Brenden Hein

CSE 802

Dr. Ross

Homework 2

1. Consider a set of 1-dimensional feature values…
   a. The histogram with bin size of 2 for that data is…

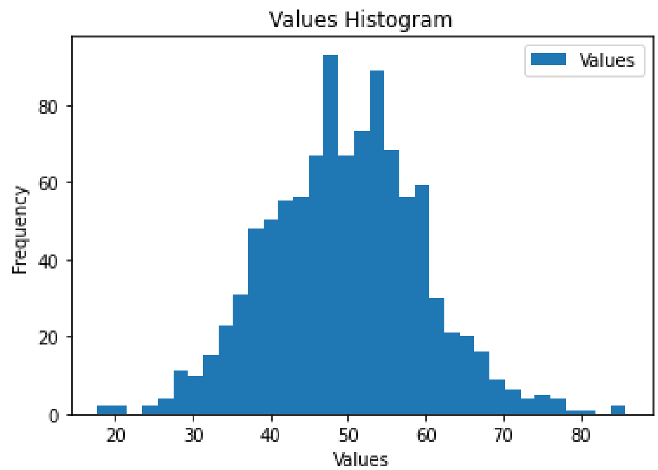

Fig. 1. A frequency of all points in the data set (bin size = 2)

   b. Mean = 49.674 and biased variance = 99.694
   c. The histogram with the pdf from (a) is…

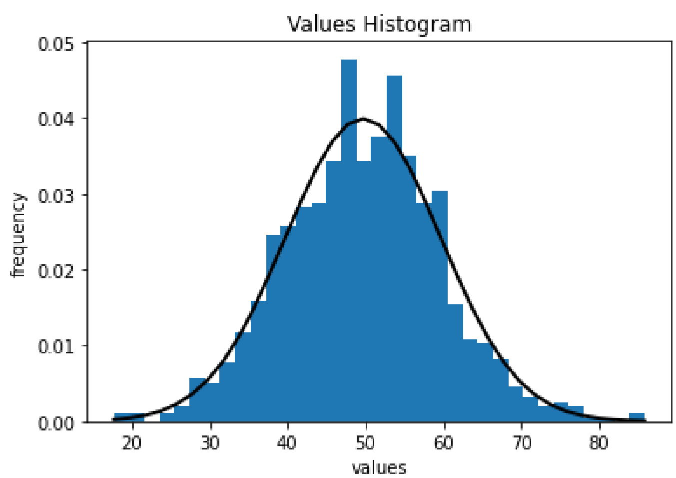

Fig. 2. A frequency of all points in the data set with a normal pdf (bin size = 2)

2. $P(\omega 1) = 0.6$, $P(\omega 2) = 0.4$, $p(x|\omega 1) = 0.2$, $p(x|\omega 2) = 0.4$
    a. $P(x) = 0.2 * 0.6 + 0.4 * 0.4 = 0.28$
       $P(\omega 1|x) = (0.2 * 0.6) / 0.28 = 0.423$
       $P(\omega 1|x) = (0.4 * 0.4) / 0.28 = 0.571$
    b. The class that a fish of 10 inches will be assigned to is **ω2.** Since 0.571 > 0.423, therefore by Bayes decision rule and a 0-1 loss function, the probability of encountering a fish in ω2 is greater and will be selected.
3. $P(\omega 1) = 2/3$, $P(\omega 2) = 1/3$, α1 - choose ω1, α2 - choose ω2, α3 - do not classify,
   $\lambda(\alpha 1 |\omega 1) = \lambda(\alpha 2 |\omega 2) = 0$, $\lambda(\alpha 2 |\omega 1) = \lambda(\alpha 1 |\omega 2) = 1$, $\lambda(\alpha 3 |\omega 1) = \lambda(\alpha 3 |\omega 2) = ¼$

Fig. 3. A plot of p(x|ω1) – red and p(x|ω2) – blue pdfs given x

// Class conditionals
$p(x|\omega 1) = (2-.5)/2 =$ **.75**
$p(x|\omega 2) =$ **.5**

// Scale factor
$P(x) = .75* (2/3) + .5 * (1/3) =$ **.666667**

// Posterior probabilities
$P(\omega 1|x) = (.75 * (2/3)) / .666667 =$ **.75**
$P(\omega 2|x) = (.5 * (1/3) / .666667 =$ **.25**

// Risk functions
$R(\alpha 1|x) = 0 * .75 + 1 * .25 =$ **.25**
$R(\alpha 2|x) = 1 * .75 + 0 * .25 =$ **.75**
$R(\alpha 3|x) = .25 * .75 + .25 * .25 =$ **.25**

// Verify the posterior probability is greater than the threshold defined in question 5
Threshold = 1 - .25/1 = **.75 = .75 (posterior for ω1)**

By the decision rule, $\alpha 1$ is the best action to take. The risk of choosing $\omega 1$ is less than the risk of choosing $\omega 2$. The other condition $(P(\omega 1|x)) \geq 1 - \lambda r/ \lambda s$ results in an equality. Therefore, the risk of rejecting is the same as the risk of choosing $\omega 1$, but based on the threshold defined in question 5, we select the class $\omega 1$ over the reject option.

