Pavan Rathnakar Shetty
EECE 5644
HW3
Due date: November 16, 11:59 pm

GitHub: Files in the hw3 folder

https://github.com/Pavan-r-shetty/5644.git

H.W.3

1) Question 1

Given $p(x) = p(x|L=0)P(L=0) + p(x|L=1)P(L=1)$

L is the class label

$P(L=0) = 0.6$

$P(L=1) = 0.4$

$p(x|L=0) = w_1 g(x|m_{01}, C_{01})$

$p(x|L=1) = g(x|m_1, C_1)$

$g(x|m, C)$ - Multivariate Gaussian PDF

$w_1 = w_2 = 0.5$

Class 0: $P(L=0) = 0.6$

$m_{01} = \begin{bmatrix} 5 \\ 0 \end{bmatrix}$ $m_{02} = \begin{bmatrix} 0 \\ 4 \end{bmatrix}$

$C_{01} = \begin{bmatrix} 4 & 0 \\ 0 & 2 \end{bmatrix}$ $C_{02} = \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix}$

Class 1: $P(L=1) = 0.4$

$m_1 = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$

$C_1 = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$

4 datasets were generated with the given means and covariances

D_{train}^{100} - 100 samples and their labels for training

D_{train}^{1000} - 1000 samples and their labels for training

D_{train}^{10000} - 10,000 samples and their labels for training

$D_{validate}^{20000}$ - 20,000 samples and their labels for validation.

Part 1: Theoretical optimal classifier that achieves minimum probability of error using the known parameters and the pdf.

Minimum expected risk classification rule in the form of likelihood ratio is given by:

$$\left(D=1\right) \quad \frac{p(X|L_1)}{p(X|L_0)} \geq \frac{\lambda_{10}-\lambda_{00}}{\lambda_{01}-\lambda_{11}} * \frac{P(L_0)}{P(L_1)} = \gamma \quad \left(D=0\right)$$

Minimize probability of misclassification by setting penalty.

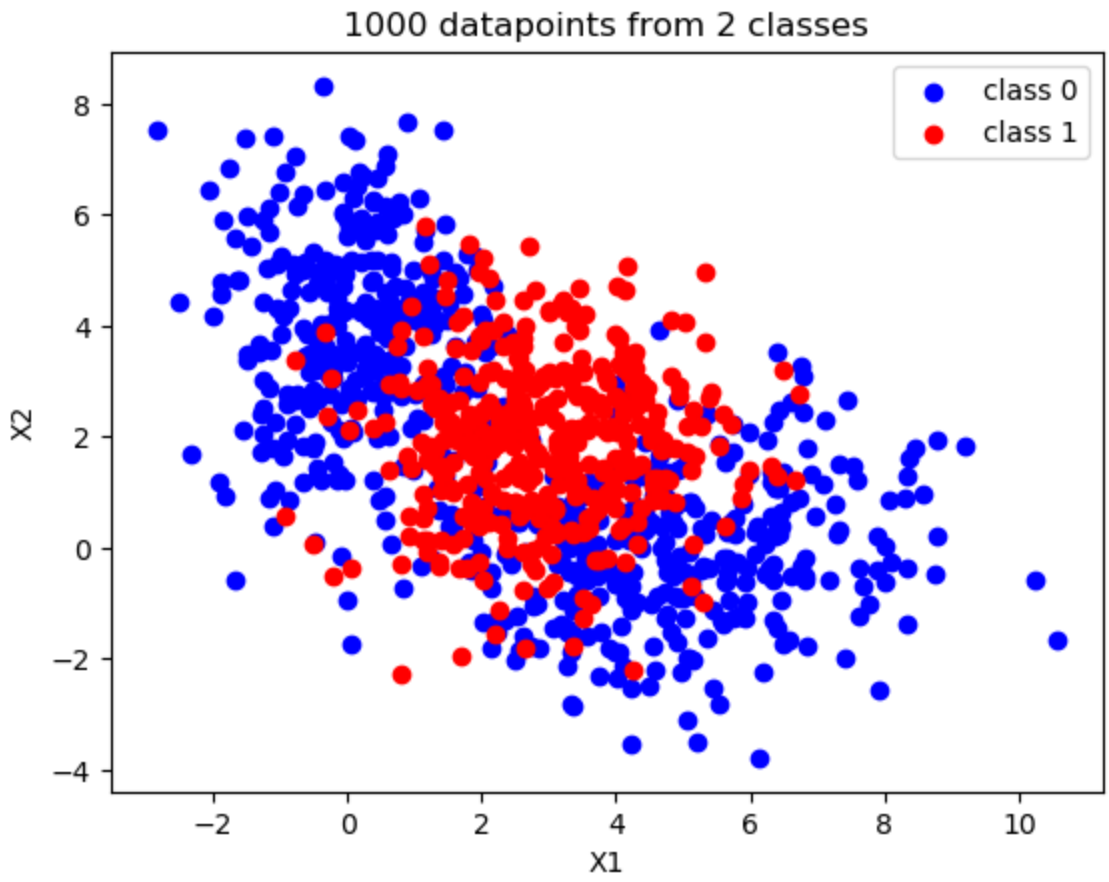
Penalty for wrong classification = 1
Penalty for correct classification = 0

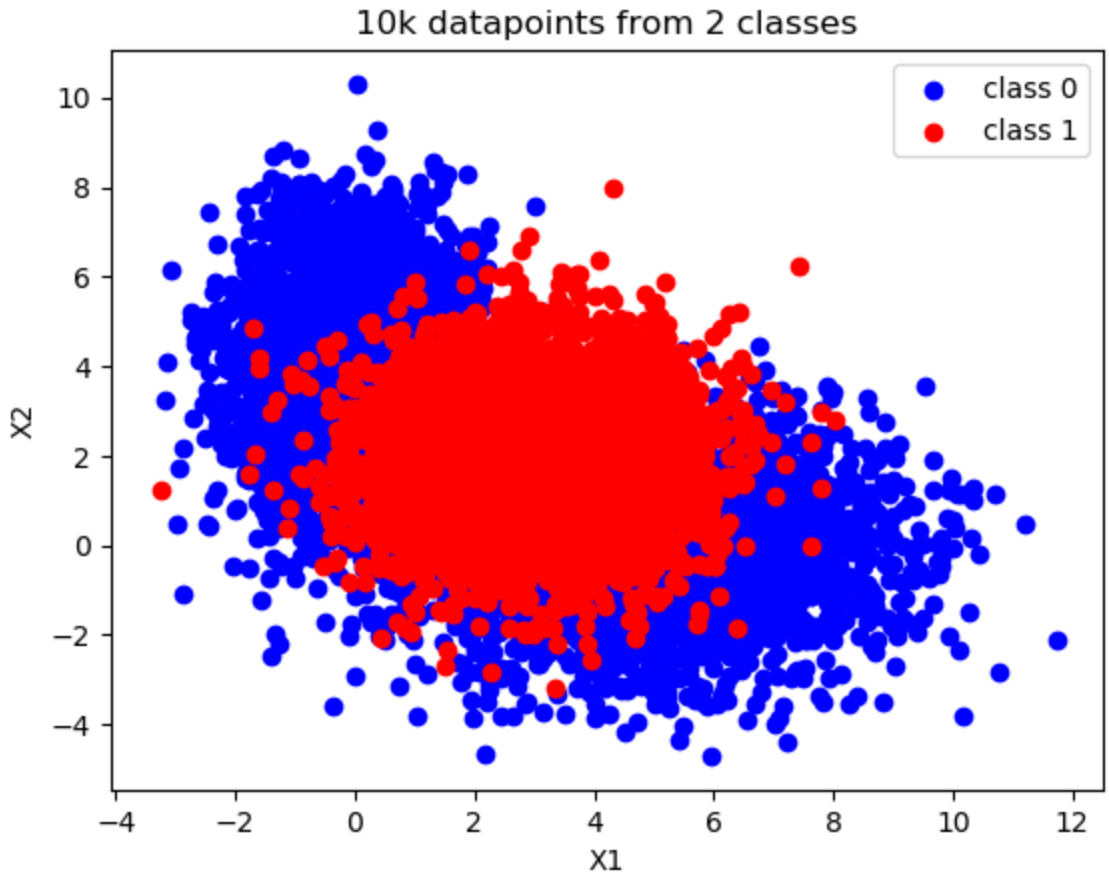
$$\left(D=1\right) \quad \frac{P(X|L_1)}{P(X|L_0)} \geq \frac{1-0}{1-0} \times \frac{0.6}{0.4} \quad \left(D=0\right)$$

