Machine Learning Lab 2

Di Wu

dw4y17@soton.ac.uk

1 Exercise 1 & 2

In the first and second exercises, we atempted to draw a line through two points on the line, verified the distance between origin and the line is equal to $-c/\sqrt{a_1^2 + b_1^2}$ visually. The result is shown at Figure 1. Then we tried to generate two bivariate datasets with respective mean $m_1 = [0; 2]$,$m_2 = [1.5; 0]$ but the same covariance matrix $C = [2 \quad 1; 1 \quad 2]$. The process is first we use the cholesky decomposition function to get matrix A while $A * A' = C$, next we transformed white data(bivariate gaussian distribution with mean=$[0; 0]$,covariance matrix $= [1 \quad 0; 0 \quad 1]$) to dataset Y with covariance matrix C, last we transform the Y to Y1 and Y2 with respective mean m_1 and m_2 by using the simple statistic method. The scatter plot of Y1 and Y2 is shown as below Figure 2.

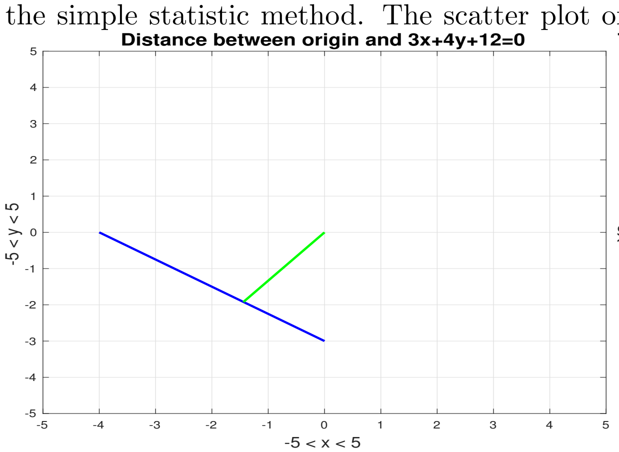


Figure 1 Perpendicular Distance

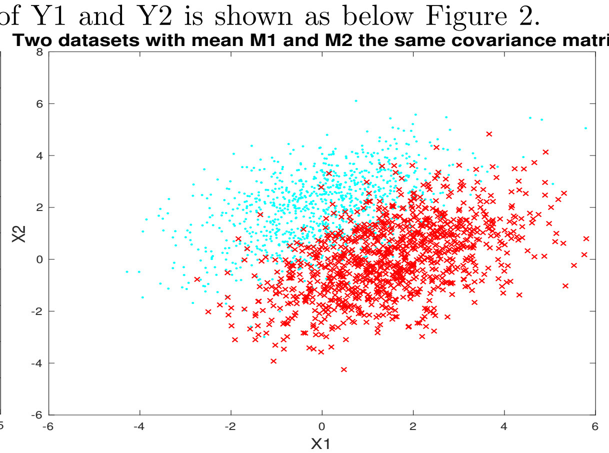
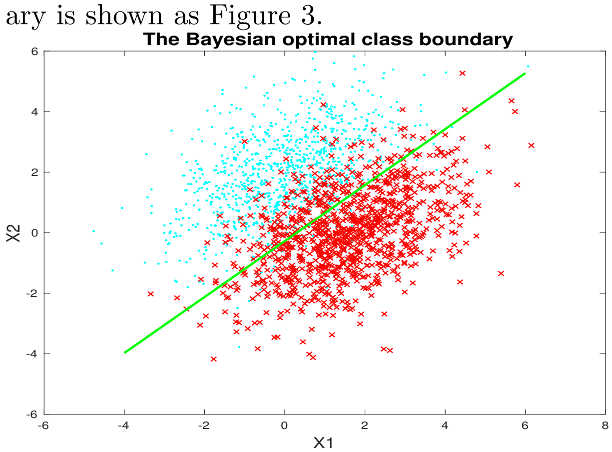


Figure 2 Two different Datasets

2 Exercise 3 & 4

2.1 Bayesian optimal class boundary

The bayesian decision theory is trying to classify the data according to the posterior probabilities and with the assumption that two dataset is contributed to gaussian(with different mean but same covariance matrix), we can reason that it is actually compare the Mahalanobis distance:$(\mathbf{X}-\mathbf{m_1})^{\mathbf{t}}\mathbf{C}^{-1}(\mathbf{X}-\mathbf{m_1})$. by calculating the formula $\mathbf{w} = \mathbf{2C}^{-1}(\mathbf{m_2}-\mathbf{m_1})$, $\mathbf{b} = \mathbf{m_1^t C}^{-1}\mathbf{m_1} - \mathbf{m_2^t C}^{-1}\mathbf{m_2}$ (suppose $\mathbf{p[w1]=p[w2]}$).The Bayesian optimal class boundary is shown as Figure 3.