4. $p(x \mid \omega 1) \sim N(50, 5)$ and $p(x \mid \omega 2) \sim N(30, 10)$
   a. The pdfs corresponding to the 2 distributions are…

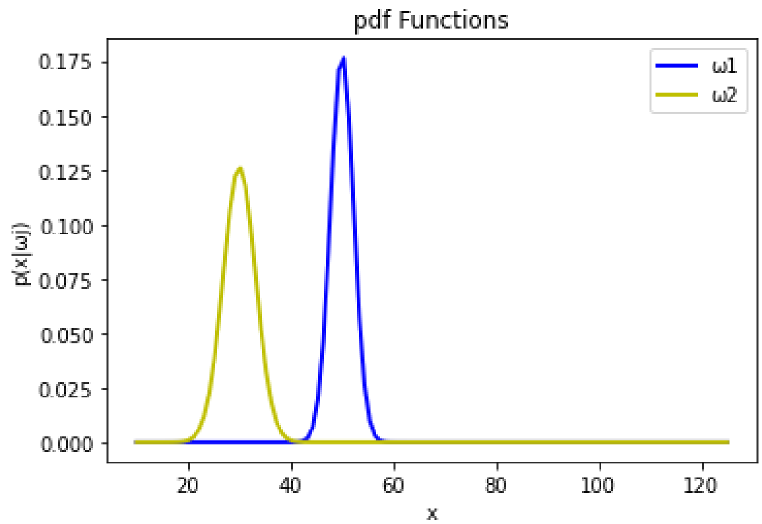

Fig. 4. The corresponding gaussian distributions of N(50, 5) and N(30, 10)

   b. The corresponding plot of the likelihood ratio is...

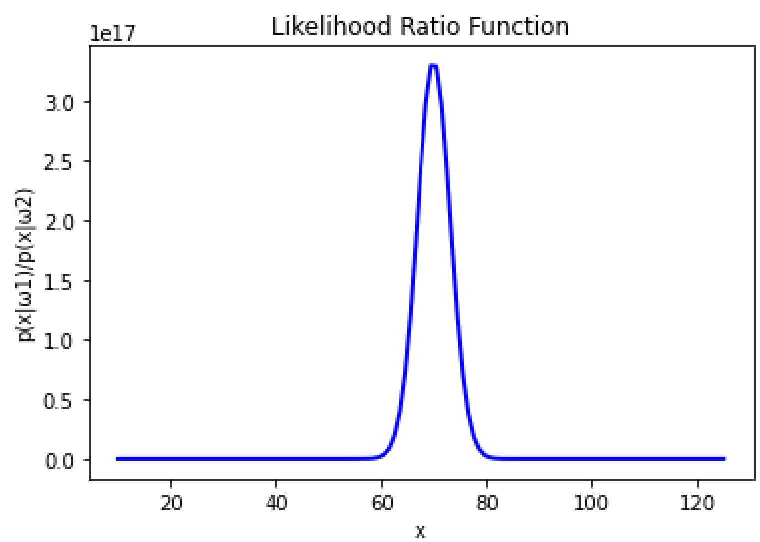

Fig. 5. The plot of the likelihood ratio given $p(x|\omega 1)$ and $p(x|\omega 2)$

c. Threshold = [(2-0) / (1-0)] * (.5/.5) = **2**

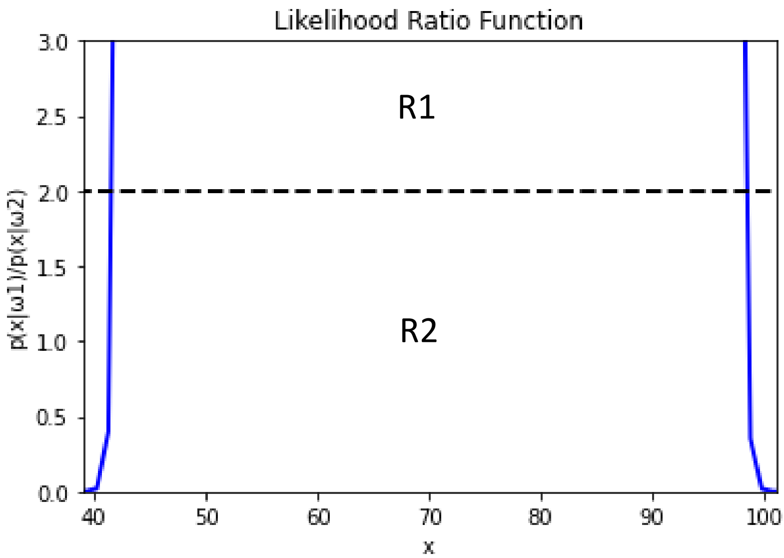

Fig. 5. The plot of the likelihood ratio given p(x|ω1) and p(x|ω2) with the threshold = 2.