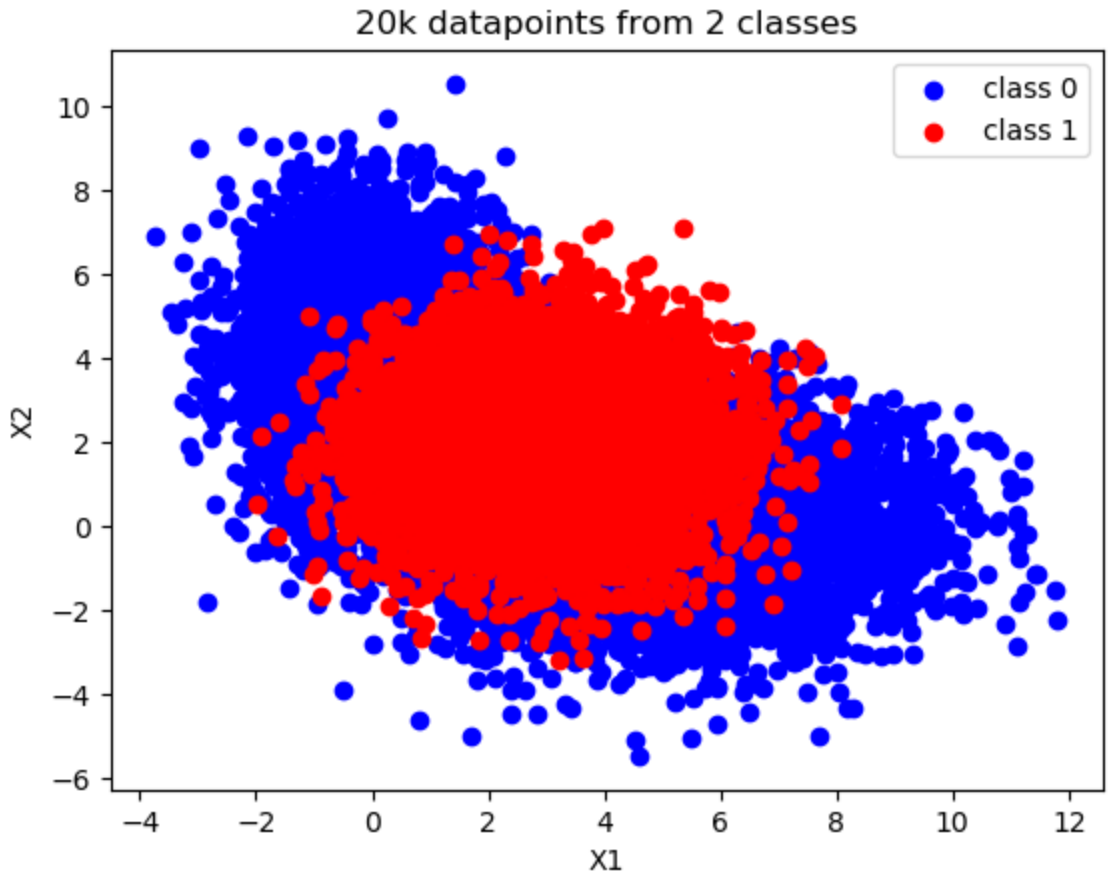
$$= 1.5 = \gamma_{theoretical}$$

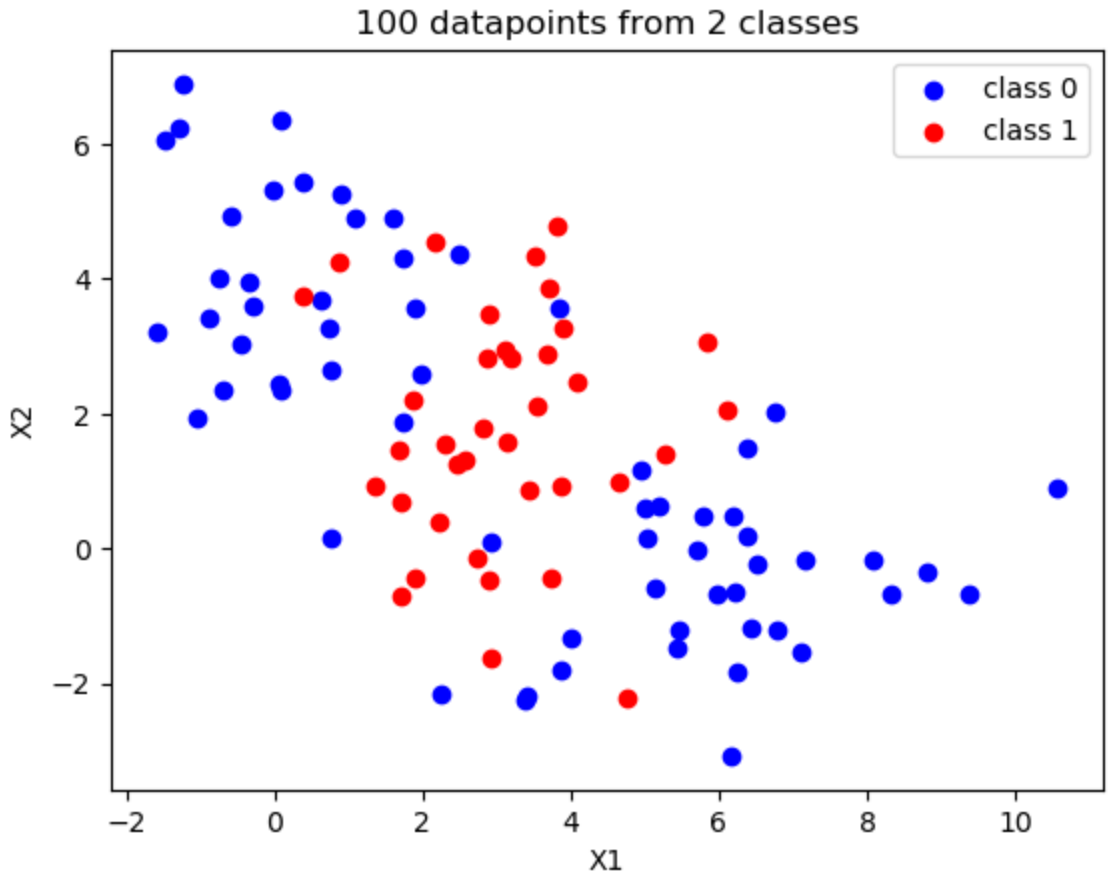
The classifier is implemented on the validation set and ROC curve is plotted as described:

