Question 1

1. 对于多元 t 分布 $T_\nu(\mu,\Sigma;p)$，说明其 Copula 完全由矩阵

$$M=\{\mathrm{diag}(\Sigma)\}^{-1/2}\Sigma\{\mathrm{diag}(\Sigma)\}^{-1/2}$$

确定。这里，$\mathrm{diag}(\Sigma)$ 表示由矩阵 Σ 生成的对角矩阵。

记X~$T(\mu,\Sigma;p)$,我们有 $Z=diag(\Sigma)^{-1/2}(X-\mu)$~$T(0,M;p)$,其中 $M=diag(\Sigma)^{-1/2}\Sigma diag(\Sigma)^{-1/2}$

由Lemma1可知X与Z的Copula一致，进而X的Copula完全由矩阵$M=diag(\Sigma)^{-1/2}\Sigma diag(\Sigma)^{-1/2}$确定

Question 2

2. 假设 $(X_1,\ldots,X_p)$ 的 Copula 为 C。令 $T_i:\mathbb{R}\to\mathbb{R}$, $i=1,\ldots,p$, 为严格单增函数。证明：$(T_1(X_1),\ldots,T_p(X_p))$ 的 Copula 仍然是 C.

由$\tilde{F}(z):=P(T(X)\le z)=P(X\le T^{-1}(z))=F\circ T^{-1}(z)$同理有 $\tilde{F}(z_1,\ldots,z_p)=F(T_1^{-1}(z_1),\ldots,T_p^{-1}(z_p))\tilde{F}_i=F_1\circ T_i^{-1}$

进而由Lemma1可知：$(T_1(X_1),\ldots,T_p(X_p))$的Copula为
$\tilde{C}(u_1,\ldots,u_p)=\tilde{F}(\tilde{F}_1^{-1}(u_1),\ldots,\tilde{F}_p^{-1}(u_p))$
$=F(T_1^{-1}(T_1\circ F_1^{-1}(u_1)),\ldots,T_p^{-1}(T_p\circ F_p^{-1}(u_p)))=F(F_1^{-1}(u_1),\ldots,F_p^{-1}(u_p))$
$=C(u_1,\ldots,u_p)$

进而二者Copula相同

```
import numpy as np
import matplotlib.pyplot as plt
```

Question 3

5. 设 $X\sim N(0,1)$, 则 $\theta=P(X>4.5)=3.398\times10^{-6}$.

(1) 如果直接生成 N 个 X 的随机数, 用 $X_i>4.5$ 的比例估计 $P(X>4.5)$, 平均多少个样本点中才能有一个样本点满足 $X_i>4.5$?

(2) 取 V 为指数分布 $\mathrm{Exp}(1)$, 令 $W=V+4.5$, 用 W 的样本进行重要抽样估计 θ, 取样本点个数 $N=1\,000$, 求估计值并估计误差大小.

(1)

使得$\Sigma_i^m I_{\{X_i>4.5\}}\ge 1$的最小的m即为所求，我们知道$\hat{\theta}=\Sigma_i^m I_{\{X_i>4.5\}}/m$,

在大样本情形下，认为$\theta=\hat{\theta}$,进而$m\theta\ge1$当且仅当$m\ge\frac{1}{\theta}\approx 294291$

```
#(2)
N=1000
np.random.seed(1)
V=np.random.exponential(1,N)
W=V+4.5
def g(x):
    return np.exp(-x**2/2+x-4.5)/np.sqrt(2*np.pi)
print(f'对于theta的估计为{g(W).mean():.3e},标准差为{g(W).std():.3e}')
print(f'绝对误差为{g(W).mean()-3.398e-6:.3e},相对误差为{g(W).mean()/3.398e-6-
1:.3e}')
```

```
对于theta的估计为3.398e-06,标准差为4.473e-06
绝对误差为-2.615e-11,相对误差为-7.697e-06
```

Question 4

9. 用随机模拟法计算二重积分 $\int_0^1 \int_0^1 e^{(x+y)^2}\, dy\, dx$，用对立变量法改善精度.