d. Threshold = [(1-0) / (2-0)] * (.5/.5) = **.5**

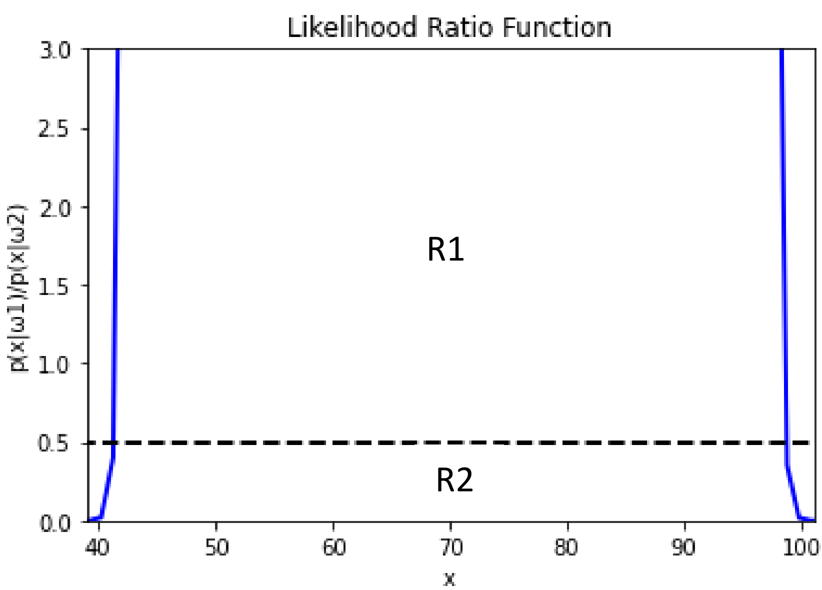

Fig. 6. The plot of the likelihood ratio given p(x|ω1) and p(x|ω2) with the threshold = .5.

e. The reason for the change from 4c to 4d is that, in 4c, the loss of misclassifying the first class was greater than misclassifying the second class, so the threshold was greater. This means that the ratio had to be larger for accurate classification of selecting ω1 (a higher conditional probability for ω1 was required for that class to be selected). In 4d, the loss of misclassifying the second class was greater than misclassifying the first class, so the threshold was smaller. This means that the ratio had to be smaller for accurate classification of selecting ω2 (a higher conditional probability for ω2 was required for that class to be selected).

5. Given a pattern recognition problem with a specified loss function…

a. R(αr|x) = $\sum_{j=1}^{c} \lambda_r * P(\omega j|x)$
R(αr|x) = $\lambda_r \sum_{j=1}^{c} P(\omega j|x)\ \ since\ \lambda_r\ is\ a\ constant$
R(αr|x) = $\lambda_r * 1\ since\ [P(\omega 1|x) +\ P(\omega 2|x) + \cdots P(\omega c|x)] = 1\ for\ a\ given\ x$
R(αr|x) = λ_r

R(αi|X) = $\sum_{j=1}^{c} \lambda_s * P(\omega j|x)$
R(αi|X) = $\sum_{j!=1}^{c} \lambda_s * P(\omega j|x)$
R(αi|X) = $\lambda_s \sum_{j!=i}^{c} P(\omega j|x)$
R(αi|X) = $\lambda_s * [1 - P(\omega i|x)]$

Given that λ_s is a constant, in order to minimize the risk of taking action, αi, we want to maximize $P(\omega j|x)$. Therefore, we assign a pattern to ωi if $\boldsymbol{P(\omega i|x)}$ **≥** $\boldsymbol{P(\omega i|x)}$**∀j.**

In order to select αi…

$$R(\alpha\mathrm{i}|\mathrm{x}) \leq R(\alpha\mathrm{r}|\mathrm{x})$$
$$\lambda_s * [1 - P(\omega i|x)] \leq \lambda_r$$
$$1 - P(\omega i|x) \leq \frac{\lambda_r}{\lambda_s}$$
$$-P(\omega i|x) \leq \frac{\lambda_r}{\lambda_s} - 1$$
$$P(\omega i|x) \geq 1 - \frac{\lambda_r}{\lambda_s}$$

