters. Perhaps then if training data is sparse and one is only interested in the separating hyperplane, it would be more robust to estimate the separating hyperplane directly as opposed to trying to estimate the underlying generative model of the two classes.

Exercises 15: *Fisher's Linear Discriminant and MSE**

The least squares approach to the determination of a linear discriminant was based on the goal of making the model predictions as close as possible to a set of target values. By constrast, the Fisher creterion is derived by requiring maximum class separation in the output space in conjuction with minimum within class spread. For the two-class problem the Fisher criterion can be seen as a special case of least squares.

Take the targets for class ω_1 and to be $\frac{n}{n_1}$ where n_1 is the number of patterns from class ω_1 and n is the total number of patterns. For class ω_2 take the targets to be $-\frac{n}{n_2}$ where n_2 is the number of patterns from class ω_2.

The sum-of-squares error function is written as

$$J = \frac{1}{2}\sum_{i=1}^{n}(\mathbf{w}^T\mathbf{x}_i + w_0 - t_i)^2$$

where each $t_i = \frac{n}{n_1}$ or $-\frac{n}{n_2}$ depending if $\mathbf{x}_i$ belongs to class ω_1 or ω_2. Show that J is minimized when

$$\mathbf{w} \propto S_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

where

$$\mathbf{m}_i = \frac{1}{n_i}\sum_{\mathbf{x}\in\omega_i}\mathbf{x} \quad \text{and} \quad S_\mathbf{w} = \sum_{\mathbf{x}\in\omega_1}(\mathbf{x}-\mathbf{m}_1)(\mathbf{x}-\mathbf{m}_1)^T + \sum_{\mathbf{x}\in\omega_2}(\mathbf{x}-\mathbf{m}_2)(\mathbf{x}-\mathbf{m}_2)^T$$

Solution:

We want to maximize J with respect to w_0 and $\mathbf{w}$. Therefore we compute the partial derivatives and set to zero:

$$\frac{\partial J}{\partial w_o} = \sum_{i=1}^{n}(\mathbf{w}^T\mathbf{x}_i + w_0 - t_i) = 0 \tag{3}$$

$$\frac{\partial J}{\partial \mathbf{w}} = \sum_{i=1}^{n}(\mathbf{w}^T\mathbf{x}_i + w_0 - t_i)\,\mathbf{x}_i = 0 \tag{4}$$

Expanding equation (**??**) gives

$$n\,w_0 = -\sum_{i=1}^{n}\mathbf{w}^T\mathbf{x}_i + \sum_{i=1}^{n}t_i$$

However, filling in the values assigned to t_i when it is from class ω_1 or ω_2 get

$$\sum_{i=1}^{n}t_i = n_1\frac{n}{n_1} - n_2\frac{n}{n_2} = 0$$

Thus

$$w_0 = -\frac{1}{n}\sum_{i=1}^{n}\mathbf{w}^T\mathbf{x}_i = -\mathbf{w}^T\frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i = -\mathbf{w}^T\mathbf{m}$$

Rearranging equation (**??**) get

$$\sum_{i=1}^{n}(\mathbf{w}^T\mathbf{x}_i)\mathbf{x}_i + w_0\sum\mathbf{x}_i - \sum t_i\mathbf{x}_i = 0 \tag{5}$$

Now

$$\sum_{i=1}^{n}(\mathbf{w}^T\mathbf{x}_i)\,\mathbf{x}_i = \sum_{i=1}^{n}\mathbf{x}_i\,\mathbf{x}_i^T\mathbf{w} = \left(\sum_{\mathbf{x}_i\in\omega_1}\mathbf{x}_i\,\mathbf{x}_i^T + \sum_{\mathbf{x}_i\in\omega_2}\mathbf{x}_i\,\mathbf{x}_i^T\right)\mathbf{w}$$

and

$$\begin{aligned}\sum t_i\mathbf{x}_i = \sum_{\mathbf{x}_i\in\omega_1}t_i\,\mathbf{x}_i + \sum_{\mathbf{x}_i\in\omega_2}t_i\,\mathbf{x}_i &= \frac{n}{n_1}\sum_{\mathbf{x}_i\in\omega_1}\mathbf{x}_i - \frac{n}{n_2}\sum_{\mathbf{x}_i\in\omega_2}\mathbf{x}_i\\ &= \frac{n}{n_1}n_1\,\mathbf{m}_1 - \frac{n}{n_2}n_2\,\mathbf{m}_2\\ &= n\,(\mathbf{m}_1 - \mathbf{m}_2)\end{aligned}$$

and

$$\sum_{i=1}^{n}w_0\,\mathbf{x}_i = -\mathbf{w}^T\,\mathbf{m}\sum_{i=1}^{n}\mathbf{x}_i = -(\sum_{i=1}^{n}\mathbf{x}_i)\,\mathbf{m}^T\,\mathbf{w} = -n\,\mathbf{m}\,\mathbf{m}^T\,\mathbf{w}$$

