Homework 7

Due November 12 at 11 pm

1. (Spider on a wall) There's a spider living on a wall of your living room that has a painting behind which the spider likes to hide. Figure 1 shows a diagram of the wall; it is 10 feet high and 10 feet wide.

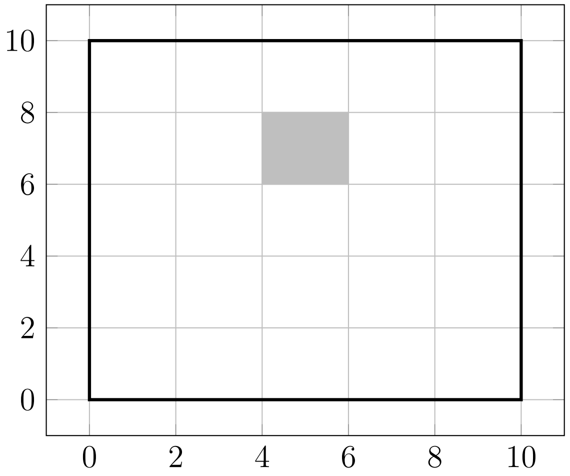


Figure 1: Wall and painting (in gray).

After observing the spider for a while you determine that (1) it spends twice the time behind the painting than on the rest of the wall, (2) it never crawls on the painting or leaves the wall, (3) if it is not behind the painting then it is equally likely to be anywhere on the wall. Since you cannot see it behind the painting, you assume that when it is there it is also equally likely to be at any spot.

(a) Model the position of the spider as a bivariate random variable and give its pdf.

We use $\tilde{x}, \tilde{y}$ to model the position of the spider. Then according to the problem statement we have:

$$\int_4^6 \int_6^8 f_{\tilde{x},\tilde{y}}(x,y)dydx = 2\left(1 - \int_4^6 \int_6^8 f_{\tilde{x},\tilde{y}}(x,y)dydx\right) \tag{1}$$

$$\iint_{(x,y)\notin \text{ Painting}} f_{\tilde{x},\tilde{y}}(x,y)dydx + \iint_{(x,y)\in \text{ Parting}} f_{\tilde{x},\tilde{y}}(x,y)dydx = 1 \tag{2}$$

By assumption, we have $f_{\tilde{x},\tilde{y}}(x,y) = \begin{cases} a, & (x,y) \in (4,6] \times [6,8] \\ b, & (x,y) \in [0,10] \times (0,10] \cap (x,y) \notin [4,6] \times [6,8]. \\ 0, & \text{otherwise.} \end{cases}$

Plugging into the formula (1) we get:

$$\int_4^6 \int_6^8 adydx = 2\left(1 - \int_4^6 \int_6^8 adydx\right)$$
$$4a = 2(1 - 4a)$$
$$a = \frac{1}{6}$$

Plugging into formula (2) we get:

$$(10 \times 10 - 2 \times 2)b + 4a = 1$$
$$b = \frac{1}{288}$$

Thus

$$f_{\tilde{x},\tilde{y}}(x, y) = \begin{cases} \frac{1}{6}, & (x, y) \in [4, 6] \times [6 \times 8] \\ \frac{1}{288}, & (x, y) \in [0, 10] \times [0, 10] \cap (x, y) \notin [4, 6] \times [6, 8]. \\ 0, & \text{otherwise.} \end{cases}$$

(b) Compute the pdf of the height at which the spider is located and sketch it.

The pelf of the height should be the marginal distribution of $\tilde{y}$.

$$\begin{aligned} f_{\tilde{y}}(y) &= \int_0^{10} f_{\tilde{x},\tilde{y}}(x,y)dx \\ &= \begin{cases} \int_0^{10} \frac{1}{288}dx = \frac{5}{144} & ,y \in [0,6] \cup [8,10] \\ \int_0^4 \frac{1}{288}dx + \int_6^{10} \frac{1}{288}dx + \int_4^6 \cdot \frac{1}{6}dx = \frac{13}{36}, & y \in [6,8] \\ 0 & , \text{ otherwise} \end{cases} \end{aligned}$$