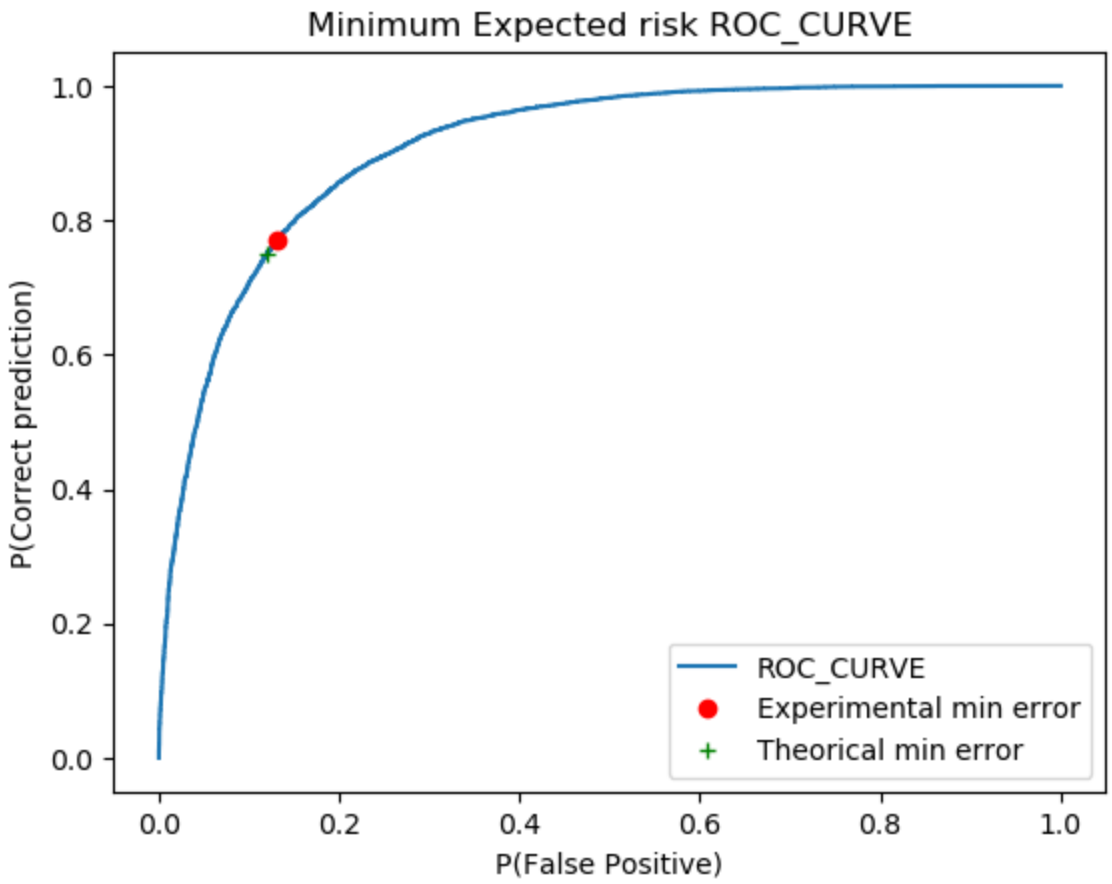
	γ	Min Perror
Theoretical	1.5	0.1740
Estimated from data	1.64	0.1688


1000 datapoints from 2 classes
class 0
class 1
X1
X2
8
6
4
2
0
−2
−4
−2
0
2
4
6
8
10


10k datapoints from 2 classes
class 0
class 1
X1
X2
10
8
6
4
2
0
−2
−4
−4
−2
0
2
4
6
8
10
12


20k datapoints from 2 classes
class 0
class 1
X1
X2
10
8
6
4
2
0
−2
−4
−6
−4
−2
0
2
4
6
8
10
12


100 datapoints from 2 classes
class 0
class 1
X1
X2
−2
0
2
4
6
8
10
−2
0
2
4
6


Minimum Expected risk ROC_CURVE
P(Correct prediction)
P(False Positive)
0.0
0.2
0.4
0.6
0.8
1.0
ROC_CURVE
Experimental min error
Theorical min error

Part B:

ML estimation is performed on the three training sets.

Parameter Estimation for class 0 and class 1 was done using sklearn.mixture.GaussianMixture.

EM Estimation was based on the Maximum likelihood Estimation of the class prior as well as the mean and covariance of the conditional pdf.

class 0:

$$\arg\max Q(\theta, \theta^g) = \sum_{c=1}^{M} \sum_{i=1}^{N} \ln(\alpha_c)\, p(c \mid x_i; \theta^g) + \sum_{c=1}^{M} \sum_{i=1}^{N} p(c \mid x_i; \theta^g) \ln\left(p_c(x_i \mid \theta_c)\right)$$

Maximizing with respect to α, μ, Σ we get.

$$\hat{\alpha}_c = \frac{1}{N} \sum_{i=1}^{N} p_c(x_i \mid x_c, \theta^g)$$

$$\hat{\mu}_c = \frac{\sum_{i=1}^{N} x_i\, p(c \mid x_i; \theta)}{\sum_{i=1}^{N} p(c \mid x_i; \theta)}$$

$$\hat{\Sigma}_c = \sum_{i=1}^{N} \frac{p(c \mid x_i; \theta^g)\,(x_i - \mu_c)(x_i - \mu_c)^T}{\sum_{i=1}^{N} p(c \mid x_i; \theta^g)}$$

class 1: since it has 1 gaussian component, the maximum likelihood estimates are sample average and covariance

$$\hat{\mu} = \frac{1}{N} \sum x_i \qquad \hat{\Sigma} = \frac{1}{N} \sum (x_i - \hat{\mu})(x_i - \hat{\mu})^T$$

Estimated Means

	D^{train}_{100}	D^{train}_{1000}	D^{train}_{10000}
w_{01}	$\begin{bmatrix} -0.13 \\ 3.91 \end{bmatrix}$	$\begin{bmatrix} 5.20 \\ 0.13 \end{bmatrix}$	$\begin{bmatrix} 5.11 \\ -0.09 \end{bmatrix}$
w_{02}	$\begin{bmatrix} 4.96 \\ -0.08 \end{bmatrix}$	$\begin{bmatrix} -0.017 \\ 3.89 \end{bmatrix}$	$\begin{bmatrix} 0.026 \\ 3.95 \end{bmatrix}$
w_1	$\begin{bmatrix} 2.58 \\ 2.081 \end{bmatrix}$	$\begin{bmatrix} 2.91 \\ 2.06 \end{bmatrix}$	$\begin{bmatrix} 2.95 \\ 1.98 \end{bmatrix}$

Estimated Covariance

	D^{train}_{100}	D^{train}_{1000}	D^{train}_{10000}
Cov_{01}	$\begin{bmatrix} 0.86 & -0.35 \\ -0.35 & 2.66 \end{bmatrix}$	$\begin{bmatrix} 3.019 & 0.170 \\ 0.170 & 1.887 \end{bmatrix}$	$\begin{bmatrix} 3.65 & 0.049 \\ 0.049 & 1.95 \end{bmatrix}$
Cov_{02}	$\begin{bmatrix} 4.96 & 0.137 \\ 0.137 & 1.79 \end{bmatrix}$	$\begin{bmatrix} 093 & 0.11 \\ 0.11 & 3.15 \end{bmatrix}$	$\begin{bmatrix} 1.03 & -0.04 \\ -0.04 & 3.10 \end{bmatrix}$
Cov_{11}	$\begin{bmatrix} 2.16 & -0.66 \\ -0.66 & 1.91 \end{bmatrix}$	$\begin{bmatrix} 1.90 & -0.092 \\ -0.092 & 2.02 \end{bmatrix}$	$\begin{bmatrix} 2.03 & 0.04 \\ 0.04 & 2.01 \end{bmatrix}$

Estimated Alphas

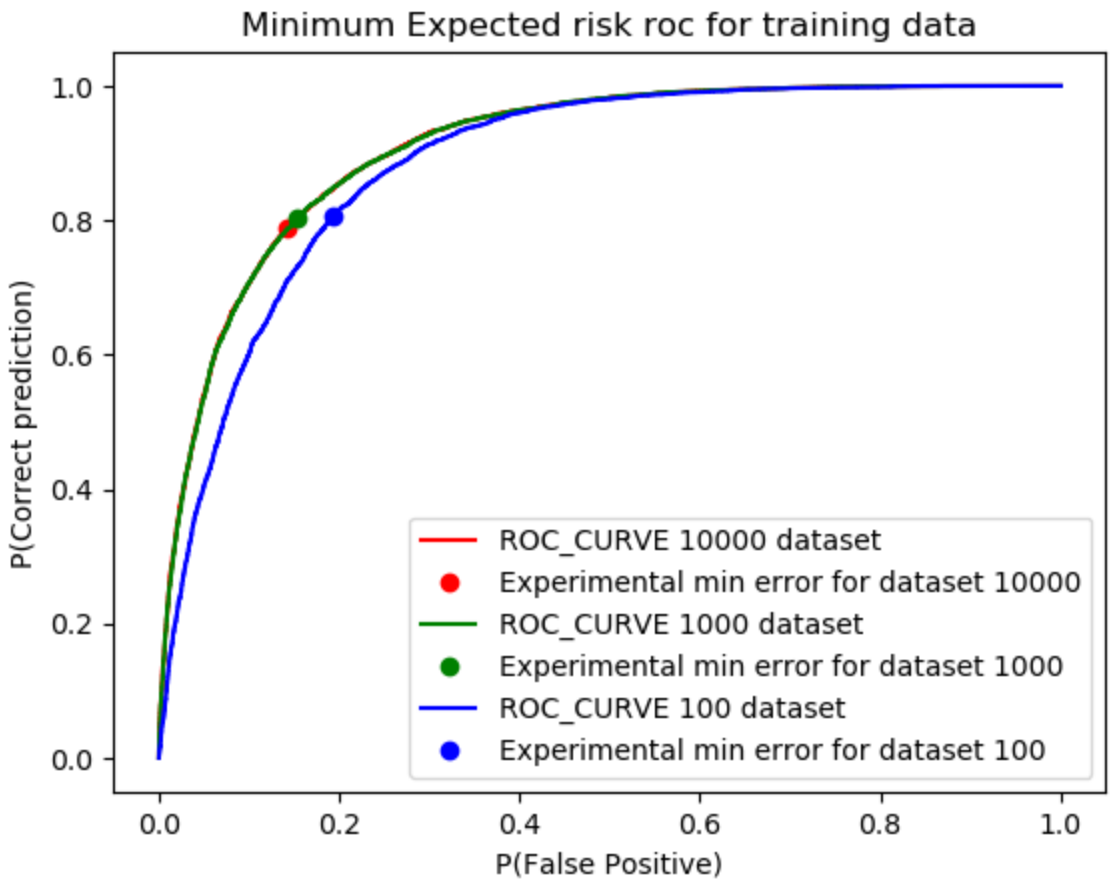
	D^{train}_{100}	D^{train}_{1000}	D^{train}_{10000}
α_{01}	0.46	0.515	0.49
α_{02}	0.55	0.484	0.50