This means that the risk of rejecting the pattern all together must be larger than the risk to classify the pattern ωi, or that $P(\omega i|x) \geq 1 - \frac{\lambda_r}{\lambda_s}$. Therefore, the minimum risk rule follows the decision policy…

$\boldsymbol{P(\omega i|x)}$ **≥** $\boldsymbol{P(\omega i|x)}$**∀j AND** $\boldsymbol{P(\omega i|x) \geq 1 - \frac{\lambda_r}{\lambda_s}}$

b. If $\lambda_r = 0$, the probability of selecting a class given a feature vector of x, $P(\omega i|x)$, must be 1 (as probabilities cannot exceed 1). This intuitive. Rejecting patterns will incur some loss. If there were no loss associated with the reject option, it would appear to be the best option for all patterns where c > 1.

   If $\lambda_r > \lambda_s, \frac{\lambda_r}{\lambda_s} > 1$. This would make the threshold negative: $P(\omega i|x) \geq -T$. As probabilities are fixed on the interval [0, 1], every value of $P(\omega i|x)$ would exceed the reject threshold which would render the rule pointless. As reject options are a safer bet while still incurring some loss, such as folding on a poker hand, having a great loss for reject over misclassifying seems counterintuitive.

6. Considering a 2 class 1 dimensional classification problem where…
   p(x|ω1) = 2 − 2x, x ∈ [0, 1]; p(x|ω2) = 2x, x ∈ [0, 1]

   a. The plot of these 2 density functions is…

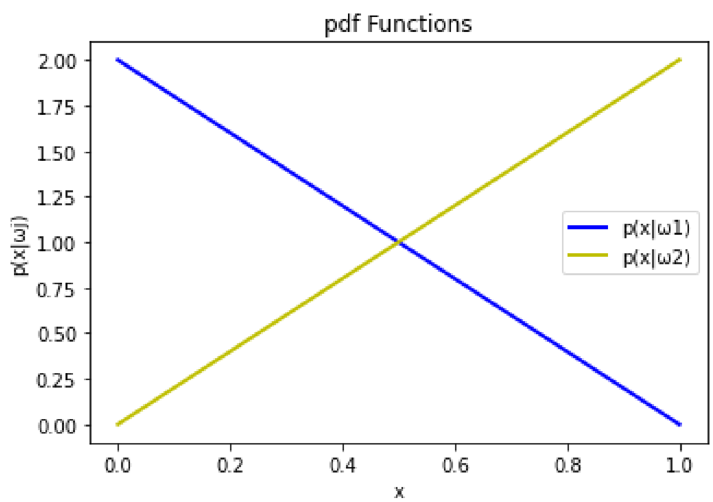

Fig. 7. The pdfs of the two class conditional probability functions (2-2x, 2x)

b. Given that P(ω1) = P(ω2) = ½ and the loss function is 0-1 in a 1-dimensional problem with equal priors, the decision boundary can be found by…
2-2x=2x => 2=4x => **x=1/2**
Bayes decision rule: if $x \leq .5$, decide ω1; else, decide ω2

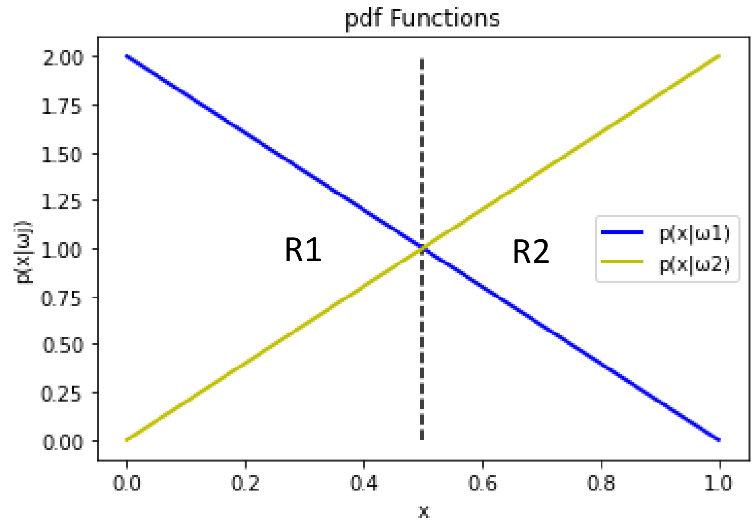

Fig. 7. The pdfs of the two class conditional probability functions with a decision boundary of .5

c. Given that P(ω1) = P(ω2) = ½ and the loss function is λ11 = λ22 = 0, λ12 = 2 and λ21 = 1 in a 1-dimensional problem with equal priors, the decision boundary can be found by using the discriminant function g(x)=R(αi|x)
R(α1|x) = R(α2|x)
λ11 * P(x|ω1) + λ12 * P(x|ω2) = λ21 * P(x|ω1) + λ22 * P(x|ω2) //priors and scale cancel
2 * P(x|ω2) = P(x|ω1) => 2 = P(x|ω1)/ P(x|ω2)