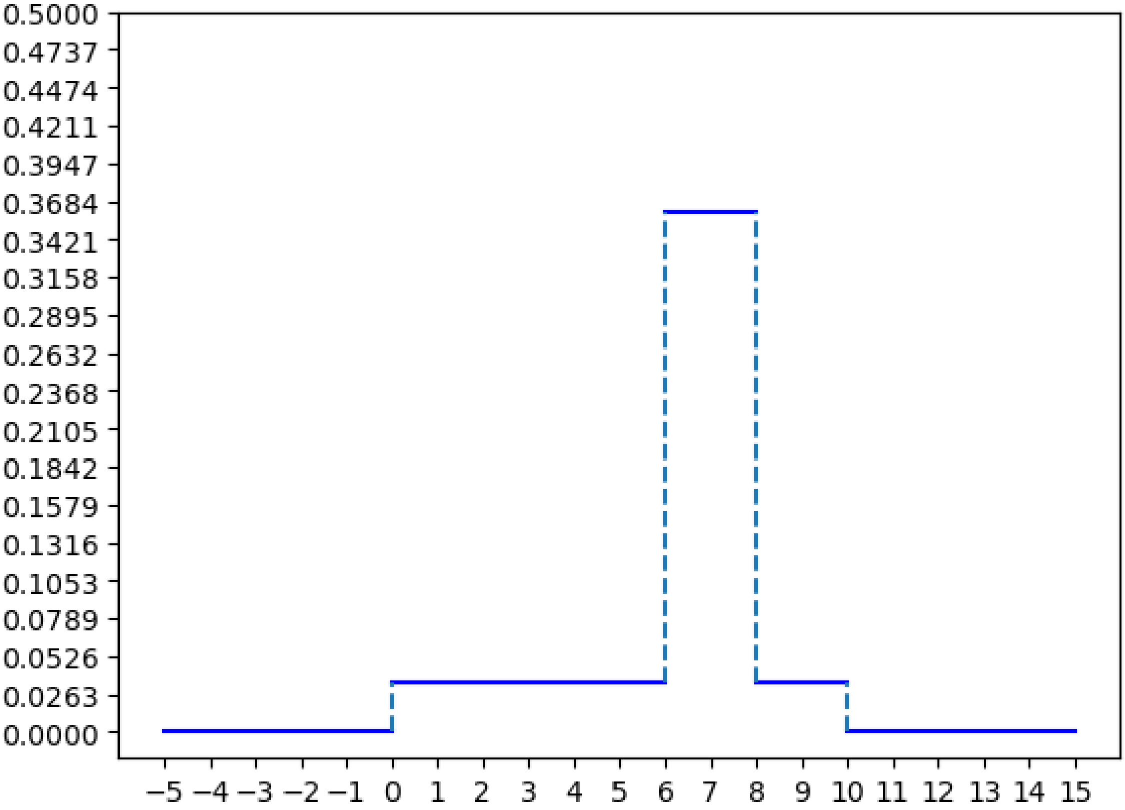


Figure 2: Marginal PMF Matrix

Attched is the link to the code(click it): Jupyter notebook link

```
# Problem 1(b)
x_1 = np.linspace(-5,0,100)
x_2 = np.linspace(0, 6,100)
x_3 = np.linspace(6, 8,100)
x_4 = np.linspace(8, 10, 100)
x_5 = np.linspace(10,15, 100)

plt.plot(x_1, 0 * np.ones(len(x_1)), c = "b", linestyle="-")
plt.plot(x_2, 5 / 144 * np.ones(len(x_2)), c = "b", linestyle="-")
plt.plot(x_3, 13 / 36 * np.ones(len(x_3)), c = "b", linestyle="-")
plt.plot(x_4, 5 / 144 * np.ones(len(x_4)), c = "b", linestyle="-")
plt.plot(x_5, 0 * np.ones(len(x_5)), c = "b", linestyle="-")
plt.vlines(0, ymin = 0, ymax = 5 /144, linestyle="--")
plt.vlines(6, ymin = 5 / 144 , ymax = 13 / 36, linestyle="--")
plt.vlines(8, ymin = 5 / 144, ymax = 13 / 36, linestyle="--")
plt.vlines(10, ymin = 0, ymax = 5 /144, linestyle="--")
plt.yticks(np.linspace(0,0.5,20))
plt.xticks(np.linspace(-5,15,21))
plt.show()
```

(c) Compute the conditional cdf of the height at which the spider is located, given that you can see it (i.e. it's not under the painting) and sketch it.