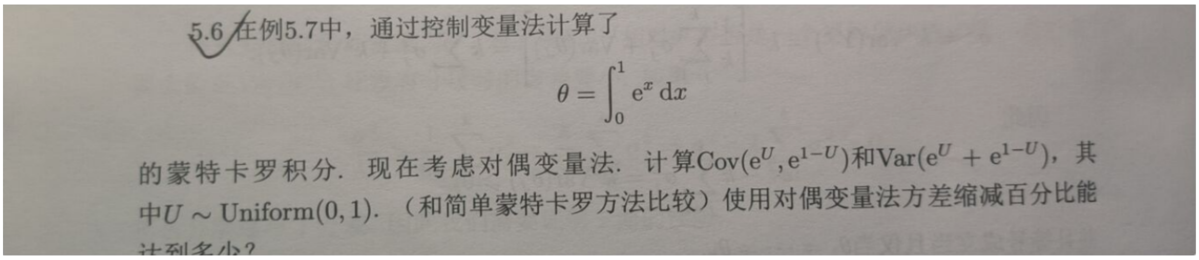
```
def f(x,y):
    return np.exp((x+y)**2)
N=100000
np.random.seed(1)
X=np.random.uniform(0,1,N)
np.random.seed(np.random.randint(1))
Y=np.random.uniform(0,1,N)
simple_sample=f(X,Y)
print(f'简单Monte-Carlo求积分：{simple_sample.mean():.4e}')
print(f'样本标准差：{simple_sample.std():.2f}')
#对偶变量法
dual_sample=(f(1-X,1-Y)+f(X,Y))/2
print(f'对偶变量法求积分：{dual_sample.mean():.4e}')
print(f'样本标准差：{dual_sample.std():.2f}')
print(f'方差缩减百分比{(simple_sample.var()-
dual_sample.var())/simple_sample.var()*100:.2f}%')
```

```
简单Monte-Carlo求积分：4.9148e+00
样本标准差：6.03
对偶变量法求积分：4.9203e+00
样本标准差：3.42
方差缩减百分比67.90%
```

```python
#计算积分exp((x+y)^2)在[0,1]^2上的积分
import numpy as np
import matplotlib.pyplot as plt
from scipy import integrate
def f(x,y):
    return np.exp((x+y)**2)
print(f'用库函数直接求积分检验之前求的的答案，库函数积分结果：{integrate.dblquad(f,0,1,lambda x:0,lambda x:1)[0]:.3e}')
```

```
用库函数直接求积分检验之前求的的答案，库函数积分结果：4.899e+00
```

Question 5



5.6 在例5.7中，通过控制变量法计算了

$$\theta = \int_0^1 e^x \, dx$$

的蒙特卡罗积分. 现在考虑对偶变量法. 计算$\mathrm{Cov}(e^U, e^{1-U})$和$\mathrm{Var}(e^U + e^{1-U})$，其中$U \sim \mathrm{Uniform}(0,1)$. （和简单蒙特卡罗方法比较）使用对偶变量法方差缩减百分比能达到多少?

```python
#
N=1000000
np.random.seed(1)
U=np.random.uniform(0,1,N)
theta_hat=np.exp(U).mean()
print(f'对theta的估计为{theta_hat:.4f}')
print(f'样本标准差：{np.exp(U).std():.2f}')
#对偶变量法
U_tilde=(np.exp(1-U)+np.exp(U))/2
print(f'对偶变量法求积分：{U_tilde.mean():.4f}')
print(f'样本标准差：{U_tilde.std():.2f}')
print(f'方差缩减百分比{(np.exp(U).var()-U_tilde.var())/np.exp(U).var()*100:.2f}%')
```

```
对theta的估计为1.7181
样本标准差：0.49
对偶变量法求积分：1.7182
样本标准差：0.06
方差缩减百分比98.38%
```

```python
#下求cov(exp(U),exp(1-))；Var(exp(U)+exp(1-U))
print(f'下面求出题目要求的两个值：\ncov(exp(U),exp(1-U))={np.cov(np.exp(U),np.exp(1-U))[0,1]:.4f}')
print(f'Var(exp(U)+exp(1-U))={np.var(np.exp(U)+np.exp(1-U)):.4f}')
```

```
下面求出题目要求的两个值：
cov(exp(U),exp(1-U))=-0.2340
Var(exp(U)+exp(1-U))=0.0156
```

Question 6

5.14 使用重要抽样法得到

$$\int_1^{+\infty} \frac{x^2}{\sqrt{2\pi}} e^{-x^2/2} dx$$

的蒙特卡罗估计.

```
def f(x):
    return x**2*(x>1)
N=1000000
np.random.seed(1)
X=np.random.randn(N)
print(f'用重要抽样法积分为：{f(X).mean():.4e}')
```

```
用重要抽样法积分为：4.0068e-01
```

```
#用库函数直接积分x^2*exp(-x^2/2)/sqrt(2*pi)在区间[1,inf)上的积分
from scipy import integrate
print(f'用库函数直接求积分检验之前求的的答案，库函数积分结果：'
      f'{integrate.quad(lambda x:x**2*np.exp(-x**2/2)/np.sqrt(2*np.pi),1,np.inf)[0]:.4e}')
```

```
用库函数直接求积分检验之前求的的答案，库函数积分结果：4.0063e-01
```