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

Q1

(a)

```
N = 10000

Ua = np.random.uniform(-1,1,size = N)
Ub = np.random.uniform(-1,1,size = N)

P = sum(2*(Ua**2 + Ub**2) < 3) / N

print('probability:', P)
```

```
probability: 0.9611
```

(b)

In order to estimate $\theta = P(2U_a^2 + 2U_b^2 < 3) = E(Y),\ Y = 1_{\{2U_a^2+2U_b^2<3\}}$.

First, we generate samples of U_a, and compute $g(U_{ai}) := E(Y|U_{ai}) = F_{U_b}\left(\sqrt{1.5 - U_{ai}^2}\right) - F_{U_b}(-\sqrt{1.5 - U_{ai}^2})$.

Estimate θ via the average $\frac{\sum_{i=1}^{n} g(U_{ai})}{n}$.

```
N = 10000

Ua = np.random.uniform(-1,1,size = N)

P = np.mean(uniform.cdf(np.sqrt(1.5-Ua**2), -1, 2) - \
                  uniform.cdf(-np.sqrt(1.5-Ua**2), -1, 2))

print('probability:', P)
```

```
probability: 0.9608750374517444
```

Q2

$$\theta = P(X+Y \geq 4) \quad X \sim \exp(\tfrac{1}{2}),\ Y \sim \exp(\tfrac{1}{3})$$

$$Z = 1_{\{X+Y \geq 4\}}$$

$$E[Z|X=x] = P(X+Y \geq 4 \mid X=x) = P(Y \geq 4-x)$$

$$E[Z|Y=y] = P(X \geq 4-y)$$

$$g(x) = \begin{cases} e^{-\frac{1}{3}(4-x)}, & 0 \leq x \leq 4 \\ 1, & x > 4 \end{cases} \qquad g(y) = \begin{cases} e^{-\frac{1}{2}(4-y)}, & 0 \leq y \leq 4 \\ 1, & y > 4 \end{cases}$$

$$E[g(X)] = \int_0^4 e^{-\frac{4}{3}} e^{\frac{x}{3}} \frac{1}{2} e^{-\frac{x}{2}} dx + \int_4^{+\infty} \frac{1}{2} e^{-\frac{x}{2}} dx$$
$$= 3e^{-\frac{4}{3}} - 3e^{-2} + e^{-2}$$
$$= 3e^{-\frac{4}{3}} - 2e^{-2}$$

$$E[g(Y)] = \int_0^4 e^{-2} e^{\frac{y}{2}} \frac{1}{3} e^{-\frac{y}{3}} dy + \int_4^{+\infty} \frac{1}{3} e^{-\frac{y}{3}} dy$$
$$= 2e^{-\frac{4}{3}} - 2e^{-2} + e^{-\frac{4}{3}}$$
$$= 3e^{-\frac{4}{3}} - 2e^{-2}$$

$$E[g(X)^2] = \int_0^4 e^{-\frac{8}{3}} e^{\frac{2}{3}x} \frac{1}{2} e^{-\frac{x}{2}} + \int_4^{+\infty} \frac{1}{2} e^{-\frac{x}{2}} dx$$
$$= 3e^{-2} - 3e^{-\frac{8}{3}} + e^{-2}$$
$$= 4e^{-2} - 3e^{-\frac{8}{3}}$$

$$E[g(Y)^2] = \int_0^4 e^{-4} e^{y} \frac{1}{3} e^{-\frac{y}{3}} dy + \int_4^{+\infty} \frac{1}{3} e^{-\frac{y}{3}} dy$$
$$= \frac{1}{2} e^{-\frac{4}{3}} - \frac{1}{2} e^{-4} + e^{-\frac{4}{3}}$$
$$= \frac{3}{2} e^{-\frac{4}{3}} - \frac{1}{2} e^{-4}$$

$\because E[g(Y)^2] > E[g(X)^2]$

$\therefore$ we choose condition on X

```
lbdx = 1/2
lbdy = 1/3
N = 10000

X = -np.log(np.random.rand(N)) / lbdx

P = np.mean([min(np.exp(-lbdy*(4-x)),1) for x in X])

print('probability:', P)
```

probability: 0.5217025576925624

Q3

$$\text{For } X_1, \quad \frac{f(x)}{g(x)} = \frac{\lambda_1 e^{-\lambda_1 x}}{\lambda_2 e^{-\lambda_2 x}} = \frac{\lambda_1}{\lambda_2} e^{-(\lambda^1-\lambda^2)x} = 20e^{-0.95x}$$

$$\text{For } X_2, \quad \frac{f(x)}{g(x)} = \frac{\lambda_1 e^{-\lambda_1 x}}{\lambda_2 e^{-\lambda_2 x}} = \frac{\lambda_1}{\lambda_2} e^{-(\lambda^1-\lambda^2)x} = 40e^{-1.95x}$$

Since X_1 *is independent with* X_2

The likelihood ratio is $800e^{-0.95x_1-1.95x_2}$

```
N = 100000
lbdx1 = 1
lbdx2 = 2

lbd = 1/20

Y1 = -np.log(np.random.rand(N)) / lbd
P1 = np.mean((Y1>20) * ((lbdx1/lbd)*np.exp(-(lbdx1-lbd)*Y1)))
Y2 = -np.log(np.random.rand(N)) / lbd
P2 = np.mean((Y2>20) * ((lbdx2/lbd)*np.exp(-(lbdx2-lbd)*Y2)))
P = 1 - (1-P1)*(1-P2)

print('probability:', P)
```

```
probability: 2.073138660740881e-09
```

Q4

```
N = 100000
mu = 5
sigma2 = 2
Y1 = np.random.randn(N) + mu
P1 = np.mean((Y1>5) * np.exp(-mu*Y1+mu**2/2))
Y2 = np.random.randn(N)*np.sqrt(2) + mu
P2 = np.mean((Y2>5) * np.exp(-mu*Y2/sigma2+mu**2/(2*sigma2)))
P = P1*P2

print('probability:', P)
```

```
probability: 5.854763414650687e-11
```