Denote the event A : The spider isn't under the painting; then the conditional CDF of interest could be written as:

$$F_{\tilde{y}|A}(y \mid A) = \frac{P(\tilde{y} \leq y \cap A)}{P(A)} \text{ where } P(A) = 1 - \int_4^6 \int_6^8 f_{\tilde{x},\tilde{y}}(x,y)dydx = \frac{1}{3}$$

$$= \begin{cases} 0 & , y \in (-\infty, 0) \\ \frac{P(\tilde{y} \leq y)}{P(A)} = \frac{\int_0^y y(\tilde{y}(y)}{1/3} = \frac{5}{48}y & , y \in (0,6) \\ \frac{P(\tilde{y} \leq y) - \int_4^6 \int_6^y \frac{1}{288}dydx}{1/3} = \frac{1}{12}y + \frac{1}{8}, & y \in [6,8) \\ \frac{P(\tilde{y} \leq y) - \int_4^6 \int_6^8 \frac{1}{288}dydx}{1/3} = \frac{5}{48}y - \frac{1}{24}, & y \in [8,10) \\ 1 & y \in [10, \infty) \end{cases}$$

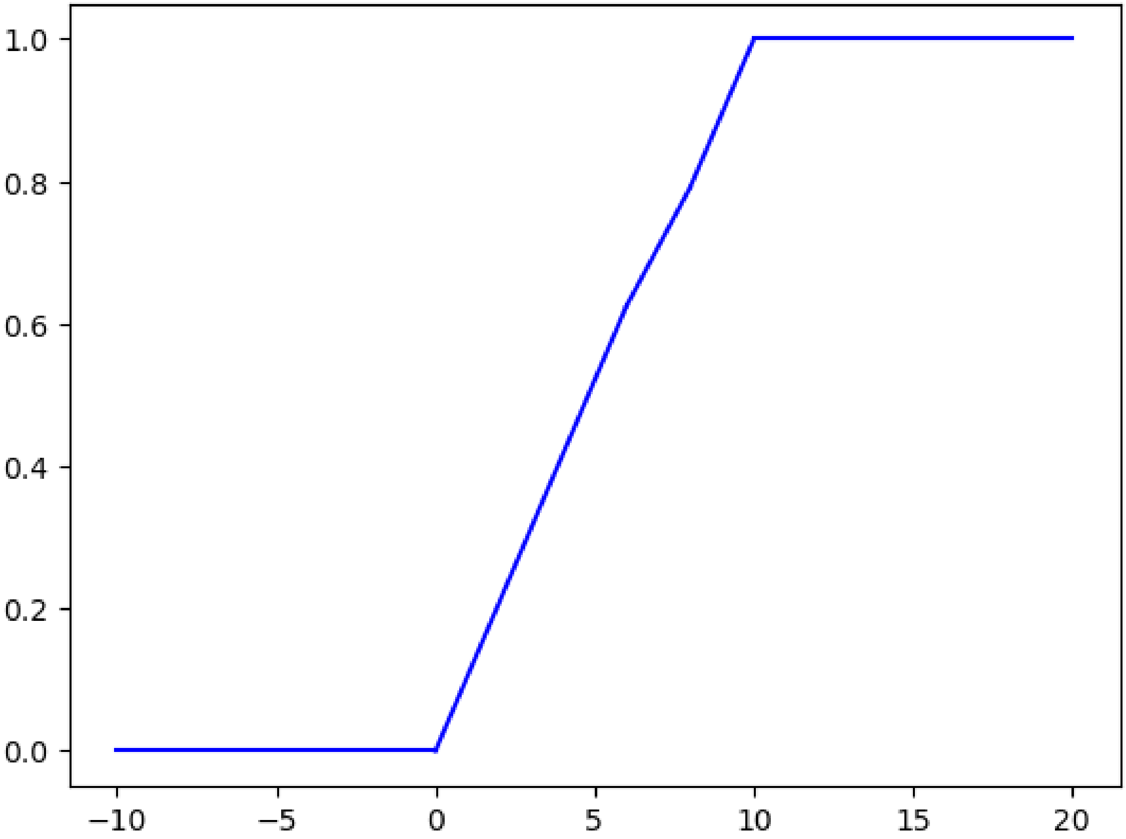


Figure 3: Marginal PMF Matrix

Attched is the link to the code(click it): Jupyter notebook link

```
# Problem 1(c)
def generate_piece_wise_function(v):

    if v == 0:
        return lambda x:0
    elif v == 1:
        return lambda x: (5 / 48) * x
    elif v == 2:
        return lambda x: (1 / 12) * x + 1 / 8
    elif v == 3:
        return lambda x: (5 / 48) * x - 1 / 24
    else:
        return lambda x: 1


def plot_piece_wise_function(pieces):

    for index, piece in enumerate(pieces):
        func = generate_piece_wise_function(index)
        plt.plot(piece, [func(y) for y in piece],c = "blue")

    plt.show()


pieces = [np.linspace(-10,0)
        ,np.linspace(0,6)
        ,np.linspace(6,8)
        ,np.linspace(8,10)
        ,np.linspace(10,20)]

plot_piece_wise_function(pieces)
```

2. (Sonar) A scientist is trying to determine the depth of the sea at a certain location. She knows that it must be deeper than 5 km but that is all she knows. To capture this uncertainty she models the depth as uniformly distributed between 5 and 10 km. In order to measure the depth she uses sonar, taking 2 measurements, which we model as two random variables $_1$ and $_2$. If the depth is equal to x then each sonar measurement is uniformly distributed between $x - 0.25$ and $x + 0.25$. The two measurements are conditionally independent given the depth.

   (a) Compute and sketch the pdf of the first sonar measurement $_1$.