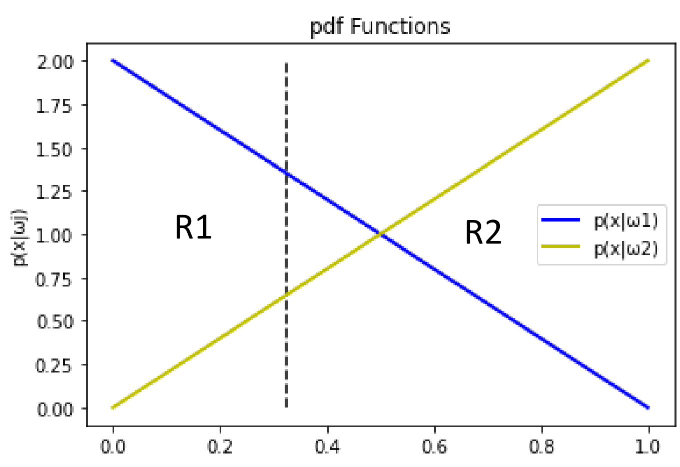

Fig. 8. The pdfs of the two conditional pdfs with a decision boundary of 2 = P(x|ω1)/ P(x|ω2)

d. Boundary b is centered at the intersection point because the priors are equal and the loss function is 0-1, so the cost of misclassification is the same for ω1 and ω2. However, the boundary is shifted to the left for c because the cost of misclassifying ω2 as ω1 is twice as larger as the opposite. Because the loss is greater in the former, the boundary is shifted to the left, so the model attempts reduce the error. It wants to be more careful with misclassifying ω2 as ω1, so if the pattern has a greater conditional probability for ω1 but is close, to play it safe the model will still classify it as ω2.

7. Consider a two-class problem with the following class-conditional probability density functions (pdfs): p(x|ω1) ~ N(0, σ^2) and p(x|ω2) ~ N(1, σ^2)

   a. $\frac{\lambda_{12}-\lambda_{11}}{\lambda_{21}-\lambda_{22}} * \frac{P(\omega 2)}{P(\omega 1)} \quad => \quad \frac{P(x|\omega 1)}{P(x|\omega 2)} > \frac{\lambda_{12}}{\lambda_{21}} * \frac{P(\omega 2)}{P(\omega 1)}$

$$\frac{P(x|\omega 1)}{P(x|\omega 2)} = \frac{e^{[-\frac{1}{2}*\frac{(x^2)}{\sigma^2}]}}{\sqrt{2\pi\sigma^2}} * \frac{\sqrt{2\pi\sigma^2}}{e^{[-\frac{1}{2}*\frac{((x-1)^2)}{\sigma^2}]}}$$

$$\frac{P(x|\omega 1)}{P(x|\omega 2)} = \frac{e^{[-\frac{1}{2}*\frac{(x^2)}{\sigma^2}]}}{e^{[-\frac{1}{2}*\frac{((x-1)^2)}{\sigma^2}]}}$$

$$\frac{P(x|\omega 1)}{P(x|\omega 2)} = \frac{e^{\frac{(-x^2)}{2\sigma^2}}}{e^{\frac{-x^2+2x-1}{2\sigma^2}}}$$

$$\frac{P(x|\omega 1)}{P(x|\omega 2)} = e^{[\frac{-x^2+x^2-2x+1}{2\sigma^2}]}$$

$$\ln [e^{[\frac{-2x+1}{2\sigma^2}]}] = \ln [\frac{\lambda_{12}}{\lambda_{21}} * \frac{P(\omega 2)}{P(\omega 1)}]$$

$$\frac{-2x+1}{2\sigma^2} = \ln [\frac{\lambda_{12}}{\lambda_{21}} * \frac{P(\omega 2)}{P(\omega 1)}]$$

$$-2x - 1 = 2\sigma^2 * \ln [\frac{\lambda_{12}}{\lambda_{21}} * \frac{P(\omega 2)}{P(\omega 1)}]$$

$$\tau = \frac{1}{2} - \sigma^2 * \ln [\frac{\lambda_{12}}{\lambda_{21}} * \frac{P(\omega 2)}{P(\omega 1)}]$$

