ECE M146 - HW 1

John Rapp Farnes | 405461225

1.

We have

$$\begin{cases} P(\text{red} \mid \text{first urn}) = \frac{4}{10} \\ P(\text{blue} \mid \text{first urn}) = \frac{3}{10} \\ P(\text{white} \mid \text{first urn}) = \frac{3}{10} \\ P(\text{red} \mid \text{second urn}) = \frac{2}{10} \\ P(\text{blue} \mid \text{second urn}) = \frac{4}{10} \\ P(\text{white} \mid \text{second urn}) = \frac{4}{10} \\ P(\text{first urn}) = \frac{4}{10} \\ P(\text{second urn}) = 1 - P(\text{first urn}) = \frac{6}{10} \end{cases}$$

a)

$$\begin{aligned} &P(\text{both red}) \\ &= P(\text{first red} \cap \text{second red}) \\ &= P(\text{first urn})P(\text{first red} \cap \text{second red} \mid \text{first urn}) \\ &+ P(\text{second urn})P(\text{first red} \cap \text{second red} \mid \text{second urn}) \\ &= \frac{4}{10}P(\text{first red} \mid \text{first urn})P(\text{second red} \mid \text{first red} \cap \text{first urn}) \\ &+ \frac{6}{10}P(\text{first red} \mid \text{second urn})P(\text{second red} \mid \text{first red} \cap \text{second urn}) \\ &= \frac{4}{10}\frac{4}{10}\frac{3}{9} + \frac{6}{10}\frac{2}{10}\frac{1}{9} \\ &= \frac{1}{15} = 6.\bar{6}6\% \end{aligned}$$

b)

$$
\begin{aligned}
&P(\text{second blue})\\
&= P(\text{first blue})P(\text{second blue} \mid \text{first blue}) + P(\text{first not blue})P(\text{second blue} \mid \text{first not blue})\\
&= P(\text{first urn})[P(\text{first blue} \mid \text{first urn})P(\text{second blue} \mid \text{first blue} \cap \text{first urn})\\
&+ P(\text{first not blue} \mid \text{first urn})P(\text{second blue} \mid \text{first not blue} \cap \text{first urn})]\\
&+ P(\text{second urn})[P(\text{first blue} \mid \text{second urn})P(\text{second blue} \mid \text{first blue} \cap \text{second urn})\\
&+ P(\text{first not blue} \mid \text{second urn})P(\text{second blue} \mid \text{first not blue} \cap \text{second urn})]\\
&= \frac{4}{10}[\frac{3}{10}\frac{2}{9} + \frac{7}{10}\frac{3}{9}] + \frac{6}{10}[\frac{4}{10}\frac{3}{9} + \frac{6}{10}\frac{4}{9}]\\
&= \frac{9}{25} = 36\%
\end{aligned}
$$

c)

$$
\begin{aligned}
&P(\text{second blue} \mid \text{first red})\\
&= P(\text{second blue} \mid \text{first red} \cap \text{first urn})P(\text{first urn} \mid \text{first red})\\
&+ P(\text{second blue} \mid \text{first red} \cap \text{secod urn})P(\text{secod urn} \mid \text{first red})\\
&= \frac{3}{9}\frac{P(\text{first red} \mid \text{first urn})P(\text{first urn})}{P(\text{first red})} + \frac{4}{9}[1 - P(\text{first urn} \mid \text{first red})]\\
&= \frac{3}{9}\underbrace{\frac{\frac{4}{10}\cdot\frac{4}{10}}{\frac{4}{10}\frac{4}{10} + \frac{6}{10}\frac{2}{10}}}_{=\frac{4}{7}} + \frac{4}{9}[\underbrace{1 - P(\text{first urn} \mid \text{first red})}_{=\frac{3}{7}}]\\
&= \frac{3}{9}\frac{4}{7} + \frac{4}{9}\frac{3}{7}\\
&= \frac{8}{21} \approx 38.1\%
\end{aligned}
$$

By Bayes' theorem.

2.

$$P(\text{three different numbers each appear twice when 6 identical dice thrown})$$
$$= \frac{\underbrace{\binom{6}{3}}_{\text{N.o. ways to pick 3 different numbers from 6}} \times \underbrace{\binom{6}{2}\binom{4}{2}\binom{2}{2}}_{\text{N.o. ways to order 3 numbers in 6 positions s.t. there are 2 each of them}}}{\underbrace{6^6}_{\text{N.o. combinations of 6 dice in 6 positions}}}$$
$$= \frac{20 \times 90}{46656} = \frac{25}{648} \approx 3.9\%$$

3.