   Denote $\tilde{x}$ with $f_{\tilde{x}} = \begin{cases} \frac{1}{5}, & x \in [5, 10] \\ 0, & \text{otherwise.} \end{cases}$ Denote $\tilde{m}_1$ as measurement $1, \tilde{m}_2$ as measurement 2 . Then by problem statement we have:

$$f_{\tilde{m}_1 \mid \tilde{x}}\left(m_1 \mid x\right) = \begin{cases} 2, & m_1 \in [x - 0.25, x + 0.25] \\ 0, & \text{otherwise.} \end{cases}$$

$$f_{\tilde{m}_2 \mid \tilde{x}}\left(m_2 \mid x\right) = \begin{cases} 2, & m_2 \in [x - 0.25, x + 0.25] \\ 0, & \text{otherwise.} \end{cases}$$

$$f_{\tilde{m}_1, \tilde{m}_2 \mid \tilde{x}}\left(m_1, m_2 \mid x\right) = f_{\tilde{m}_1 \mid \tilde{x}}\left(m_1 \mid x\right) f_{\tilde{m}_2 \mid \tilde{x}}\left(m_2 \mid x\right).$$

   In order to compute $f_{\tilde{m}_1}\left(m_1\right)$ (Similarly for $\tilde{m}_2$), we have:

$$f_{\tilde{m}_1}\left(m_1\right) = \int_5^{10} f_{\tilde{m}_1 \mid \tilde{x}}\left(m_1 \mid x\right) f_{\tilde{x}}(x) \cdot dx$$

$$= \begin{cases} \int_{m_1 - 0.25}^{m_1 + 0.25} 2 \times \frac{1}{5} dx = \frac{1}{5}, & 5.25 \leqslant m_1 \leqslant 9.75 \\ \int_5^{m_1 + 0.25} 2 \times \frac{1}{5} dx = \frac{2}{5}\left(m_1 - 4.75\right), & 4.75 \leqslant m_1 \leqslant 5.25 \\ \int_{m_1 - 0.25}^{10} 2 \times \frac{1}{5} dx = \frac{2}{5}\left(10.25 - m_1\right), & 9.75 \leqslant m \leqslant 10.25 \\ 0 & otherwise. \end{cases}$$

(b) Compute the conditional pdf of the depth conditioned on the measurements being equal to 7 km and 7.1 km.

The conditional probability of interest is:

$$\begin{aligned} f_{\tilde{x}|\tilde{m}_1,\tilde{m}_2}(x \mid 7, 7.1) &= \frac{f_{\tilde{m}_1,\tilde{m}_2|\tilde{x}}(7, 7.1 \mid x) f_{\tilde{x}}(x)}{f_{\tilde{m}_1,\tilde{m}_2}(7, 7.1)} \\ &= \frac{f_{\tilde{m}_1|\tilde{x}}(7 \mid x) \cdot f_{\tilde{m}_2|\tilde{x}}(7.1 \mid x) f_{\tilde{x}}(x)}{\int_5^{10} f_{\tilde{m}_1|\widetilde{x}}(7 \mid x) f_{\tilde{m}_2|\tilde{x}}(7.1 \mid x) f_{\tilde{x}}(x) dx} \\ &= \begin{cases} 0 & , x \in [5, 6.85) \\ \frac{2\times 2\times \frac{1}{5}}{\int_{6.85}^{7.25} 2\times 2\times \frac{1}{5} dx} = \frac{4/5}{4/5\times 0.4} = 2.5, & x \in [6.85, 7.25) \\ 0, & x \in [7.25, 10] \end{cases} \end{aligned}$$