Prior Probabilities

	D^{train}_{100}	D^{train}_{1000}	D^{train}_{10000}
$P(L=0)$	0.590	0.598	0.59
$P(L=1)$	0.402	0.41	0.40

Minimum Probability Error for the datasets

Training samples	$D_{minerror}$	Min Probability of error
D^{train}_{100}	1.49	0.1023
D^{train}_{1000}	1.54	0.179
D^{train}_{10000}	1.53	0.176


Minimum Expected risk roc for training data
P(Correct prediction)
P(False Positive)
0.0
0.2
0.4
0.6
0.8
1.0
ROC_CURVE 10000 dataset
Experimental min error for dataset 10000
ROC_CURVE 1000 dataset
Experimental min error for dataset 1000
ROC_CURVE 100 dataset
Experimental min error for dataset 100

Part C: classifier using logistic function

Maximum likelihood estimation techniques were used to train logistic linear and logistic quadratic based approximation of class label posterior function given sample.

Logistic function:

$$h(x, w) = \frac{1}{1 + e^{w^T z(x)}}$$

for linear function $z(x) = [1, x_1, x_2]^T$

for quadratic $z(x) = [1 \; x_1 \; x_2 \; x_1^2 \; x_1 x_2 \; x_2^2]^T$

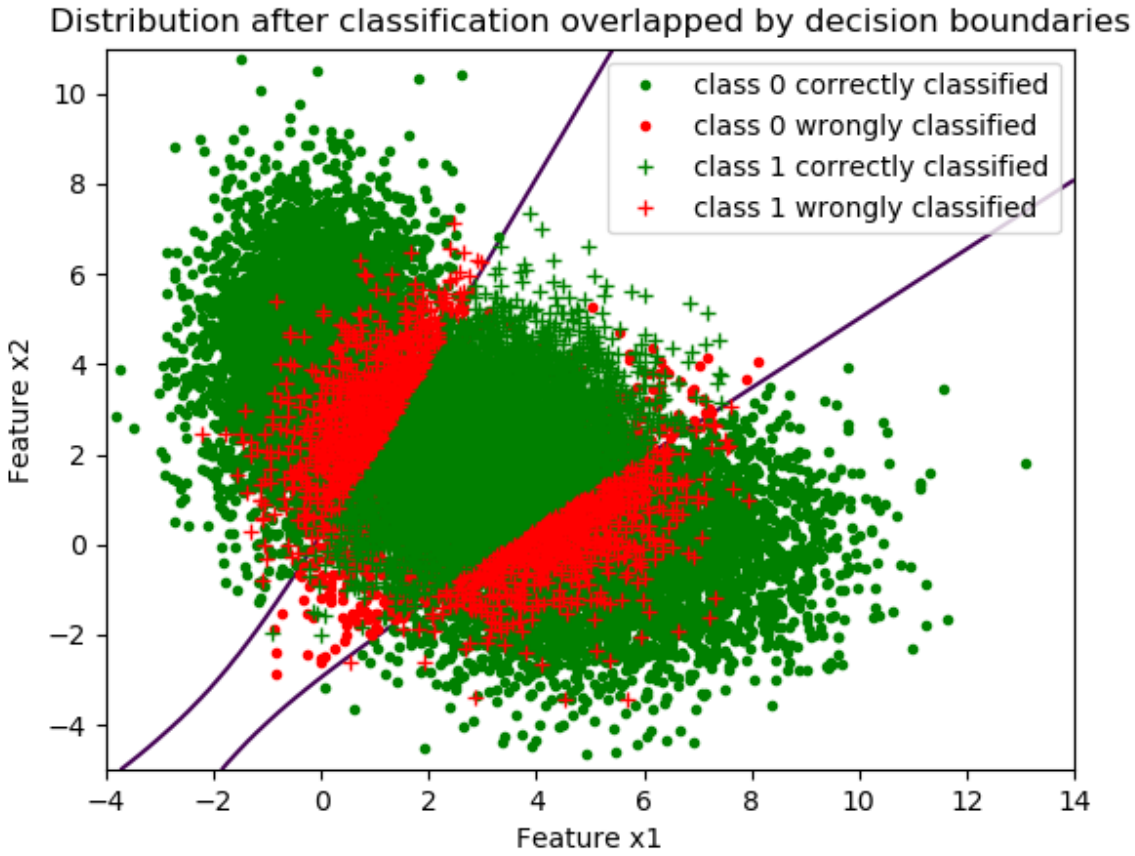
The weights are estimated using numerical optimization techniques with the cost function

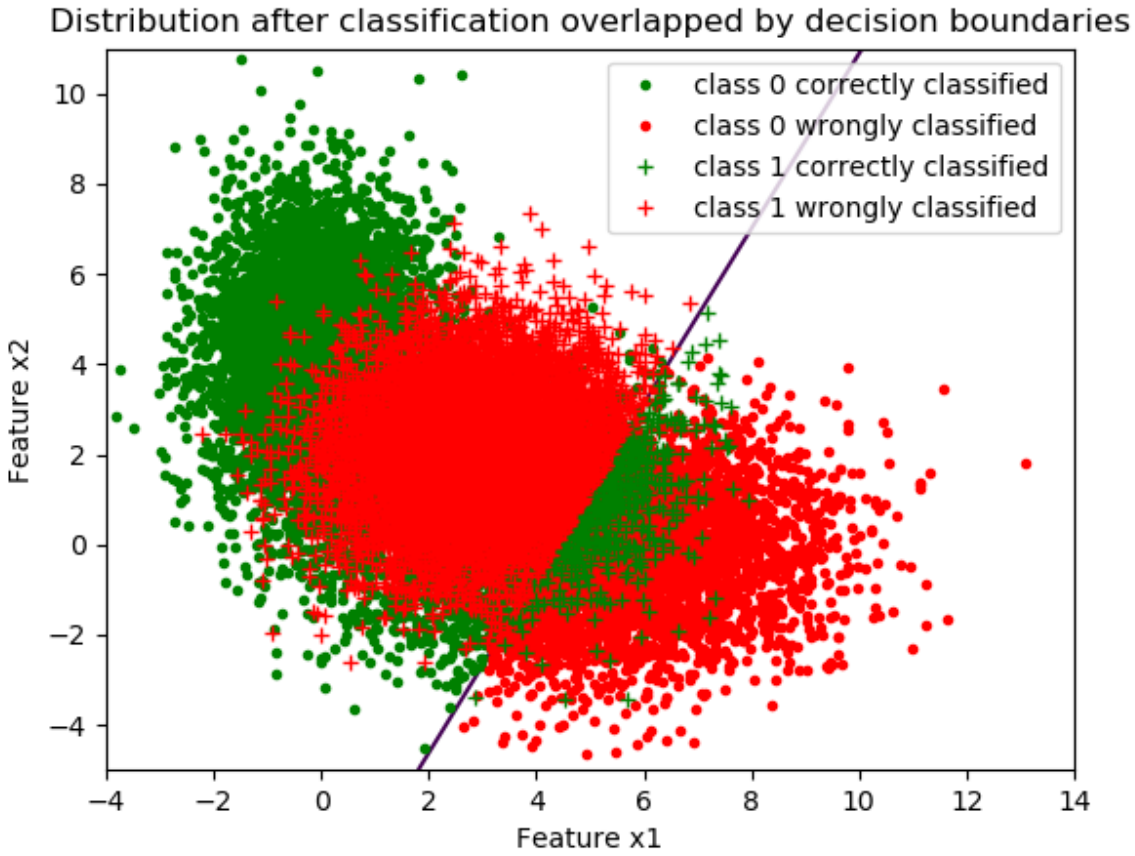
$$\hat{\theta}_{ML} = \frac{-1}{N} \sum_{n=1}^{N} l_n \ln\left(h(x_n, \theta)\right) + (1 - l_n)(\ln(1 - h(x_n, \theta)))$$