We have

$$\begin{cases} P(A) = 25\% \\ P(B) = 35\% \\ P(C) = 40\% \\ P(\text{defective} \mid A) = 5\% \\ P(\text{defective} \mid B) = 4\% \\ P(\text{defective} \mid C) = 2\% \end{cases}$$

Which implies

$$\begin{aligned} P(\text{defective}) &= P(A)P(\text{defective} \mid A) + P(B)P(\text{defective} \mid B) + P(C)P(\text{defective} \mid C) \\ &= 25\% \cdot 5\% + 35\% \cdot 4\% + 40\% \cdot 2\% \\ &= \frac{69}{2000} = 3.45\% \end{aligned}$$

As such, by Bayes' Theorem:

$$\begin{cases} P(A \mid \text{defective}) = \frac{P(\text{defective} \mid A)P(A)}{P(\text{defective})} = \frac{25\% \cdot 5\%}{3.45\%} = \frac{25}{69} \\ P(B \mid \text{defective}) = \frac{P(\text{defective} \mid B)P(B)}{P(\text{defective})} = \frac{35\% \cdot 4\%}{3.45\%} = \frac{28}{69} \\ P(C \mid \text{defective}) = \frac{P(\text{defective} \mid C)P(C)}{P(\text{defective})} = \frac{40\% \cdot 2\%}{3.45\%} = \frac{16}{69} \end{cases}$$

4.

a)

$$\begin{aligned}\mathbb{E}[X+Y] &= \sum_{j=0}^{n_1}\sum_{k=0}^{n_2}(x_j+y_k)P(X=x_j,Y=y_k)\\ &= \sum_{j=0}^{n_1}\sum_{k=0}^{n_2}x_jP(X=x_j,Y=y_k)+\sum_{j=0}^{n_1}\sum_{k=0}^{n_2}y_kP(X=x_j,Y=y_k)\\ &= \sum_{j=0}^{n_1}x_jP(X=x_j)+\sum_{k=0}^{n_2}y_kP(Y=y_k)\\ &= \mathbb{E}[X]+\mathbb{E}[Y]\end{aligned}$$

As $\sum_{k=0}^{n_2} x_jP(X=x_j,Y=y_k)=x_jP(X=x_j)$ and vice versa

b)

$$\begin{aligned}var[X+Y] &= \mathbb{E}[(X+Y)^2]-\mathbb{E}[X+Y]^2\\ &= \mathbb{E}[X^2+2XY+Y^2]-[\mathbb{E}[X]+\mathbb{E}[Y]]^2\\ &= \mathbb{E}[X^2]+2\mathbb{E}[XY]+\mathbb{E}[Y^2]-\mathbb{E}[X]^2-2\mathbb{E}[X]\mathbb{E}[Y]-\mathbb{E}[Y]^2\\ &= \underbrace{\mathbb{E}[X^2]-\mathbb{E}[X]^2}_{var[X]}+\underbrace{\mathbb{E}[Y^2]-\mathbb{E}[Y]^2}_{var[Y]}+2\underbrace{(\underbrace{\mathbb{E}[XY]}_{=\mathbb{E}[X]\mathbb{E}[Y]}-\mathbb{E}[X]\mathbb{E}[Y])}_{=0}\\ &= var[X]+var[Y]\end{aligned}$$

As X, Y indepedent $\implies \mathbb{E}[XY]=\mathbb{E}[X]\mathbb{E}[Y]$ as $P(X=x_j,Y=y_k)=P(X=x_j)(Y=y_k)$, proof similar to a)

5.

We have $P(T\le t)=1-e^{-\lambda t}=F(t)$

a)

$$
\begin{aligned}
&P(\text{not arrive in d more seconds}) \\
&= P(T > r + d \mid T > r) \\
&= \frac{P(T > r + d \cap T > r)}{P(T > r)} \\
&= \frac{P(T > r + d)}{P(T > r)} \\
&= \frac{1 - F(r + d)}{1 - F(r)} \\
&= \frac{e^{-\lambda(r+d)}}{e^{-\lambda r}} \\
&= e^{-\lambda d} = 1 - F(d) = P(T > d)
\end{aligned}
$$

This is a result of the memorylessness property of the exponential distribution.

b)