8. Consider the three-dimensional normal distribution p(x) ~ N(μ,Σ)

a. $|\Sigma| = \left|\begin{bmatrix}1 & 0 & 0\\ 0 & 5 & 2\\ 0 & 2 & 5\end{bmatrix}\right| = 1*(5*5-2*2)-0*(0*5-2*0)-0*(0*2-5*0)$ = **21**

b. $\Sigma^{-1} = \frac{1}{21}\begin{bmatrix}5*5-2*2 & -(0*5-0*2) & 0*2-0*5\\ -(0*5-2*0) & 1*5-0*0 & -(1*2-0*0)\\ 0*2-5*0 & -(1*2-0*0) & 1*5-0*0\end{bmatrix} = \begin{bmatrix}1 & 0 & 0\\ 0 & \frac{5}{21} & -\frac{2}{21}\\ 0 & -\frac{2}{21} & \frac{5}{21}\end{bmatrix}$

c. $\lambda * \mathrm{I} = \begin{bmatrix}\lambda & 0 & 0\\ 0 & \lambda & 0\\ 0 & 0 & \lambda\end{bmatrix}$

$$\det(A-\lambda*\mathrm{I}) = \det\left(\begin{bmatrix}1-\lambda & 0 & 0\\ 0 & 5-\lambda & 2\\ 0 & 2 & 5-\lambda\end{bmatrix}\right) = (1-\lambda)*\big((5-\lambda)*(5-\lambda)-4\big)$$

$$= (1-\lambda)*(21-10\lambda+\lambda^2) = (1-\lambda)*(\lambda-7)*(\lambda-3) => \boldsymbol{\lambda = 7, 1, 3}$$

$\begin{bmatrix}1-7 & 0 & 0\\ 0 & 5-7 & 2\\ 0 & 2 & 5-7\end{bmatrix} = \begin{bmatrix}-6 & 0 & 0\\ 0 & -2 & 2\\ 0 & 2 & -2\end{bmatrix}$ = B

$\begin{bmatrix}1-1 & 0 & 0\\ 0 & 5-1 & 2\\ 0 & 2 & 5-1\end{bmatrix} = \begin{bmatrix}0 & 0 & 0\\ 0 & 4 & 2\\ 0 & 2 & 4\end{bmatrix}$ = C

$\begin{bmatrix}1-3 & 0 & 0\\ 0 & 5-3 & 2\\ 0 & 2 & 5-3\end{bmatrix} = \begin{bmatrix}-2 & 0 & 0\\ 0 & 2 & 2\\ 0 & 2 & 2\end{bmatrix}$ = D

B * x = 0:

$\begin{bmatrix}-6 & 0 & 0\\ 0 & -2 & 2\\ 0 & 2 & -2\end{bmatrix}\begin{bmatrix}x1\\ x2\\ x3\end{bmatrix} = \begin{bmatrix}0\\ 0\\ 0\end{bmatrix}$

$r1 = -\frac{1}{6}*r1\text{:}\ \begin{bmatrix}1 & 0 & 0\\ 0 & -2 & 2\\ 0 & 2 & -2\end{bmatrix}\begin{bmatrix}x1\\ x2\\ x3\end{bmatrix} = \begin{bmatrix}0\\ 0\\ 0\end{bmatrix}$

$r2 = -\frac{1}{2}*r2\text{:}\ \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & -1\\ 0 & 2 & -2\end{bmatrix}\begin{bmatrix}x1\\ x2\\ x3\end{bmatrix} = \begin{bmatrix}0\\ 0\\ 0\end{bmatrix}$

$r3 = -2*r2+r3\text{:}\ \begin{bmatrix}1 & 0 & 0\\ 0 & 1 & -1\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}x1\\ x2\\ x3\end{bmatrix} = \begin{bmatrix}0\\ 0\\ 0\end{bmatrix}$

x1 = 0, x2 = x3 => **v1 = [0, 1, 1]**

C * x = 0:

$\begin{bmatrix}0 & 0 & 0\\ 0 & 4 & 2\\ 0 & 2 & 4\end{bmatrix}\begin{bmatrix}0\\ 0\\ 0\end{bmatrix}$

$r2 = \frac{1}{2}*r2\text{:}\ \begin{bmatrix}0 & 0 & 0\\ 0 & 2 & 1\\ 0 & 2 & 4\end{bmatrix}\begin{bmatrix}x1\\ x2\\ x3\end{bmatrix} = \begin{bmatrix}0\\ 0\\ 0\end{bmatrix}$

$r3 = \frac{1}{2}*r3\text{:}\ \begin{bmatrix}0 & 0 & 0\\ 0 & 2 & 1\\ 0 & 1 & 2\end{bmatrix}\begin{bmatrix}x1\\ x2\\ x3\end{bmatrix} = \begin{bmatrix}0\\ 0\\ 0\end{bmatrix}$

x3 = -2x2, x2 = -2x3 (only true when both are 0)

x1=x1 since it is not dependent on system of equations

v2 = [1, 0, 0]