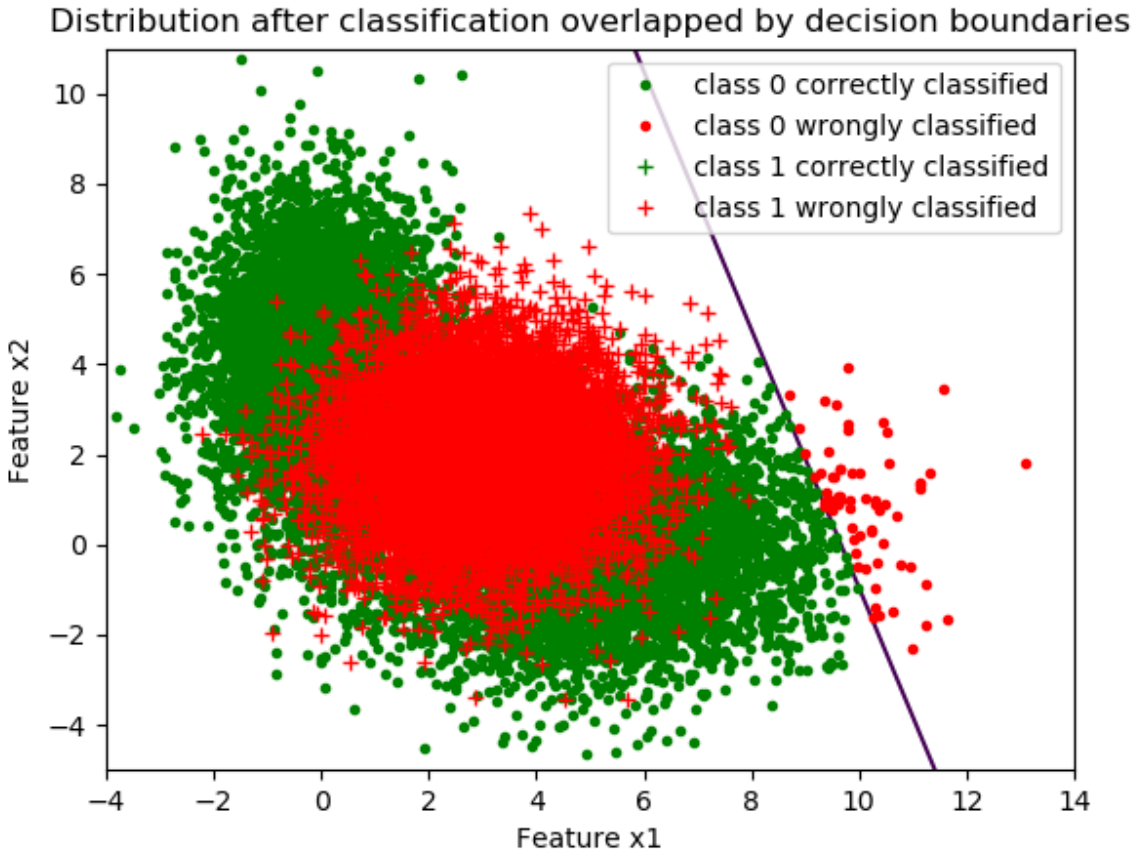
The minimum expected risk classification criteria is

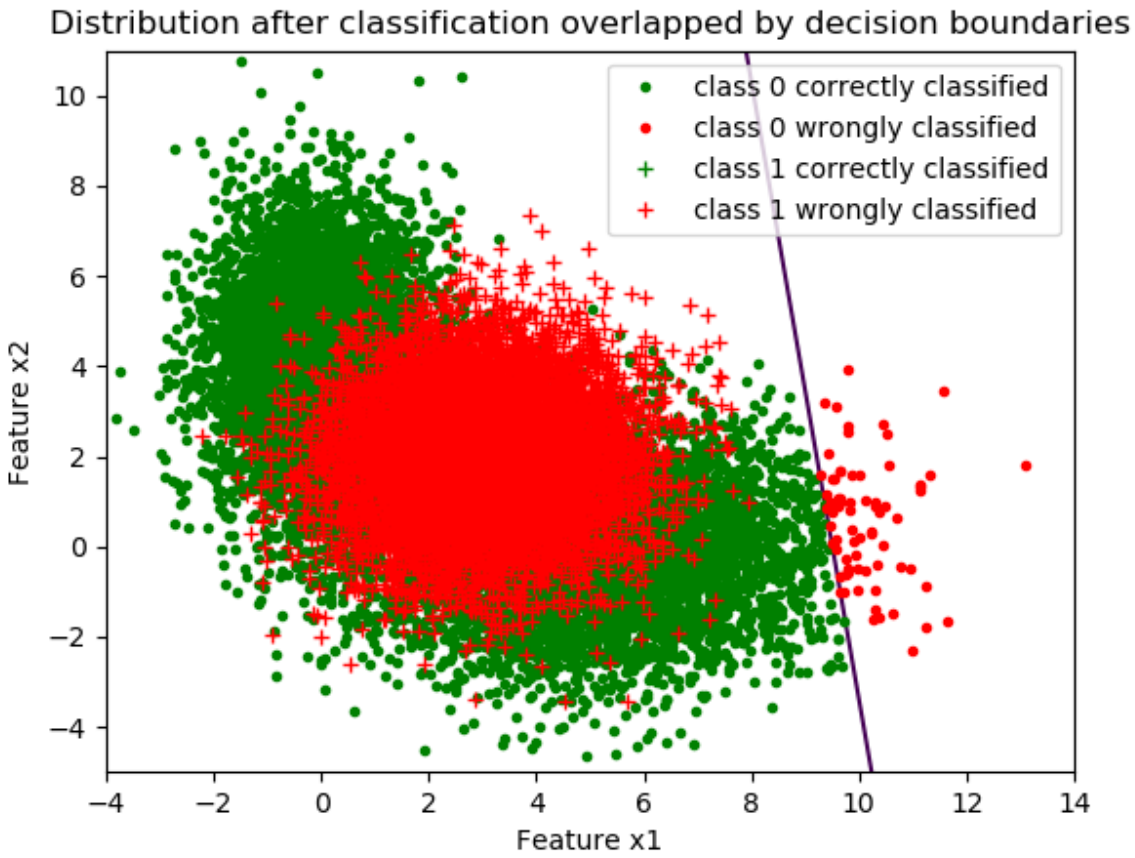
$$(l_n = 1) \quad w^T z(x) \geq 0 \quad (l_n = 0)$$