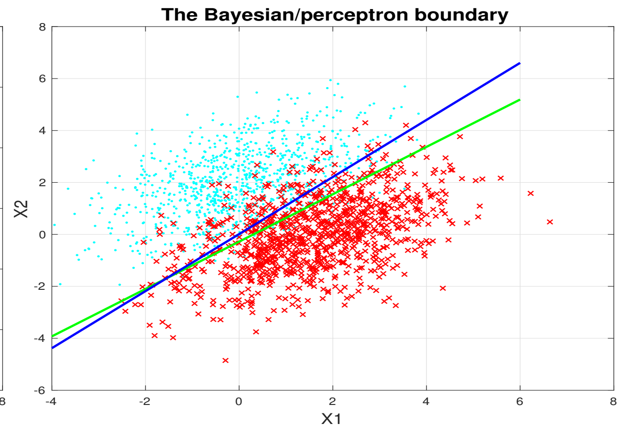
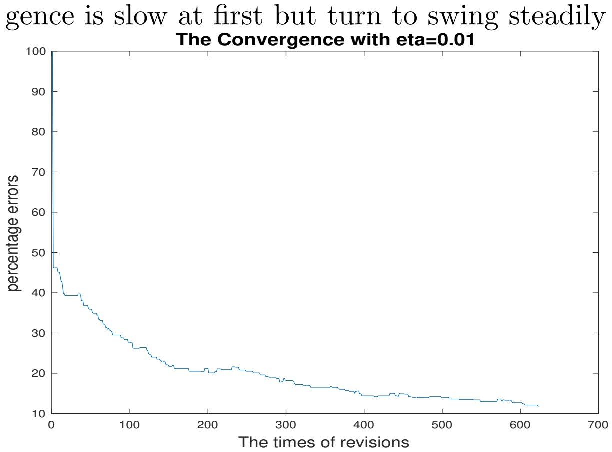


Figure 3 Bayesian boundary **Figure 4 Bayesian/Perceptron boundary**

2.2 Implement of perceptron learning algorithm

The update rule of perceptron learning algorithm is $\mathbf{w} = \mathbf{w} + \mathbf{eta} * \mathbf{ytr}_{[\mathbf{j}]} * \mathbf{Xtr}_{[\mathbf{j},:]}$, the update rule confirm that the revision is in correct direction. Using the perceptron algorithm, we can get a new linear classifier shown in Figure 4 with blue line. The convergence of perceptron algorithm is because $\mathbf{Bk^2} <= \mathbf{w_k} - \mathbf{w_0} <= \mathbf{Ak}$ where k is the times of update,A and B are a certain coefficient. The **eta** is the learning rate.when *eta* is a relative big number the convergence is fast at first but turn to swing drastically when it is close to convergency boundary. when eta is a relative small number the convergence is slow at first but turn to swing steadily when it is close to convergency boundary.



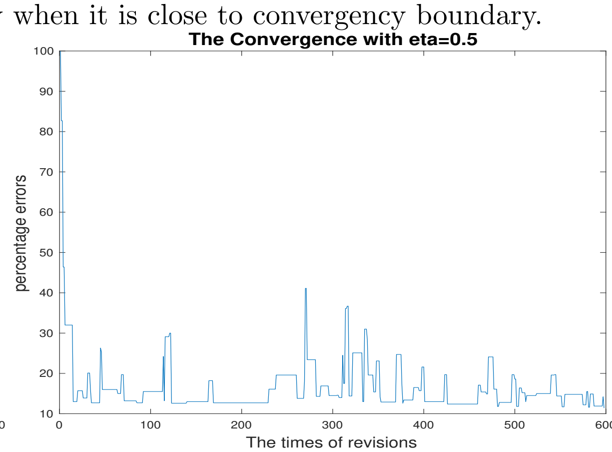


Figure 5 convergency analysis(eta=0.01) **Figure 6 convergency analysis(eta=0.5)**