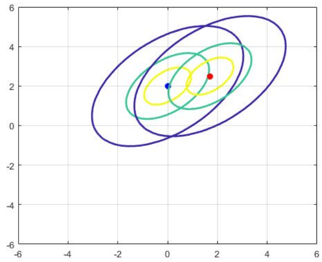
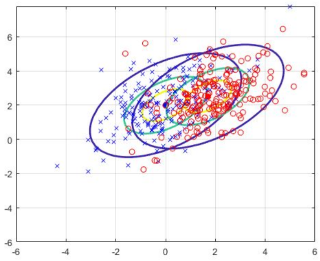
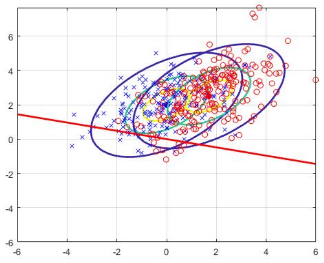
COMP6229(2017/18): Machine Learning Lab 3
Name: Ivanova Iustina
Email: ii1n17@soton.ac.uk

We will compare different methods of classifiers. First of all, we prepare 2 classes that are Gaussian distributed with means $m1 = [0\ 2]^t$ and $m2 = [1.7\ 2.5]^t$ and common covariance matrix $C = [\begin{smallmatrix}2 & 1\\ 1 & 2\end{smallmatrix}]$. We draw 3 contours on the density, then we create 200 samples for 2 different classes on that density. In the first task we programm Fisher Linear Discriminant: the optimal vector to which the points can be projected in a an optimal way, so the classification should the best one. Further we will prove this statement







Img1. Contours of 2 classes Img2. 200 variables of classes   Img3. Fisher LD direction

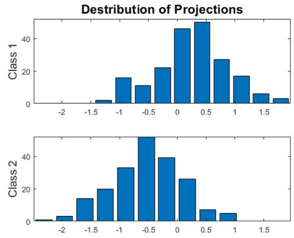
Then we project the data onto Fisher discriminant directions and plot histograms of the distribution of projections: score function for 2 classes $I_F = \frac{|U^T m1 - U^T m2|^2}{U^T C_1 U + U^T C_2 U}$ is the optimal, when 1) $|U^T m1 - U^T m2|^2$ is high ; 2) $U^T C_1 U + U^T C_2 U$ is small. Fisher discriminant finds the U vector, that maximises this function. Define

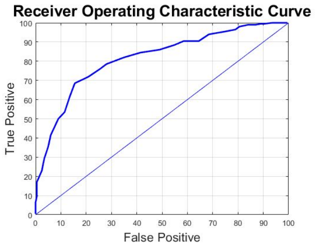
$$S_B = (m1 - m2)(m1 - m2)^T; S_W = C1 + C2; \quad => \quad I_F = \frac{U^T S_B U}{U^T S_W U} \quad => \quad \frac{dI_F}{dU} = \frac{2S_B U(U^T S_W U\ ) - 2S_W U(U^T S_B U\ )}{(U^T S_W U\ )^2} = 0 =>$$

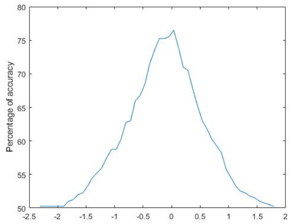
$$S_B U = \frac{(U^T S_B U\ )}{(U^T S_W U\ )} S_W U \quad => define\ \lambda = \frac{(U^T S_B U\ )}{(U^T S_W U\ )} => \lambda U = S_W^{-1} S_B U \quad => \quad U \infty S_W^{-1}(m1 - m2).$$

We see that comparing to Img2 the classes plotted in Img4 can be distinguished more clearer. Then we compute a Receiver Operating Characteristic (ROC) curve to see the effectiveness of clusterization. We draw 2 dimensions: horizontal direction is for points, that were predicted true (True Positive Rate TPR= TP/P = TP/(TP+FN) and vertical direction is for amount of points, that were predicted false (False Positive Rates FPR = FP/P = FP/(FP+TN)) (Img5). The area under this curve is equal to the probability that a classifier will rank a randomly chosen positive instance higher than a randomly chosen negative one (assuming 'positive' ranks higher than 'negative'). Fisher DD works with accuracy 80%. Then we plot the graphic of accuracy for the whole whole interval of projections(Img6). We see that the best prediction is in the cross-section point of 2 classes.