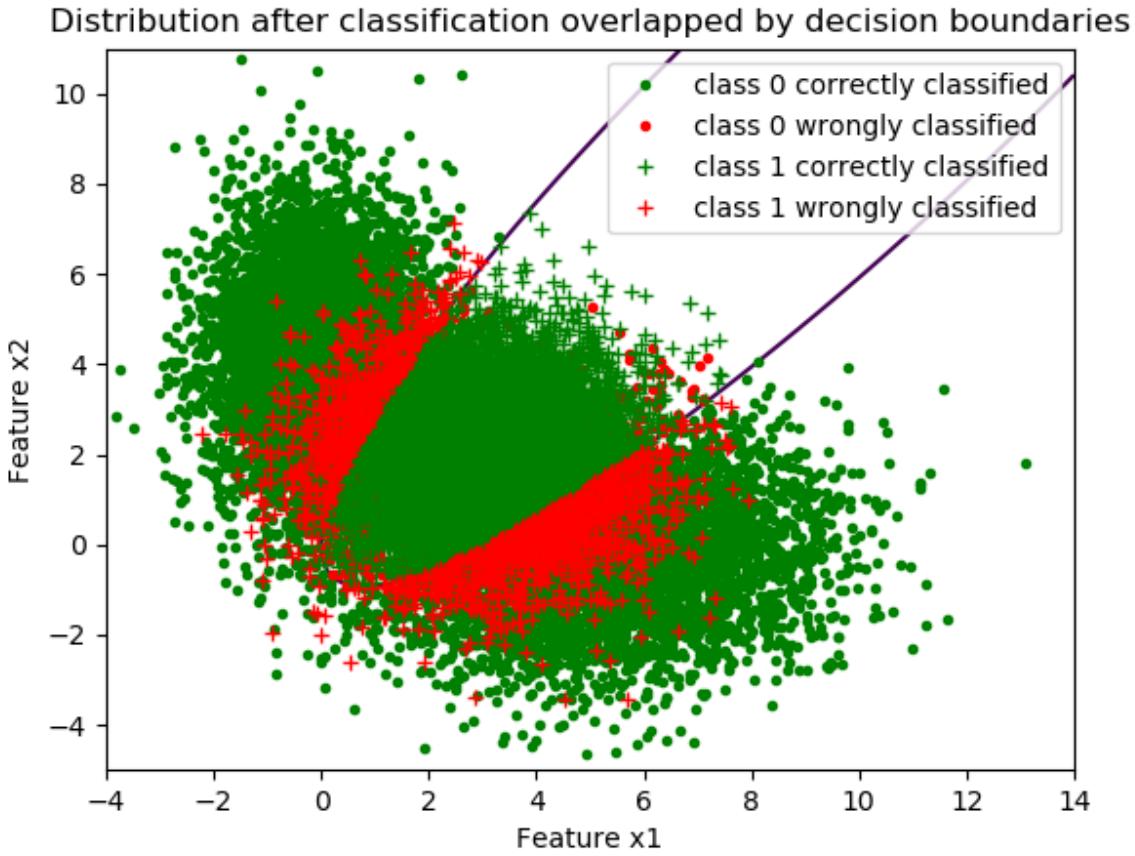
Logistic function	Probability error
	P_{error} - Linear	P_{error} - Quadratic
D_{train} 100	0.46	0.18
D_{train} 1000	0.42	0.17
D_{train} 10000	0.41	0.17

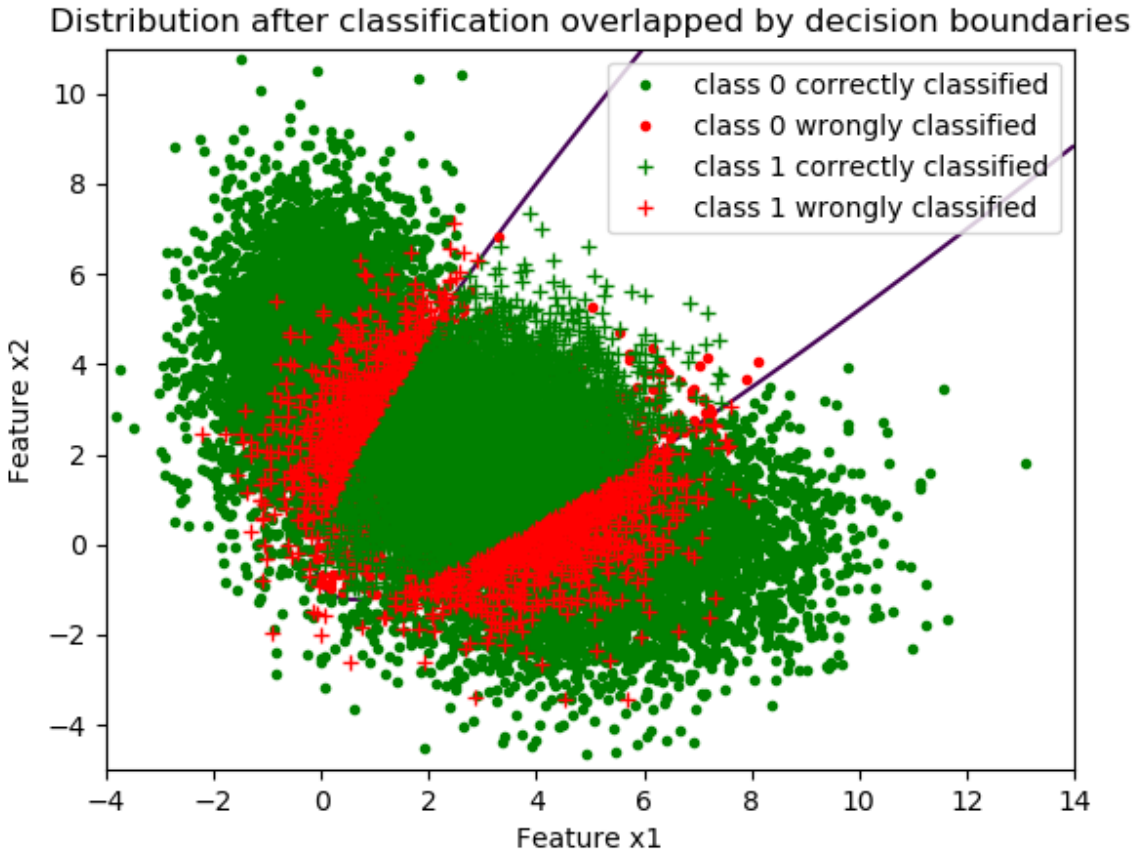

Distribution after classification overlapped by decision boundaries
class 0 correctly classified
class 0 wrongly classified
class 1 correctly classified
class 1 wrongly classified
Feature x2
Feature x1
10
8
6
4
2
0
−2
−4
−4
−2
0
2
4
6
8
10
12
14


Distribution after classification overlapped by decision boundaries
class 0 correctly classified
class 0 wrongly classified
class 1 correctly classified
class 1 wrongly classified
Feature x2
Feature x1
10
8
6
4
2
0
−2
−4
−4
−2
0
2
4
6
8
10
12
14


Distribution after classification overlapped by decision boundaries
class 0 correctly classified
class 0 wrongly classified
class 1 correctly classified
class 1 wrongly classified
Feature x2
Feature x1
10
8
6
4
2
0
−2
−4
−4
−2
0
2
4
6
8
10
12
14


Distribution after classification overlapped by decision boundaries
class 0 correctly classified
class 0 wrongly classified
class 1 correctly classified
class 1 wrongly classified
Feature x2
Feature x1
10
8
6
4
2
0
−2
−4
−4
−2
0
2
4
6
8
10
12
14


Distribution after classification overlapped by decision boundaries
class 0 correctly classified
class 0 wrongly classified
class 1 correctly classified
class 1 wrongly classified
Feature x2
Feature x1
10
8
6
4
2
0
−2
−4
−4
−2
0
2
4
6
8
10
12
14


Distribution after classification overlapped by decision boundaries
class 0 correctly classified
class 0 wrongly classified
class 1 correctly classified
class 1 wrongly classified
Feature x2
Feature x1
10
8
6
4
2
0
−2
−4
−4
−2
0
2
4
6
8
10
12
14

Question 2:

To find: $[x\ y]^T$ coordinate position with highest probability.

