

-
1. 利用分布密度函数 $f(x) = Ae^{-x}$ 做重要抽样来求积分，并分析误差与投点数的关系。

$$I = \int_0^{+\infty} x^{5/2} e^{-x} dx$$

(1) 写出算法过程；

- 1 将被积函数分解为 $h(x) = h^*(x) \cdot f(x)$ ，其中 $f(x) = e^{-x}$ 为概率密度，而

$h^*(x) = x^{5/2}$ 为待求量。

- 2 产生按照 $f(x) = e^{-x}$ 为概率密度的随机变量 x ，利用反函数法 $y = F(x) = 1 - e^{-x}$ ，

则 $x = -\ln(1 - y)$ ，其中 y 的取值为 $(0, 1)$ 。则产生 $(0, 1)$ 的随机数 ξ ，则选取

$-\ln(\xi)$ 为 x 的一个抽样，进行多次抽样获得 x 的抽样。

- 3 计算多次抽样中 $x_i^{5/2}$ ，进行 N 次抽样之后取平均值 $\frac{1}{N} \sum_i x_i^{5/2}$ ，即为所求积分。

(2) 写代码进行计算。

MATLAB 代码：

% 抽样点数

N = 10000;

% 初始化积分值

I = 0;

for i = 1 : N

xi = rand;

eta = - log(xi);

I = I + eta^(5/2);

end

I = I / N;

disp(I)

Python 代码：

import math

import random

N = 100000

result = 0

for i in range(N):

y = random.uniform(1e-10, 1)

x = -math.log(y)

```
result += math.sqrt(x ** 5)
```

```
Result = result / N
```

```
print(Result)
```

2. 仍然是上述积分，限制积分范围为[0,20]，请用分层抽样法计算此积分（总抽样点数为 10000，任何分层抽样法都可）：

(1) 写出算法过程；

1 将积分范围均匀分成 20 份。 $p_j = \int_{x_{j-1}}^{x_j} e^{-x} dx = e^{-x_{j-1}} - e^{-x_j}$, $j = 1, 2, \dots, 20$,

$f_j(x) = \frac{f(x)}{p_j}$, $x_{j-1} \leq x \leq x_j$ ，此为在此区间归一化的概率分布。

$I_j = \int_{x_{j-1}}^{x_j} f_j(x) h(x) dx$ 。

2 在每个区间内进行 500 次抽样，在区间 (x_{j-1}, x_j) 内，利用反函数法

$y = F_j(x) = (-e^{-x} + e^{-x_i})/p_j$ ，产生(0, 1)的随机数 ξ ，则 $x_i^j = -\ln(e^{-x_i} - p_j \xi)$ 为

此区间的一个抽样，并计算 $(x_i^j)^{5/2}$ ，同理获得该区间内 500 个抽样值 x_i^j ，计算该

区间内抽样的平均值 $I_j = \frac{1}{N} \sum_i (x_i^j)^{5/2}$ ，计算并存储该区间的 $p_j I_j$ 。

3 最后将各个区间得到的 I_i 进行概率求和 $I = \sum_j p_j I_i$ 。

(2) 写代码进行计算。

MATLAB 代码：

% 均匀分层 20 层，每层 500 个点

```
N = 10000;
```

```
n = 20;
```

```
Nn = N / n;
```

% 初始化每层积分

```
y = zeros(1, n);
```

```
for i = 1 : n
```

```
    x = rand(1, Nn);
```

```
    x = x + i - 1;
```

```
    y(i) = 1/Nn * x.^(5/2) * exp(-x)';
```

```
end
```

```
I = sum(y);
```

```
disp(I)
```

Python 代码：

```
import math
```

```
import random
```

```
N = 500
```

```
L = 20
```

```
result = 0
```

```
for i in range(L):
```

```
    Num = 0
```

```
    p = math.exp(-i) - math.exp(-i - 1)
```

```
    print(p)
```

```
    Sum = 0
```

```
    for j in range(N):
```

```
        y = random.uniform(1e-10, 1)
```

```
        x = - math.log(math.exp(-i) - p * y)
```

```
        Sum += math.sqrt(x ** 5) / N
```

```
    result += p * Sum
```

```
print(result)
```