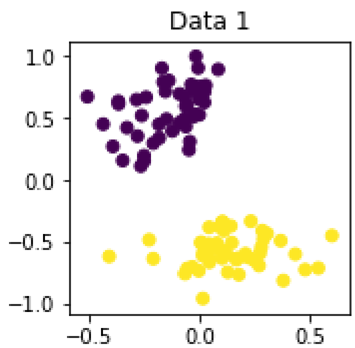
We have cumulative distribution function $F(t) \implies$ probability density function $f(t) = F'(t) = \lambda e^{-\lambda t}, t >= 0$

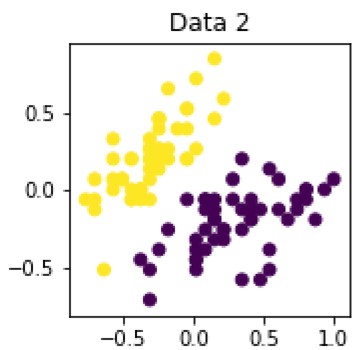
$$
\begin{aligned}
\mathbb{E}[T] &= \int_0^\infty t f(t) dt \\
&= \int_0^\infty t \lambda e^{-\lambda t} dt \\
&= -\big[te^{-\lambda t}\big]_0^\infty + \int_0^\infty e^{-\lambda t} dt \\
&= -\big[\underbrace{\lim_{t\to\infty} te^{-\lambda t}}_{=0} - \underbrace{\lim_{t\to 0} te^{-\lambda t}}_{=0}\big] + \big[-\frac{e^{-\lambda t}}{\lambda}\big]_0^\infty \\
&= \frac{1}{\lambda}(\underbrace{\lim_{t\to\infty} -e^{-\lambda t}}_{=0} + \underbrace{\lim_{t\to 0} e^{-\lambda t}}_{=1}) \\
&= \frac{1}{\lambda}
\end{aligned}
$$

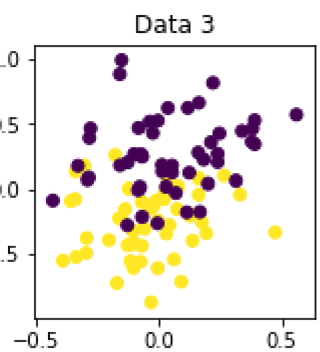
6.

a)

In [3]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

filenames = ["data1.csv", "data2.csv", "data3.csv"]

datas = list(map(lambda f : pd.read_csv(f, header=None, names = ["x1", "x2",
"y"]), filenames))

for i in range(len(datas)):
    d = datas[i]
    plt.figure(i, figsize=[2.6, 2.6])
    plt.scatter(d["x1"], d["x2"], c = d["y"])
    plt.title(f"Data {i + 1}")
```





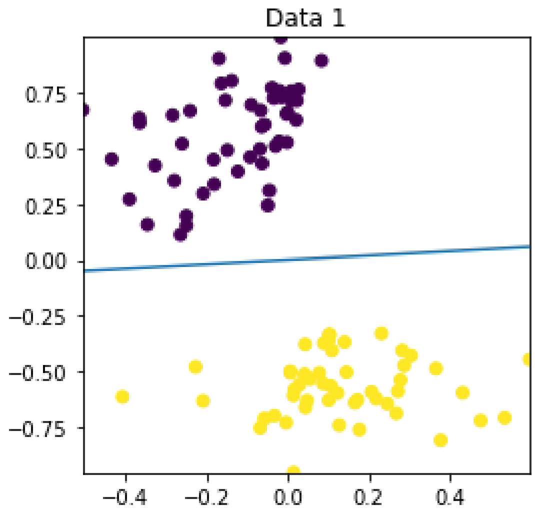


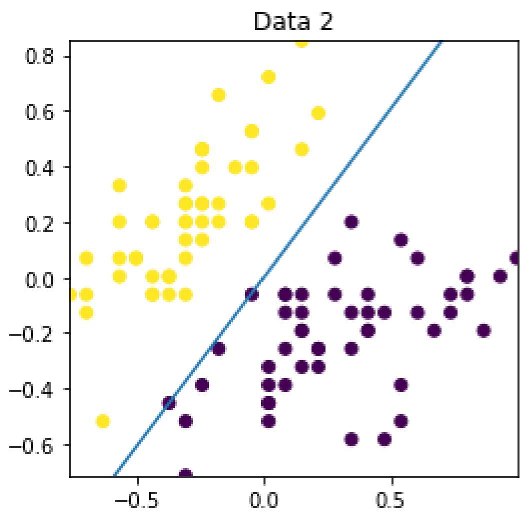
The first two datasets are linearly separable, the third is not.

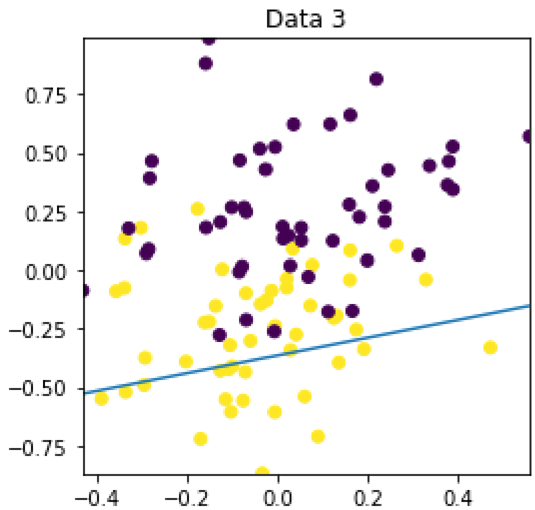
b)

In [2]:
```
MAX_ITERATIONS = 1000
def perceptron(X, y):
    N = X.shape[0]
    X = np.c_[np.ones(N), X]
    w = np.array([0, 0, 0])
    u = 0