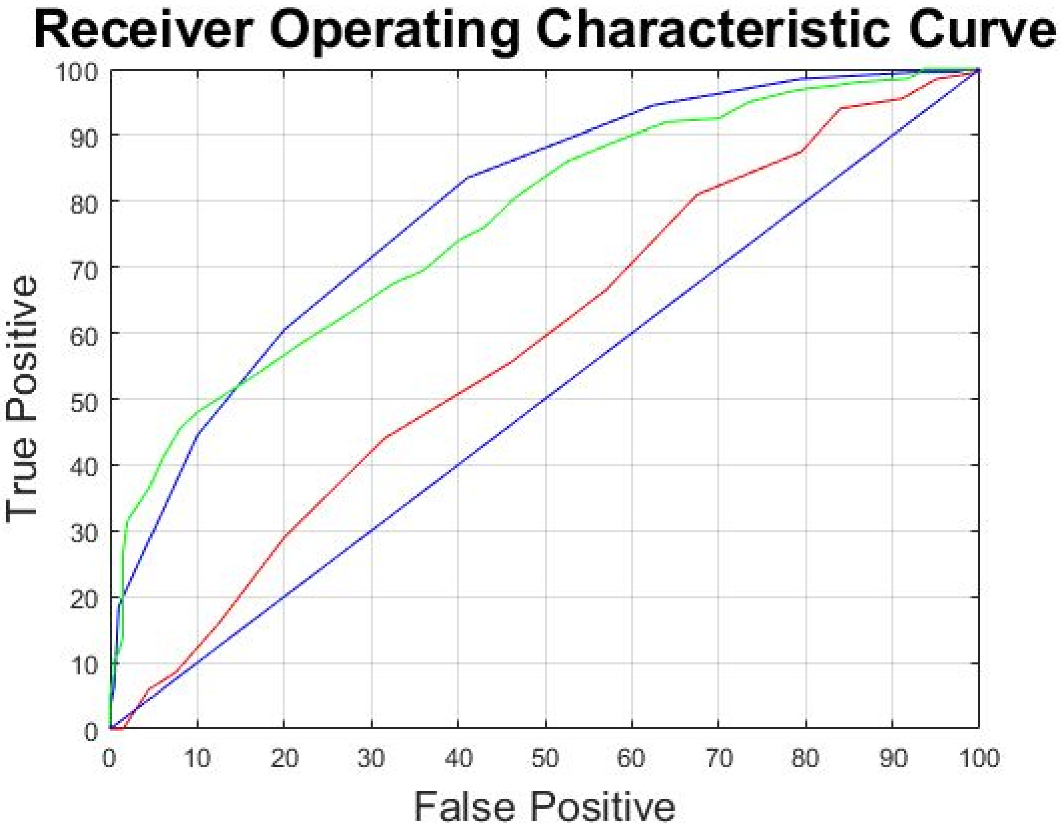
Img4 Distribution of projections   Img5 ROC-curve (Fisher)   Img6 Accuracy (Fisher)







To compare accuracy of Fisher DD we plot the ROC-curve of projections of points on random vector (red line on Img7) and onto direction connecting 2 means of the two classes (green line on Img7). From this picture we see that Fisher predicts better (area under ROC curve for random vector is 58%, for the line connecting 2 means is 78%).



Img7. ROC-curve for different vectors

Then we program a nearest neighbour classifier (1-NN) on this data. So we took the only 1 nearest neighbour for each point, assume that current point has the same class as neighbour, and compare this prediction with the real data. The accuracy is 61%. We see that 1 neighbour is not enough for prediction, so it is better to take 3 closest neighbours at least.

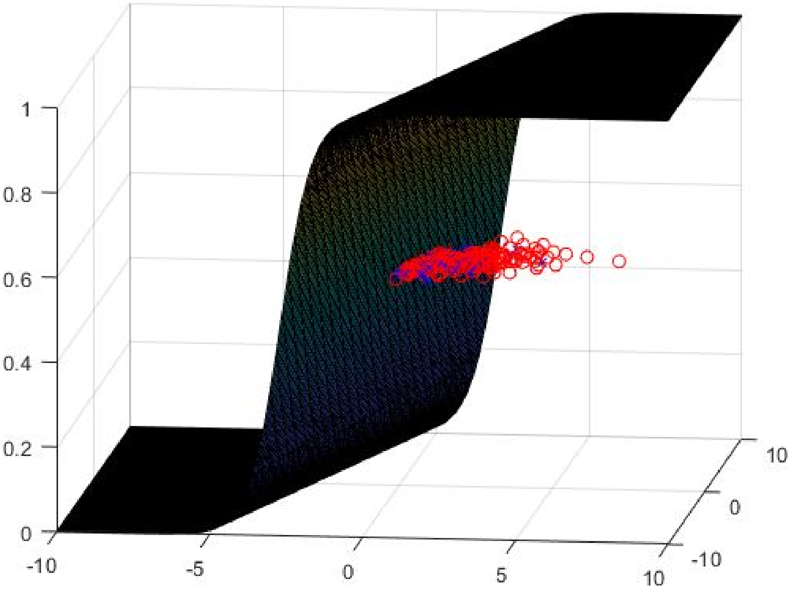
After that we implement distance-to-mean classifier. So we measure the distance from each point to means of 2 classes and predict the class from the nearest mean value. We compute distance in 2 different ways:

1) Euclidean distance: $D_{(x1,x2)} = \sqrt[2]{(x1-x2)^T(x1-x2)}$
2) Mahalanobis distance: $DM_{(x,m1)} = (x-m1)^T C^{-1}(x-m1)$

For 1st case accuracy is 71,5%, for 2nd is 72,5%.
We can summarise, that Fisher DD predicts data better than other classifiers. Mahalanobis works with the same accuracy as Euclidean (similarly).

The graph for the posterior probabilities of one of the class start with 0 number for start of the interval, where the another class is, the probability of current class tends to 50% in the cross section of 2 classes, and the probability rises up to 100% in the area where only the current class located. The proof is provided in Img8 (for the class with red points).

Img8. Plot of three dimensional graph of the posterior probability of 1 class for the Bayes classifier



Bayes classifier:

$P[w1|x] \;=\; P[w2|x]$

$\frac{1}{\sqrt{|C1|\sqrt{2\pi}}}exp\{-\frac{1}{2}(x-m1)'C1^{-1}(x-m1)\}P[w1]\} \;=\; \frac{1}{\sqrt{|C2|\sqrt{2\pi}}}exp\{-\frac{1}{2}(x-m2)'C2^{-1}(x-m2)\}P[w2]\}$

As C1 = C2 and P[w1]=P[w2]=P[w] =>

$exp\{\,-\frac{1}{2}(x-m1)'C^{-1}(x-m1)\}P[w] = exp\{-\frac{1}{2}(x-m2)'C^{-1}(x-m2)\}P[w]\}$

taking ln of each side:

$ln(P[w]) - \frac{1}{2}(x-m1)'C^{-1}(x-m1) = ln(P[w]) - \frac{1}{2}(x-m2)'C^{-1}(x-m2)$

$-\frac{1}{2}(x-m1)'C^{-1}(x-m1) =- \frac{1}{2}(x-m2)'C^{-1}(x-m2)$

$(x-m1)'C^{-1}(x-m1) = (x-m2)'C^{-1}(x-m2)$

$(x-m2)'(C^{-1}x - C^{-1}m2) - (x-m1)'(C^{-1}x - C^{-1}m1) = 0$

$x'C^{-1}x - x'C^{-1}m2 - m2'C^{-1}x + m2'C^{-1}m2 - x'C^{-1}x + x'C^{-1}m1 + m1'C^{-1}x - m1'C^{-1}m1 = 0$

$as\ m2'C^{-1}m2 - m1'C^{-1}m1 = b\ and\ x'C^{-1}x - x'C^{-1}x = 0 =>$

$-x'C^{-1}m2 + x'C^{-1}m1 - m2'C^{-1}x + m1'C^{-1}x + b = 0$

$x'C^{-1}(m1 - m2) + (-m2 + m1)C^{-1}x + b = 0$

$2C^{-1}(m1 - m2)x + b = 0$

Also

$$P[w1\;|x] \;=\; \frac{P[w1]p(x|w1)}{P[w1]p(x|w1)+P[w2]p(x/w2)} \;=\; \frac{1}{1+\frac{P[w2]p(x|w2)}{P[w1]p(x|w1)}} \;=$$

$$=\; \frac{1}{1+\frac{\frac{1}{\sqrt{|C|\sqrt{2\pi}}}}{\frac{1}{\sqrt{|C|\sqrt{2\pi}}}}*\frac{exp\{-\frac{1}{2}(x-m2)'C^{-1}(x-m2)\}P[w2]}{exp\{-\frac{1}{2}(x-m1)'C^{-1}(x-m1)\}P[w1]}} \;=$$

$$= \frac{1}{1+e^{(-\frac{1}{2}(x-m2)'C^{-1}(x-m2)+\frac{1}{2}(x-m1)'C^{-1}(x-m1))}*e^{ln(\frac{P[w2]}{P[w1]})}} =$$

$= \frac{1}{1+e^{-\frac{1}{2}(x-m2)'C^{-1}(x-m2)+\frac{1}{2}(x-m1)'C^{-1}(x-m1)}}$ => function similar to sigmoid

$$y = \frac{1}{1+e^{-x}}$$

(and Img8 is sigmoid also).