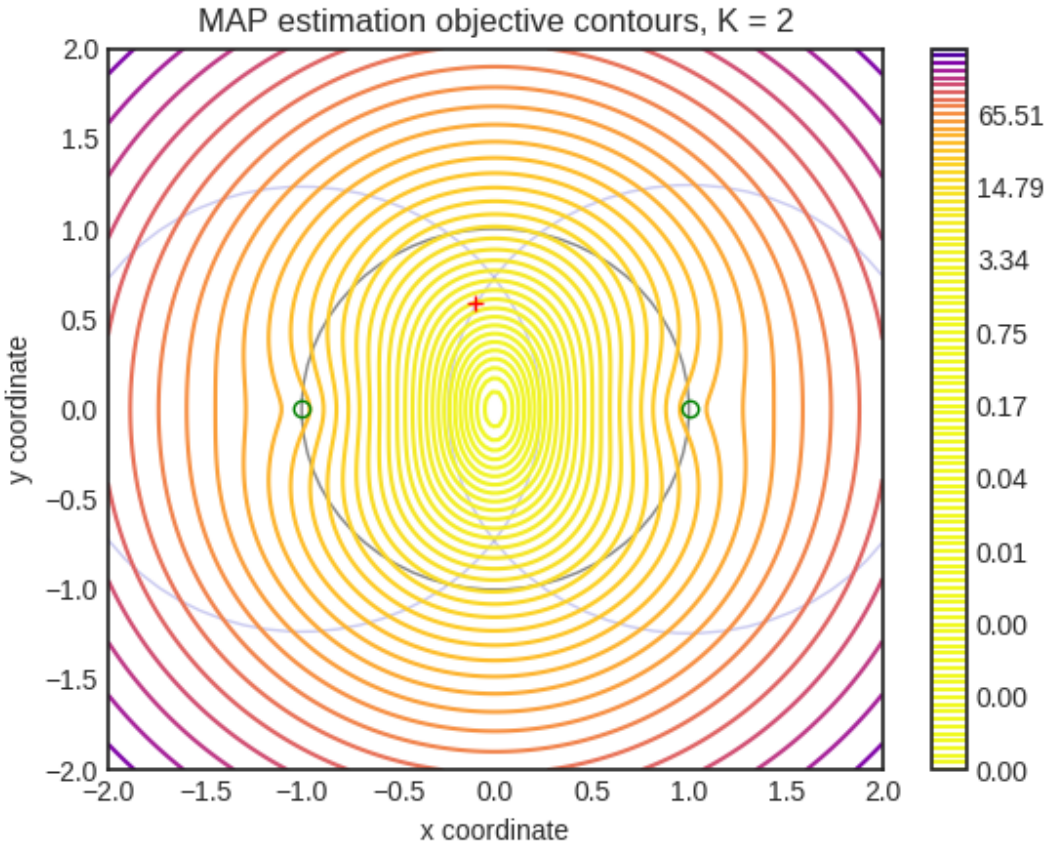
given: Prior distribution of range of measurements for each of the K reference coordinates.

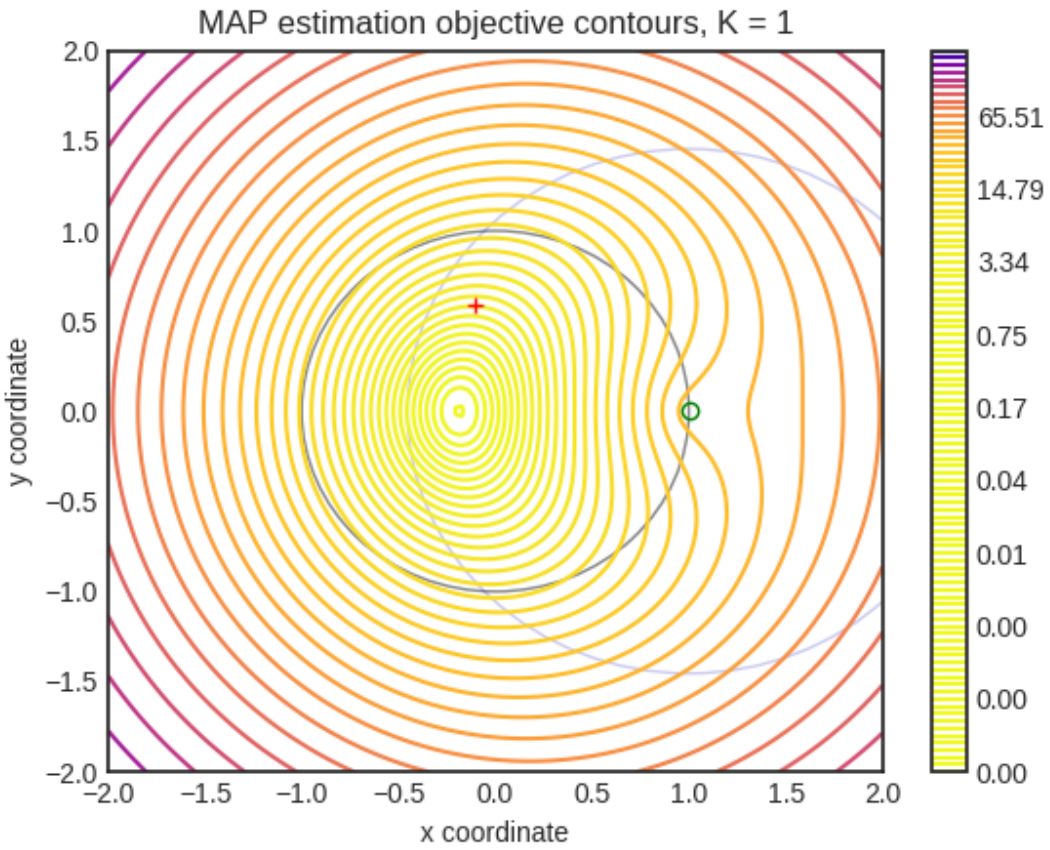
$$\begin{bmatrix} x_{map} \\ y_{map} \end{bmatrix} = \underset{\begin{bmatrix} x \\ y \end{bmatrix}}{\arg\max}\ p\begin{bmatrix} x \\ y \end{bmatrix} \{r_1 \dots r_k\}$$

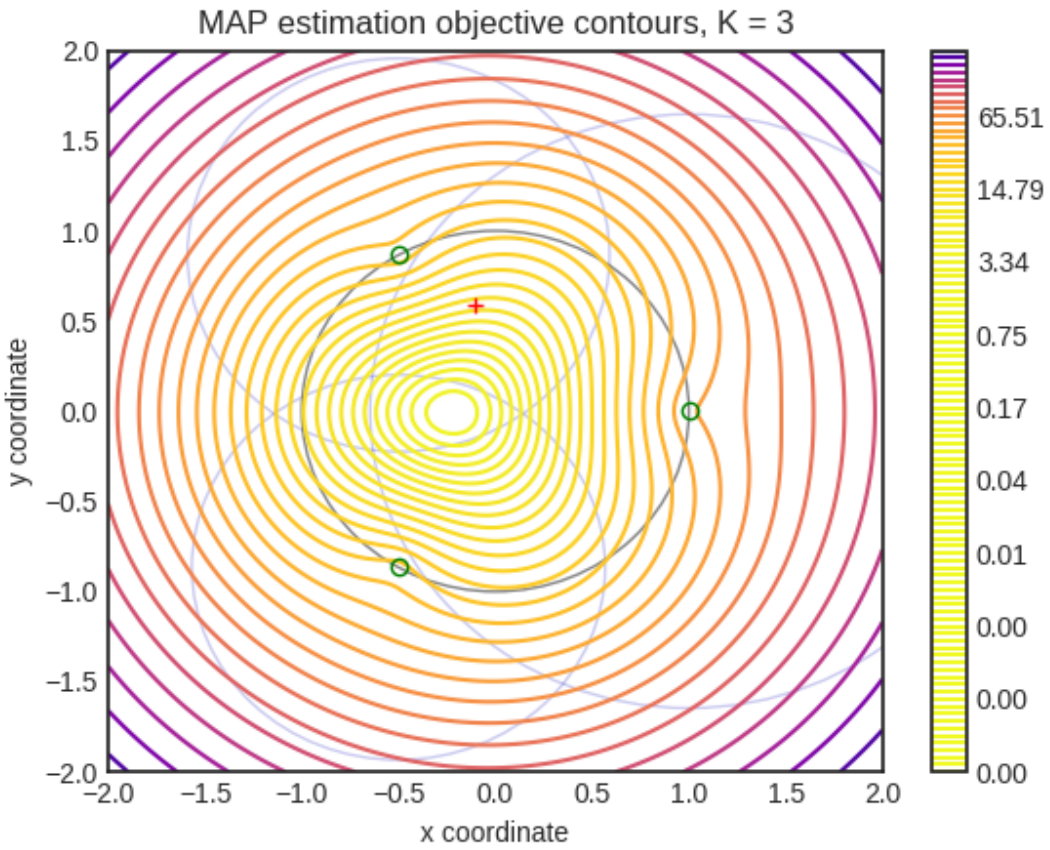
$$= \underset{\begin{bmatrix} x \\ y \end{bmatrix}}{\arg\max} \left(\left(2\pi \sigma_x \sigma_y \right)^{-1} \cdot e^{-\frac{1}{2}[x\ y] \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \end{bmatrix}} \right) \prod_{i=1}^{k} p\left[\begin{matrix} x \\ y \end{matrix} \right] \Big| r_i$$