Now

$$\mathbf{m} = \frac{n_1}{n}\mathbf{m}_1 + \frac{n_2}{n}\mathbf{m}_2$$

Therefore

$$\begin{aligned}n\,\mathbf{m}\,\mathbf{m}^T &= \frac{1}{n}(n_1\mathbf{m}_1 + n_2\mathbf{m}_2)(n_1\mathbf{m}_1 + n_2\mathbf{m}_2)^T\\ &= \frac{1}{n}(n_1^2\mathbf{m}_1\mathbf{m}_1^T + n_2^2\mathbf{m}_2\mathbf{m}_2^T + 2n_1n_2\mathbf{m}_1\mathbf{m}_2^T)\\ &= n_1\mathbf{m}_1\mathbf{m}_1^T - \frac{n_1\,n_2}{n}\mathbf{m}_1\mathbf{m}_1^T + n_2\mathbf{m}_2\mathbf{m}_2^T - \frac{n_1\,n_2}{n}\mathbf{m}_2\mathbf{m}_2^T + 2\frac{n_1\,n_2}{n}\mathbf{m}_1\mathbf{m}_2^T\\ &= n_1\mathbf{m}_1\mathbf{m}_1^T + n_2\mathbf{m}_2\mathbf{m}_2^T - \frac{n_1\,n_2}{n}(\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T\end{aligned}$$

But through some algebra trickery

$$\begin{aligned}\sum_{\mathbf{x}_i \in \omega_1} \mathbf{x}_i \, \mathbf{x}_i^T - n_1 \mathbf{m}_1 \mathbf{m}_1^T &= \sum_{\mathbf{x}_i \in \omega_1} \mathbf{x}_i \, \mathbf{x}_i^T - 2 n_1 \mathbf{m}_1 \mathbf{m}_1^T + n_1 \mathbf{m}_1 \mathbf{m}_1^T \\ &= \sum_{\mathbf{x}_i \in \omega_1} \mathbf{x}_i \, \mathbf{x}_i^T - 2 (\sum_{\mathbf{x}_i \in \omega_1} \mathbf{x}_i) \mathbf{m}_1^T + \sum_{\mathbf{x}_i \in \omega_1} \mathbf{m}_1 \mathbf{m}_1^T \\ &= \sum_{\mathbf{x}_i \in \omega_1} (\mathbf{x}_i \, \mathbf{x}_i^T - 2 \mathbf{x}_i \, \mathbf{m}_1^T + \mathbf{m}_1 \mathbf{m}_1^T) \\ &= \sum_{\mathbf{x}_i \in \omega_1} (\mathbf{x}_i - \mathbf{m}_1) \, (\mathbf{x}_i - \mathbf{m}_1)^T\end{aligned}$$

Similarly

$$\sum_{\mathbf{x}_i \in \omega_2} \mathbf{x}_i \, \mathbf{x}_i^T - n_2 \mathbf{m}_2 \mathbf{m}_2^T = \sum_{\mathbf{x}_i \in \omega_2} (\mathbf{x}_i - \mathbf{m}_2) \, (\mathbf{x}_i - \mathbf{m}_2)^T$$

Putting some of this together

$$\begin{aligned}\sum_{i=1}^{n} (\mathbf{w}^T \mathbf{x}_i) \mathbf{x}_i + w_0 \sum \mathbf{x}_i &= \left(\sum_{i=1}^{n} \mathbf{x}_i \, \mathbf{x}_i^T - n \, \mathbf{m} \, \mathbf{m}^T \right) \mathbf{w} \\ &= \left(S_W + \frac{n_1 n_2}{n} (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T \right) \mathbf{w}\end{aligned}$$

From equation (**??**) and all the enusing algebra get

$$S_W \, \mathbf{w} + \frac{n_1 n_2}{n} (\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T \mathbf{w} = n \, (\mathbf{m}_2 - \mathbf{m}_1)$$

Now notice that

$$(\mathbf{m}_1 - \mathbf{m}_2)(\mathbf{m}_1 - \mathbf{m}_2)^T \, \mathbf{w} \propto (\mathbf{m}_1 - \mathbf{m}_2)$$

Thus

$$\begin{aligned} S_W \, \mathbf{w} &\propto (\mathbf{m}_2 - \mathbf{m}_1) \\ \implies \mathbf{w} &= S_W^{-1} (\mathbf{m}_2 - \mathbf{m}_1)\end{aligned}$$

Have ignored the irrelevant scale factors.