D * x = 0:

$$\begin{bmatrix}-2 & 0 & 0\\0 & 2 & 2\\0 & 2 & 2\end{bmatrix}\begin{bmatrix}x1\\x2\\x3\end{bmatrix}=\begin{bmatrix}0\\0\\0\end{bmatrix}$$

$$r2=\frac{1}{2}r2\begin{bmatrix}-2 & 0 & 0\\0 & 1 & 1\\0 & 2 & 2\end{bmatrix}\begin{bmatrix}x1\\x2\\x3\end{bmatrix}=\begin{bmatrix}0\\0\\0\end{bmatrix}$$

$$r3=\frac{1}{2}r3\begin{bmatrix}-2 & 0 & 0\\0 & 1 & 1\\0 & 1 & 1\end{bmatrix}\begin{bmatrix}x1\\x2\\x3\end{bmatrix}=\begin{bmatrix}0\\0\\0\end{bmatrix}$$

$$r3=r2-r3\begin{bmatrix}-2 & 0 & 0\\0 & 1 & 1\\0 & 0 & 0\end{bmatrix}\begin{bmatrix}x1\\x2\\x3\end{bmatrix}=\begin{bmatrix}0\\0\\0\end{bmatrix}$$

-2x1=0 => x1=0, x2 = -x3 => **v3 = [0, -1, 1]**

$$\boldsymbol{eiganvector}:\begin{bmatrix}\mathbf{0} & \mathbf{1} & \mathbf{0}\\\mathbf{1} & \mathbf{0} & \mathbf{-1}\\\mathbf{1} & \mathbf{0} & \mathbf{1}\end{bmatrix}\ \boldsymbol{and\ eiganvalues}:[\mathbf{7},\mathbf{1},\mathbf{3}]$$

d. The density at $(0,0,0)^t$ is **0.0072850**
   The density at $(5,5,5)^t$ is **4.727065e-7**
e. The Euclidean distance between the points between…

$$\sqrt{(5-0,5-0,5-0)^t*(5-0,5-0,5-0)}=\mathbf{8.66}$$

f. The Mahalanobis distance between the points between…

$$\sqrt{\begin{bmatrix}5 & 5 & 5\end{bmatrix}*\begin{bmatrix}1 & 0 & 0\\0 & \frac{5}{21} & -\frac{2}{21}\\0 & -\frac{2}{21} & \frac{5}{21}\end{bmatrix}*\begin{bmatrix}5\\5\\5\end{bmatrix}}$$

$$\sqrt{\begin{bmatrix}5 & \frac{15}{21} & \frac{15}{21}\end{bmatrix}*\begin{bmatrix}5\\5\\5\end{bmatrix}}=\sqrt{25+\frac{75}{21}+\frac{75}{21}}=\mathbf{5.67}$$

Appendix

Q1:

```
import numpy as np
import pandas as pd
from scipy.stats import norm
import matplotlib.pyplot as plt
import math

def plot_histo(data, mu, std):
    """Plots a histogram of the data points from the 1-d feature vector"""
    data = data["Values"]

    # Finds the bin range
    mini = min(data)
    maxi = max(data)
    ran = math.ceil(maxi) - math.floor(mini)
    binsize = ran // 2 + ran % 2
    plt.hist(data, bins=binsize, density=1)

    # Gets the pdf
    x = np.linspace(mini, maxi, binsize)  # the x-axis points
    p = norm.pdf(x, mu, std)  # the corresponding y points for the pdf
    plt.plot(x, p, 'k', linewidth=2)

    plt.xlabel("values")
    plt.ylabel("frequency")
    plt.title("Values Histogram")


def mean_histo(data):
    """Finds the mean and biased variance of the data points"""
    mu = data["Values"].mean()
    var = data["Values"].var(ddof=0)
    print("Mean:", mu)
    print("Variance:", var)
    return mu, var ** (1 / 2)


def main():
    fp = open("hw02_data01.txt")
    data = []
    for line in fp:
        data.append(float(line.strip()))
    data = pd.DataFrame(data, columns=["Values"])

    mu, std = mean_histo(data)
    plot_histo(data, mu, std)


if __name__ == "__main__":
    main()
```