(c) Compute the joint pdf of the two sonar measurements $_1$ and $_2$. Are the two measurements independent? Justify your answer mathematically and explain it intuitively.

$$f_{\tilde{m}_1,\tilde{m}_2}\left(m_1, m_2\right) = \int_5^{10} f_{\tilde{m}_1|\tilde{x}}\left(m_1 \mid x\right) f_{\tilde{m}_2|\tilde{x}}\left(m_2 \mid x\right) f_{\tilde{x}}(x)dx.$$

$$= \begin{cases} 0, \quad |m_1 - m_2| \geqslant 0.5 \cup \min\left(10, \min\left(m_1, m_2\right) + 0.25\right) < \max\left(5, \max\left(m_1, m_2\right) - 0.25\right). \\ \int_{\max(m_1,m_2)-0.25}^{\min(m_1,m_2)+0.25} f_{\tilde{m}_1,|\tilde{x}}\left(m_1 \mid x\right) f_{\tilde{m}2}|\tilde{x}\left(m_2 \mid x\right) f_{\tilde{x}}(x)dx, \quad | m_1 - m_2 | < 0.5 \end{cases}$$

If we pick $m_1 = 6, m_2 = 8$, then

$$f_{m_1}\left(m_1\right) = f_{\tilde{m}_2}\left(m_2\right) = \frac{1}{5}$$

However $f_{\tilde{m}_1,\tilde{m}_2}(6, 8) = 0$

$$\therefore f_{\tilde{m}_1}\left(m_1\right) f_{\tilde{m}_2}\left(m_2\right) \neq f_{\tilde{m}_1}, \tilde{m}_2\left(m_1, m_2\right)$$

$\therefore$ Mathematically specking, $\tilde{m}_1, \tilde{m}_2$ aren't independent. Intuitively speaking, if we know the measurement 1, we will know that the depth will likely be within the range of $(m_1 - 0.25, m_1 + 0.25)$. Then this depth will give clues to the range of values within which m_2 would likely be.

3. (Exotic fruit) A pomologist is studying a newly-discovered plant that produces edible fruit. They have available 10 specimens, which she measures to obtain the following data:

Fruit weight (kg)	1.5	2.3	0.8	1.2	2.0	1.2	0.7	2.7	2.3	0.6
Stem height (cm)	20	15	18	19	17	22	21	14	17	22
Stem radius (mm)	8	12	6	10	17	12	9	14	10	8
Number of leafs	110	94	152	123	78	60	111	83	85	90
Median leaf length (cm)	12	21	9	14	19	15	7	29	22	15

The pomologist wants to use these data to model the conditional pdf of the fruit weight given the other four features.

(a) Why is it difficult to estimate the conditional pdf using a simple nonparametric estimator where we apply kernel density estimation to a relevant subset of the data?

Denote the conditional probability that we want to estimate by $f_{\tilde{\omega}|\tilde{h},\tilde{r},\tilde{l},\tilde{m}}(\omega \mid h, r, l, m)$, where $\widetilde{\omega}$ corresponds to weight, $\tilde{h}$ to height, $\tilde{r}$ to radius, $\widetilde{l}$ to # of leafs, $\tilde{m}$ to median leaf length. Non parametric model falls victims to curse of dimensionality. In this example, in order to estimate that conditional probability, we have to know the joint distribution of $(\tilde{h}, \tilde{r}, \tilde{l}, \tilde{m})$. This is tractable when each random variable only takes finite values. But $\tilde{h}, \tilde{r}, \tilde{l}, \tilde{m}$ are all continuous RV's. thus estimating $f_{\tilde{h},\tilde{r},\tilde{l},\tilde{m}}$ is just impossible given that we only have finite amount of data.

(b) Model all the features as jointly Gaussian and use the model to compute the conditional pdf of the fruit weight for a plant that has a stem height of 15 cm, a stem radius of 20 mm, 120 leafs and median leaf length of 8 cm.