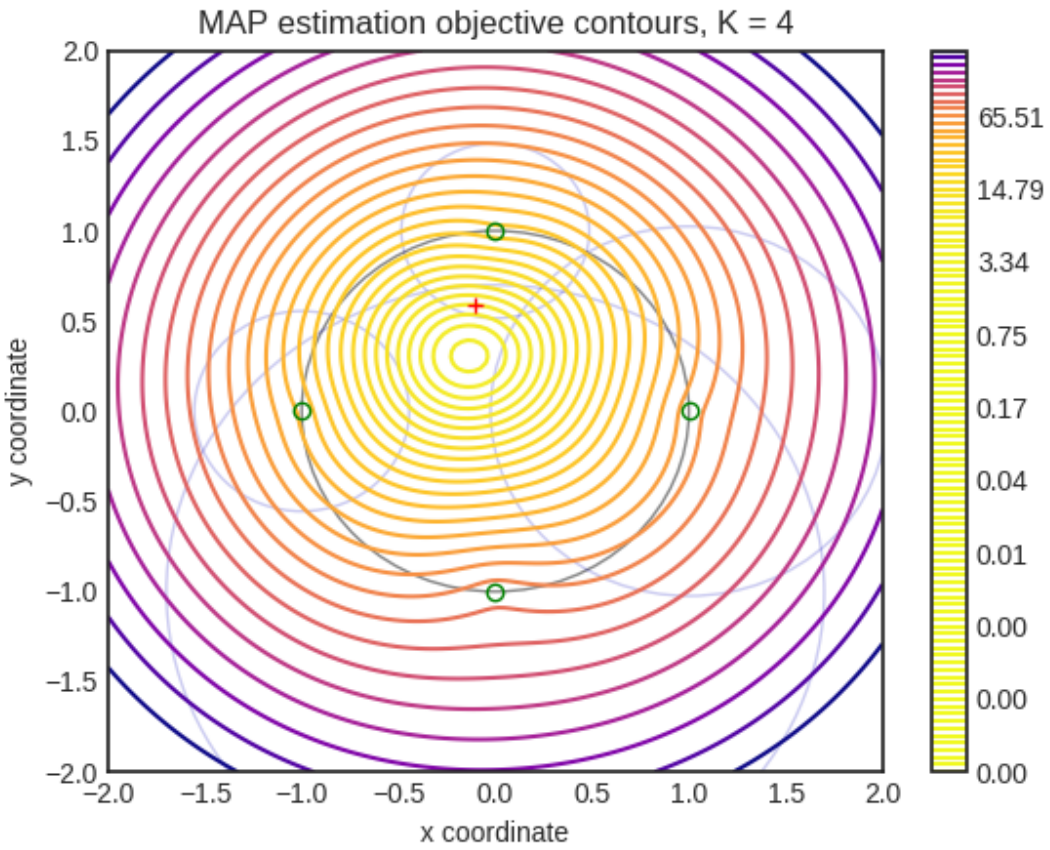
$$= \underset{\begin{bmatrix} x \\ y \end{bmatrix}}{\arg\max}\ \frac{-1}{2} [x\ y] \begin{bmatrix} \sigma_x^2 & 0 \\ 0 & \sigma_y^2 \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \end{bmatrix} + \sum_{i=1}^{k} \frac{(r_i - d_i)^2}{2\sigma_i^2}$$

where $d_i = \left\| \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} x_i \\ y_i \end{bmatrix} \right\|$


MAP estimation objective contours, K = 2
y coordinate
x coordinate
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
65.51
14.79
3.34
0.75
0.17
0.04
0.01
0.00
0.00
0.00


MAP estimation objective contours, K = 1
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
y coordinate
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x coordinate
65.51
14.79
3.34
0.75
0.17
0.04
0.01
0.00
0.00
0.00


MAP estimation objective contours, K = 3
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
y coordinate
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x coordinate
65.51
14.79
3.34
0.75
0.17
0.04
0.01
0.00
0.00
0.00


MAP estimation objective contours, K = 4
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
y coordinate
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
x coordinate
65.51
14.79
3.34
0.75
0.17
0.04
0.01
0.00
0.00
0.00

Explanation of working of the code:

- A random coordinate is generated at the origin as true coordinate.
- For every k, k is evenly distributed landmarks around the circle are used to calculate distance from the true vehicle position.
- These measurements have Gaussian white noise.
- The MAP estimation objective function is calculated for all points on 128 x 128 mesh grid, the values are then plotted as equilevel contours.
- Additionally, the unit circle (shown in gray) true location (red +), landmark location (in green) and the range reported by each landmark are plotted. (shown in faint blue circle)

Behaviour of MAP estimate of position:

- Map estimate for $k < 3$ is not accurate, estimates are symmetric around x, since the prior bias have y coordinate = 0.

- The estimate is much accurate of $k = 3$ & $k = 4$.

- In general as k increases, estimate get better, however it is not always true for $k = 1 \to 2$.

- In general as k increases, the estimator increases. This can be visualised on the contour graph by a shrinkage of area of location with a high probability.

Question 3

$$\lambda(\alpha_i|\omega_j) = \begin{cases} 0 & i = j \quad i, j = 1, \ldots, c \\ \lambda_r & i = (c+1) \\ \lambda_s & \text{otherwise} \end{cases}$$

To choose class with minimum risk, we have

$$\text{decision} = \underset{\substack{i=1\ldots c \\ j=1\ldots c}}{\operatorname{argmin}} \{\lambda_{ij} P(\omega_i|x)\}$$

we know that $\lambda_{ii} = 0$,

$\therefore$ $\operatorname{argmin} \{\lambda_{1i} P(\omega_1|x)) \ldots 0, \ldots \lambda_{cc} P(\omega_c|x)\}$

$\therefore$ minimum possible value is 0 when $j = i$.

But $j=i$, will be chosen only if $P(\omega_i|x)$ is the highest.

$\therefore$ to choose $j=i$,

$$P(\omega_i|x) \geq P(\omega_j|x) \quad \text{for } \forall\, j \in \{1, \ldots, c\}$$

The average risk of choosing class ω_i is given by

$$R(D|x) = \sum_{j=1}^{c} \lambda_{ij} \frac{P(x|\omega_j) P(\omega_j)}{P(x)}$$

$$R(D=\omega_i|x) = \sum_{j=1}^{c} \lambda_{ij} P(\omega_j|x)$$

$$= \sum_{j=1}^{i-1} \lambda_{ij} P(\omega_j|x) + \lambda_{ii} P(\omega_i|x) + \sum_{j=i+1}^{c} \lambda_{ij} P(\omega_j|x)$$

w.k.t $\lambda_{ii} = 0$

$$\therefore R(D=\omega_i|x) = \sum_{j=1, j\neq i}^{c} \lambda_{ij} P(\omega_j|x) + 0$$

$$= \lambda_{ij} \sum_{j=1, j\neq i}^{c} P(\omega_j|x) = \lambda_{ij}\left(1 - P(\omega_i|x)\right)$$

$\therefore P(\omega_i|x) \uparrow$ then risk $\downarrow$

for $i=c+1$, the risk is given by

$$R(D=c+1|x) = \lambda_r$$

Hence for achieving minimum risk (decide w_i)

we have

$$R(D=w_i|x) \leq R(D=c+1|x)$$

$$\lambda_s(1-P(w_i|x)) \leq \lambda_r$$

$$1-P(w_i|x) \leq \frac{\lambda_r}{\lambda_s}$$

$$\therefore P(w_i|x) = 1-\frac{\lambda_r}{\lambda_s}$$

Case 1: $\lambda_r = 0$

$\Rightarrow P(w_i|x) \geq 1 \Rightarrow P(w_i|x) = 0$ at $i = c+1$

$\Rightarrow$ reject always as cost of rejecting is 0

Case 2: $\lambda_r \geq \lambda_s$

$$\Rightarrow \frac{\lambda_r}{\lambda_s} > 1$$

$$\therefore P(w_i|x) \geq 1-\frac{\lambda_r}{\lambda_s}$$

$$\Rightarrow P(w_i|x) \leq \frac{\lambda_r}{\lambda_s} - 1$$

Implies cost of rejection is higher than cost of choosing any other value, then we never reject.