Thus the weight vector corresponds to that found by the Fisher criterion.

Exercises 16: *LDA II*

We have two class-conditional probabilities:

$$p(\mathbf{x}|\omega_1) \propto \exp\left(-.5\,\mathbf{x}^T\Sigma^{-1}\mathbf{x}\right)$$
$$p(\mathbf{x}|\omega_2) \propto \exp\left(-.5\,(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right)$$

where $\mathbf{x} = (x, y)$ is a two dimensional vector and

$$\boldsymbol{\mu} = (1,1)^T \qquad\qquad \Sigma = \begin{pmatrix} 1 & 0 \\ 0 & \sigma^2 \end{pmatrix}$$

Define the function $f(\mathbf{x}) = ax + by$ which is the projection of the point $\mathbf{x}$ onto a line passing through the origin.

1. Compute the values $\mu_i = \mathrm{E}\left[f(\mathbf{x})|\omega_i\right]$ and $\sigma_i^2 = \mathrm{Var}\left[f(\mathbf{x})|\omega_i\right]$ for $i = 1, 2$
2. Find the values of a and b that maximize the statistical Fisher discriminant criterion:
$$J(a,b) = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}$$

Solution

1) Compute the expected value of the projection onto the line for each class:

$$\mathrm{E}\left[f(\mathbf{x})\,|\,\omega_i\right] = \mathrm{E}\left[ax + by\,|\,\omega_i\right] = a\,\mathrm{E}\left[x\,|\,\omega_i\right] + b\,\mathrm{E}\left[y\,|\,\omega_i\right] = \begin{cases} 0 & \text{for class } \omega_1 \\ a+b & \text{for class } \omega_2 \end{cases}$$

Next compute the variance, remember

$$\mathrm{Var}\left[f(\mathbf{x})\,|\,\omega_i\right] = \mathrm{E}\left[f(\mathbf{x})^2\,|\,\omega_i\right] - \left(\mathrm{E}\left[f(\mathbf{x})\,|\,\omega_i\right]\right)^2 \tag{6}$$

First

$$\mathrm{E}\left[f(\mathbf{x})^2\,|\,\omega_i\right] = \mathrm{E}\left[a^2x^2 + b^2y^2 + 2abxy\,|\,\omega_i\right] = a^2\,\mathrm{E}\left[x^2\,|\,\omega_i\right] + b^2\,\mathrm{E}\left[y^2\,|\,\omega_i\right] + 2ab\,\mathrm{E}\left[xy\,|\,\omega_i\right]$$

Using the definition of the variance from equation (**??**) then:

$$\mathrm{Var}\left[x\,|\,\omega_i\right] = \mathrm{E}\left[x^2\,|\,\omega_i\right] - \left(\mathrm{E}\left[x\,|\,\omega_i\right]\right)^2 \quad \text{and} \quad \mathrm{Var}\left[y\,|\,\omega_i\right] = \mathrm{E}\left[y^2\,|\,\omega_i\right] - \left(\mathrm{E}\left[y\,|\,\omega_i\right]\right)^2$$

and this implies that

$$\mathrm{E}\left[x^2\,|\,\omega_i\right] = \begin{cases} 1 & \text{for class } \omega_1 \\ 2 & \text{for class } \omega_2 \end{cases} \qquad \mathrm{E}\left[y^2\,|\,\omega_i\right] = \begin{cases} \sigma^2 & \text{for class } \omega_1 \\ \sigma^2+1 & \text{for class } \omega_2 \end{cases}$$

As Σ is a diagonal matrix then the variables x and y are uncorrelated thus:

$$\mathrm{E}\left[xy \,|\, \omega_i\right] = \mathrm{E}\left[x \,|\, \omega_i\right] \mathrm{E}\left[y \,|\, \omega_i\right] = \begin{cases} 0 & \text{for class } \omega_1 \\ 1 & \text{for class } \omega_2 \end{cases}$$

Therefore by plugging in the appropriate values

$$\mathrm{E}\left[f(\mathbf{x})^2 \,|\, \omega_i\right] = \begin{cases} a^2 + \sigma^2 b^2 & \text{for class } \omega_1 \\ 2a^2 + b^2(\sigma^2 + 1) + 2ab & \text{for class } \omega_2 \end{cases}$$

and plugging in more values

$$\mathrm{Var}\left[f(\mathbf{x}) \,|\, \omega_i\right] = \begin{cases} a^2 + b^2\sigma^2 & \text{for class } \omega_1 \\ a^2 + b^2\sigma^2 & \text{for class } \omega_2 \end{cases}$$