Denote $(\widetilde{w}, \tilde{h}, \tilde{r}, \tau, \widetilde{m}) = \widetilde{\vec{x}} \in R^5$, which follows multivariate gaussian distribution with pal:

$$f_{\vec{x}}(\vec{x}) = \frac{1}{(2\pi)^{\frac{5}{2}} |\Sigma_5|} \exp\left\{-(\vec{x}-\vec{\mu})^\top \Sigma_5^{-1} (\vec{x}-\vec{\mu})\right\}$$

$$\vec{\mu}_{ML} = \frac{1}{n}\sum_{i=1}^{n} \vec{x}_i$$

$$\Sigma_{ML} = \frac{1}{n}\sum_{i=1}^{n} (\vec{x}_i - \vec{\mu}_{ML})(\vec{x}_i - \vec{\mu}_{ML})^\top$$

We then compute the conditional pdf of $\vec{x}[1]$ given $\vec{x}[2:5]$, which is parametrized by;

$$\begin{aligned} \mu_{\text{cond}} &= \mu_{\vec{x}[1]} + \Sigma_{\vec{x}[2:5],\vec{x}[1]} \Sigma_{\vec{x}[2:5]}^{-1} \left(\vec{x}[2:5] - \vec{\mu}_{\vec{x}[2:5]}\right) \\ \Sigma_{\text{cond}} &= \Sigma_{\vec{x}[1]} - \Sigma_{\vec{x}[2:5],\vec{x}(1]}^\top \Sigma_{\vec{x}[2:5]}^{-1} \Sigma_{\vec{x}[2:5],\vec{x}[1]} \end{aligned}$$

By python code we get $\mu_{\text{cond}} = 1.57772945$ and $\Sigma_{\text{cond}} = 0.03747546420415787$ where $\vec{x}[2:5] = [15, 20, 120, 8]^\top$ and get the conditional pdf at this point. Attached is the code and pdf result.

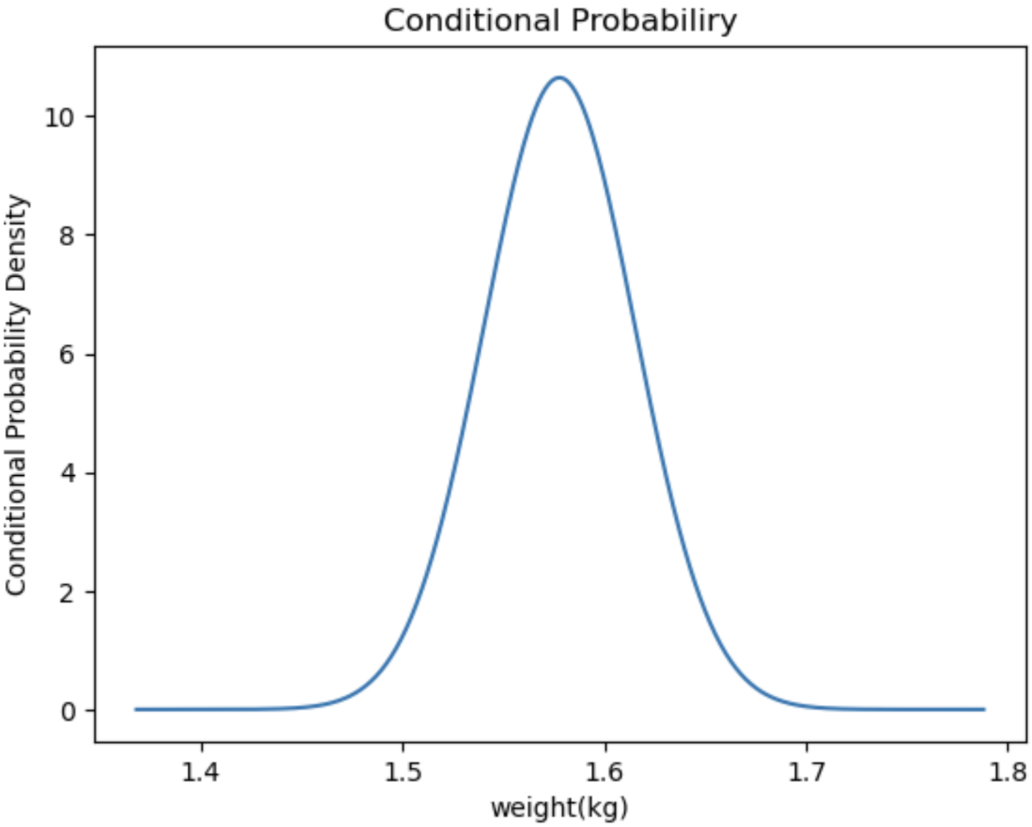
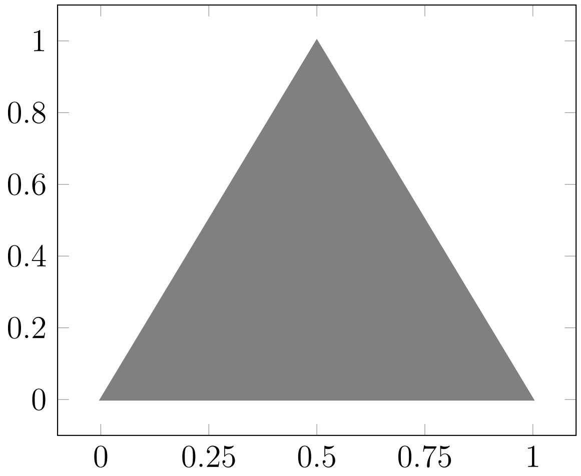