    for iter in range(1, MAX_ITERATIONS):
        miss_found = False
        for i in range(N):
            if y[i] * np.dot(w, X[i]) <= 0:
                w = w + y[i] * X[i]
                u = u + 1
                miss_found = True
        if not miss_found:
            break;
    return w, u

def plot_perceptron(X, y, w):
    plt.scatter(X[:, 0], X[:, 1], c = y)

    x1 = [min(X[:, 0]), max(X[:, 0])]
    x2 = [0, 0]
    epsilon = 0.001
    for i in range(2):
        x2[i] = -np.dot(w, [1, x1[i], 0])/(epsilon if w[2] == 0 else w[2])

    plt.plot(x1, x2)
    plt.xlim(*x1)
    plt.ylim(min(X[:, 1]), max(X[:, 1]))

def margin(X, y, w):
    N = X.shape[0]
    X = np.c_[np.ones(N), X]
    margin = float("inf")
    for i in range(N):
        margin = min(margin, y[i] * np.dot(w, X[i]))
    return margin

for i in range(len(datas)):
    d = datas[i]
    X = np.transpose([d["x1"], d["x2"]])
    y = np.array(d["y"])

    w, u = perceptron(X, y)
    m = margin(X, y, w)
    plt.figure(i, figsize=[4,4])
    plt.title(f"Data {i + 1}")
    plot_perceptron(X, y, w)

    print(f"------ Data {i + 1} ------")
    print(f"w = {w}")
    print(f"u = {u}")
    print(f"margin = {m}")
    print(f"Convergence bound <= {int(1 / m**2)}")
```

```
------ Example 1 ------
w = [ 0.         0.142172 -1.47323 ]
u = 2
margin = 0.20257509974
Convergence bound <= 24
------ Example 2 ------
w = [ 0.       -1.10918  0.91344]
u = 4
margin = 0.00038062299999996885
Convergence bound <= 6902556
------ Example 3 ------
w = [-1.         1.0376823 -2.738021 ]
u = 4497
margin = -1.891782400105088
Convergence bound <= 0
```







As all but data 3 are linearly separable, the perceptron algorithm successfully finds a decision boundary that divides the data into the correct classes for them. Based on u, the algorithm converges faster for data 1 than data 2 (as it requires fewer updates), while data 3 doesn't converge (as it runs all the iterations and still does not successfully separate the data).

c)

As seen in the printouts, data 1 has a bigger margin than data 2, and hence a lower convergence bound. This aligns with the difference in amount of updates required for the algorithm. This can also be seen in the plots, where the amount b or w can change is smaller for Data 2, as the classes are "tighter together". The number of updates for both of the first data sets are far lower than the convergence bound. This is probably because our calculated $\gamma_{w,b}$ is smaller than the theoretical γ, giving us a bigger bound $\frac{1}{\gamma_{w,b}^2}$, as well as being "lucky" with the order the data was processed in.

As data 3 is not linearly separable and the algorithm doesn't find a vector w s.t. $y_i \cdot wx_i > 0 \, \forall i$, the margin is negative.