Q4:

```
import numpy as np
from scipy.stats import norm
import matplotlib.pyplot as plt

def plot_pdf():
    """This function, given a mean and standard deviation, plots the
    corrosponding pdf on the interval [10, 75]"""
    fig, ax = plt.subplots()
    mu1, mu2 = 50, 30
    std1, std2 = 5 ** (1 / 2), 10 ** (1 / 2)

    # Plot the 2 pdf functions
    x = np.linspace(10, 125, 115)
    pdf1 = norm.pdf(x, mu1, std1)
    pdf2 = norm.pdf(x, mu2, std2)
    ax.plot(x, pdf1, color='b', linewidth=2, label="\u03c91")
    ax.plot(x, pdf2, color='y', linewidth=2, label="\u03c92")
    ax.legend()
    ax.set_xlabel("x")
    ax.set_ylabel("p(x|\u03c9j)")
    ax.set_title("pdf Functions")


def plot_liklihood(t=False, t_val=0):
    """This fuction, given a mean and standard deviation, plots the
    corrosponding liklihood functions on the interval [10, 75]"""
    fig, ax = plt.subplots()
    mu1, mu2 = 50, 30
    std1, std2 = 5 ** (1 / 2), 10 ** (1 / 2)
    x = np.linspace(10, 125, 115)

    # Finds pdfs
    pdf1 = list(norm.pdf(x, mu1, std1))
    pdf2 = list(norm.pdf(x, mu2, std2))

    # Finds the liklihood ratio for distributions
    like = []
    for i in range(len(pdf1)):
        like.append(pdf1[i] / pdf2[i])

    # Plots the liklihood ratio
    ax.plot(x, like, color='b', linewidth=2)
    ax.set_xlabel("x")
    ax.set_ylabel("p(x|\u03c91)/p(x|\u03c92)")
    ax.set_title("Likelihood Ratio Function")

    if t:
        ax.plot(x, [t_val for i in range(len(pdf1))], "--", color="k", linewidth=2)
        ax.set_ylim(0, 3)
        ax.set_xlim(39, 101)


def main():
    plot_pdf()
    plot_liklihood()
    plot_liklihood(True, 2)
    plot_liklihood(True, .5)


if __name__ == "__main__":
    main()
```

Q6:

```python
import numpy as np
from scipy.stats import norm
import matplotlib.pyplot as plt
import pprint as pp


def plot_pdf(d=None, df=False):
    """This function plots the pdfs
    of two different class conditional functions"""
    # Gets the point for the 2 pdfs
    x = np.linspace(0, 1, 100)
    w1 = [2 - 2 * p for p in x]
    w2 = [2 * p for p in x]

    # Plots the pdfs
    fig, ax = plt.subplots()
    ax.plot(x, w1, color='b', linewidth=2, label="p(x|\u03c91)")
    ax.plot(x, w2, color='y', linewidth=2, label="p(x|\u03c92)")
    ax.legend()

    # Plots a decison boundary
    if d is not None:
        ax.vlines(d, 0, 2, colors='k', linestyles='--')

        # Plots the deciosin boundary function
    if df:
        for i in range(len(w1)):
            if w1[i] / w2[i] < 2:
                break
        b = x[i - 1]
        ax.vlines(b, 0, 2, colors='k', linestyles='--')

    # Details figure
    ax.set_xlabel("x")
    ax.set_ylabel("p(x|\u03c9j)")
    ax.set_title("pdf Functions")


def main():
    plot_pdf()
    plot_pdf(d=.5)
    plot_pdf(df=True)


if __name__ == "__main__":
    main()
```

Q8:

```
import scipy.stats as sp
import scipy.spatial as sc
import numpy as np


def calc_density(v, mu, covariance):
    """Calculates the density at a specific point"""
    d = sp.multivariate_normal.pdf(v, mu, covariance)
    print("density:", d)
    return d


def calc_euc(v1, v2, mu, covariance):
    """Calculates the eucidean distance between 2 points"""
    eud = sc.distance.euclidean(v1, v2)
    print("distance:", eud)
    return eud


def calc_mah(v1, v2, mu, covariance):
    """Calculates the mahalanobis distance on a gaussian distribution"""
    diff = [v2[i] - v1[i] for i in range(len(v1))]

    inv = np.linalg.inv(covariance)
    diff = np.array(diff)
    diff_t = np.array([[v] for v in diff])

    dist = np.dot(diff, inv)
    dist = np.dot(dist, diff_t)
    print("distance:", dist[0] ** .5)
    return dist[0]


def main():
    covariance = [[1, 0, 0],
                  [0, 5, 2],
                  [0, 2, 5]]
    mu = (1, 1, 1)

    calc_density([0, 0, 0], mu, covariance)
    calc_density([5, 5, 5], mu, covariance)
    print()

    calc_euc([0, 0, 0], [5, 5, 5], mu, covariance)
    calc_mah([0, 0, 0], [5, 5, 5], mu, covariance)


if __name__ == "__main__":
    main()
```