Figure 4: Marginal PMF Matrix

Attched is the link to the code(click it): Jupyter notebook link

```
# Problem 3(b)

# A trick here, we reverse the order of the feature
# , only to apply the multi−dimensional gaussian
# estimate quickly
raw_data = np.array(list(reversed([[1.5, 2.3, 0.8, 1.2, 2.0, 1.2, 0.7, 2.7, 2.3, 0.6]
                    ,[20, 15, 18, 19 , 17 , 22 , 21 , 14 , 17 , 22]
                    ,[8, 12, 6,  10,  17,  12,  9,  14,  10,  8 ]
                    ,[110, 94, 152, 123, 78, 60, 111, 83, 85, 90]
                    ,[12, 21, 9, 14, 19, 15, 7, 29, 22, 15]
                   ])))
data = raw_data.T
n = data.shape[0]
condition_x = np.array([15, 20, 120, 8])


# Mean Vector
mu_MLE = 1 / n * data.sum(axis = 0)
# Covariance matrix
Sigma_MLE = 1 / n * (data − mu_MLE).T @ (data − mu_MLE)


sub_mu_MLE_y = mu_MLE[4:]  # (1,)   y
sub_mu_MLE_x = mu_MLE[:4]  # (4,)   x

sub_Sigma_MLE_y = Sigma_MLE[4:,4:]                    # (4,1)  y
sub_Sigma_MLE_yx = Sigma_MLE[:4,4:]                   # (x,y)
sub_Sigma_MLE_x = np.linalg.inv(Sigma_MLE[:4,:4])    # (4,4)   x

# Compute conditional model, parametrized by mu_cond and sigma_cond
mu_cond = sub_mu_MLE_y + sub_Sigma_MLE_yx.T
              @ sub_Sigma_MLE_x
              @ (condition_x − sub_mu_MLE_x)
sigma_cond = (sub_Sigma_MLE_y − sub_Sigma_MLE_yx.T
              @ sub_Sigma_MLE_x
              @ sub_Sigma_MLE_yx)[0][0]



# Generate the conditional probability
us = np.linspace(1.578 − 0.03 * 7,1.578 + 0.03 * 7,500)
plt.plot(us, norm.pdf(us, mu_cond, sigma_cond))
plt.xlabel(”weight(kg)”)
plt.ylabel(”Conditional␣Probability␣Density”)
plt.title(”Conditional␣Probabiliry”)
plt.show()
```

4. (Simulating a random vector) Explain how to simulate a random two-dimensional vector with a joint pdf $f(x)$ that is uniformly distributed in the shaded region. Assume that you have access to independent samples from a uniform distribution in $[0, 1]$. Explain your method, justifying why it works. Then implement it and submit the code along with a scatterplot of 1,000 samples.