2) Want to maximize $J(a, b)$ with respect to a and b where:

$$J(a, b) = \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2} = \frac{(a+b)^2}{2(a^2 + b^2\sigma^2)}$$

Thus compute the appropriate derivative and set to zero

$$\begin{aligned} \frac{\partial J}{\partial a} &= \frac{2(a+b)}{2(a^2 + b^2\sigma^2)} - \frac{(a+b)^2}{2(a^2 + b^2\sigma^2)^2} \times 2a \\ &= \frac{(a+b)(a^2 + b^2\sigma^2) - a(a+b)^2}{(a^2 + b^2\sigma^2)^2} = 0 \end{aligned}$$

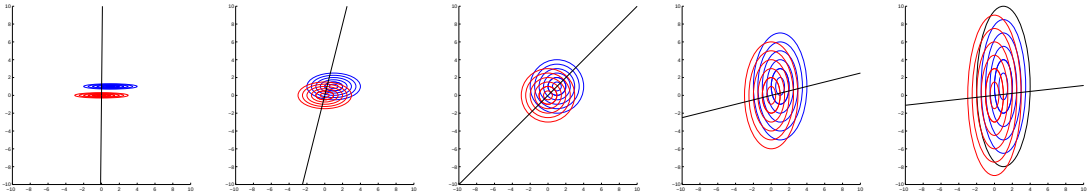
Taking the numerator we get

$$\begin{aligned} & (a+b)(a^2 + b^2\sigma^2) - a(a+b)^2 = 0 \\ \implies & (a^2 + b^2\sigma^2) - a(a+b) = 0 \\ \implies & a = b\sigma^2 \end{aligned}$$

Thus if

$$\begin{aligned} \sigma = 1 &\implies a = b \\ \sigma = 2 &\implies a = 4b \\ \sigma = 10 &\implies a = 100b \end{aligned}$$

The graphs below show the class conditional distributions and projection line found using the maximizing the Fisher criterion.



$\sigma = .1$ $\sigma = .5$ $\sigma = 1$ $\sigma = 2$ $\sigma = 3$

Exercises 17: *Gaussians**

Have the following model for IQ and test scores S

$$IQ \sim \mathcal{N}(100, 15^2), \quad S \,|\, IQ \sim \mathcal{N}(IQ, 10^2)$$

You take the test and get a score of $s_1 = 130$.

i) Derive the posterior density $p(IQ \,|\, s_1 = 130)$

ii) You are a bit disappointed that Bayesians would consider you to have an IQ less then your test score. So you decide to take the test again. Let your score on the second test be S_2. What is the posterior density for $p(S_2 \,|\, s_1 = 130)$?

Solution:

i) Applying the chain rule get

$$\begin{pmatrix} S \\ IQ \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 100 \\ 100 \end{pmatrix}, \begin{pmatrix} 15^2 + 10^2 & 10^2 \\ 10^2 & 10^2 \end{pmatrix} \right) \tag{7}$$

then if reverse the order of the variables get

$$\begin{pmatrix} IQ \\ S \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 100 \\ 100 \end{pmatrix}, \begin{pmatrix} 10^2 & 10^2 \\ 15^2 + 10^2 & 10^2 \end{pmatrix} \right) \tag{8}$$

Apply the conditional of a Gaussian then

$$IQ \,|\, S = 130 \sim \mathcal{N}(\mu, \sigma^2) \tag{9}$$

where

$$\mu = 100 + \frac{10^2}{10^2 + 15^2}(130 - 100) = 109.2308 \tag{10}$$

$$\sigma^2 = 10^2 - \frac{10^2 10^2}{10^2 + 15^2} = 8.3205^2 \tag{11}$$

ii)

$$\begin{aligned}
p(S_2 \,|\, S_1 = 130) &= \int_{q=-\infty}^{\infty} p(S_2,\, IQ = q |\, S_1 = 130)\, dq \\
&= \int_{q=-\infty}^{\infty} p(S_2 \,|\, IQ = q, S_1 = 130)\, p(IQ = q \,|\, S_1 = 130)\, dq \\
&= \int_{q=-\infty}^{\infty} p(S_2 \,|\, IQ = q)\, \mathcal{N}(q;\, \mu, \sigma^2)\, dq \\
&= \int_{q=-\infty}^{\infty} \mathcal{N}(s_2;\, q, 10^2)\, \mathcal{N}(q;\, \mu, \sigma^2)\, dq, \quad \text{after some work} \\
&= \mathcal{N}(s_2;\, \mu, 10^2 + \sigma^2) = \mathcal{N}(s_2;\, 109.23, 13.089^2)
\end{aligned}$$