If the covariance matrix is different for 2 classes, than the Bayes optimal class boundary becomes quadratic function:

$$P[w1|x] \; = \; P[w2|x]$$

$$\frac{1}{\sqrt{|C1|}\sqrt{2\pi}}exp\{-\tfrac{1}{2}(x-m1)'C1^{-1}(x-m1)\}P[w1]\} = \frac{1}{\sqrt{|C2|}\sqrt{2\pi}}exp\{-\tfrac{1}{2}(x-m2)'C2^{-1}(x-m2)\}P[w2]\}$$

$$ln(\frac{1}{\sqrt{|C1|}}exp\{-\tfrac{1}{2}(x-m1)'C1^{-1}(x-m1)\}P[w1]\}) = ln(\frac{1}{\sqrt{|C2|}}exp\{-\tfrac{1}{2}(x-m2)'C2^{-1}(x-m2)\}P[w2]\})$$

$$ln(\frac{1}{\sqrt{|C1|}}) + ln(P[w1]) - \tfrac{1}{2}(x-m1)'C1^{-1}(x-m1) = ln(\frac{1}{\sqrt{|C2|}}) + ln(P[w2]) - \tfrac{1}{2}(x-m2)'C2^{-1}(x-m2)$$

$$as\ P[w1] = P[w2]:$$

$$-\tfrac{1}{2}(x-m1)'C1^{-1}(x-m1) + \tfrac{1}{2}(x-m2)'C2^{-1}(x-m2) = ln(\frac{1}{\sqrt{|C2|}}) - ln(\frac{1}{\sqrt{|C1|}})$$

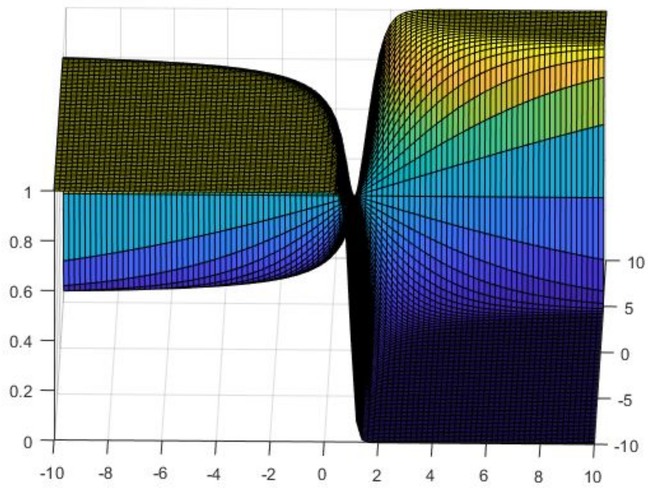
$$-\tfrac{1}{2}(x-m1)'C1^{-1}(x-m1) + \tfrac{1}{2}(x-m2)'C2^{-1}(x-m2) - ln(\frac{\sqrt{|C1|}}{\sqrt{|C2|}}) = 0$$

$$-\tfrac{1}{2}(x'C1^{-1}x - m1'C1^{-1}x - xC1^{-1}m1 + m1'C1^{-1}m1) + \tfrac{1}{2}(x'C2^{-1}x - m2'C2^{-1}x - x'C2^{-1}m2 + m2'C2^{-1}m2) -$$

$$-\,ln(\frac{\sqrt{|C1|}}{\sqrt{|C2|}}) = 0$$

$$-x'C1^{-1}x + 2m1'C1^{-1}x - m1'C1^{-1}m1 + x'C2^{-1}x - 2m2'C2^{-1}x + m2'C2^{-1}m2 - 2ln(\frac{\sqrt{|C1|}}{\sqrt{|C2|}}) = 0$$

$$x'(C2^{-1} - C1^{-1})x + (2m1'C1^{-1} - 2m2'C2^{-1})x - m1'C1^{-1}m1 + m2'C2^{-1}m2 - 2ln(\frac{\sqrt{|C1|}}{\sqrt{|C2|}}) = 0$$



Img9. the posterior probability of 1 class for the Bayes classifier for different matrices