Below is the code to simulate the joint continuous random variables and the results. Attched is the link to the code(click it): Jupyter notebook link

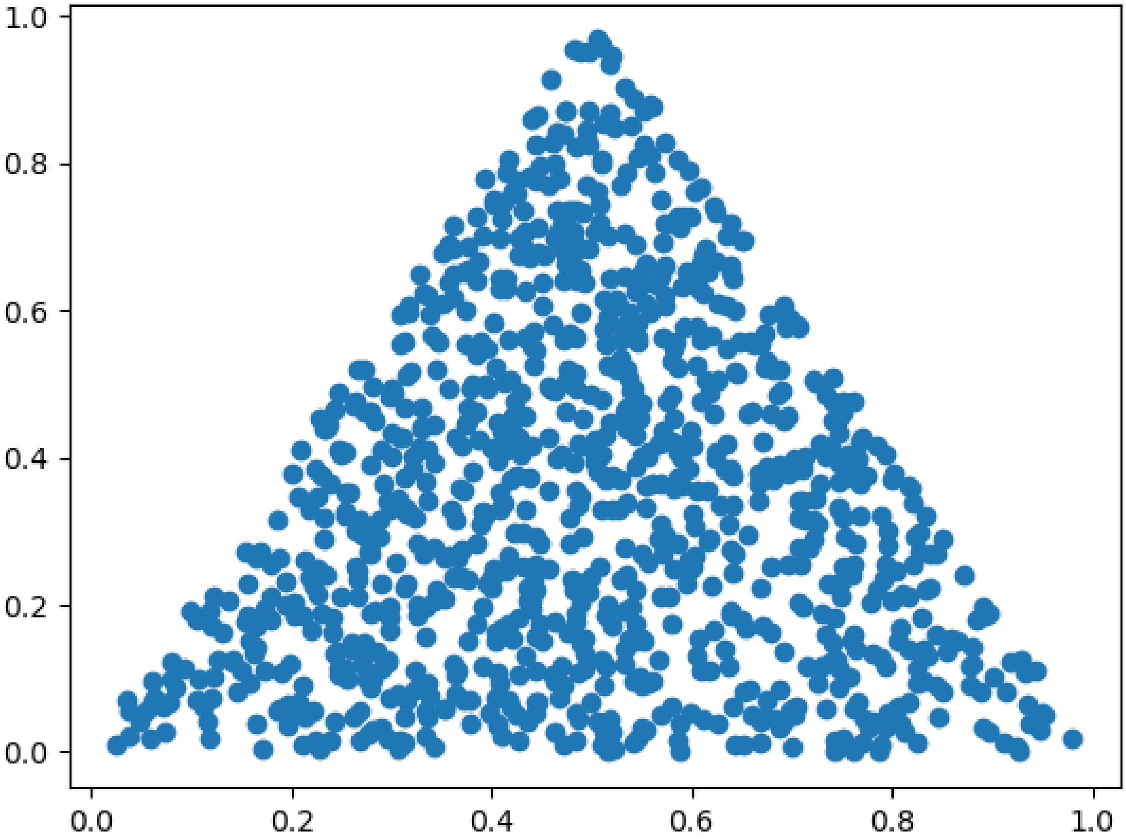


Figure 5: Marginal PMF Matrix

```python
import numpy as np
import scipy.stats
import matplotlib.pyplot as plt
%matplotlib inline

# Generate jointly distributed samples from independent uniform distribution
def generate_one_sample(inverse_CDF_u1, inverse_CDF_u2):
    u1 = generate_u1(inverse_CDF_u1)
    u2 = generate_u2(inverse_CDF_u2, u1)
    return u1, u2

# Just for this problem
def define_inverse_CDF():
    func_u1 = lambda x: 1 - np.sqrt(1 - x)
    func_u2 = lambda x, y: (1 - y) * x + (1 / 2) * y

    return func_u1, func_u2

# Apply Inverse Sampling
def generate_u1(inverse_CDF):
    u = np.random.uniform(0,1,1)
    return inverse_CDF(u)

# Applying Inverse Sampling, Conditional Probability
def generate_u2(inverse_CDF, u1):
    u = np.random.uniform(0,1,1)
    return inverse_CDF(u, u1)

def draw_scatter_plot(data):
    # Since we first generate y, then x.
    plt.scatter(data[:,1], data[:,0])
    plt.show()

def main_procedure(sample_size = 1000):
    sample_points = []
    inverse_CDF_u1, inverse_CDF_u2 = define_inverse_CDF()
    for i in range(sample_size):
        sample_points.append([*generate_one_sample(inverse_CDF_u1, inverse_CDF_u2)])

    data = np.array(sample_points)
    draw_scatter_plot(data)

main_procedure(1000)
```