1 实验课题

下面利用蒙特卡罗方法计算一个冰淇淋的体积和质量.

假设冰淇淋的下部为一锥体而上部为一个半球, 锥面方程为$z^2 = x^2 + y^2$, 球面方程为$x^2 + y^2 + (z-1)^2 = 1$. 完成以下实验:

1. 画出冰淇淋的图形.
2. 分别用确定性方法和蒙特卡罗方法计算冰淇淋的体积, 比较计算结果.
3. 假设冰淇淋的密度函数为为$f(x,y,z) = z\exp[-(x^2+y^2+z^2)]$, 分别用确定性方法和蒙特卡罗方法计算冰淇淋的质量, 比较计算结果.

2 图形绘制

联列两曲线方程我们可以得到冰淇淋图形以平面$z = 1$为分界面, 上部分为半球, 下部分为圆锥.

对于圆锥, 因为$z \geqslant 0$, 所以我们有$z = \sqrt{x^2+y^2}$. 绘制圆锥的程序如下:

```
z1 = @(x,y) sqrt(x.^2+y.^2);
g1 = ...
    ezmesh(z1,[-1/sqrt(2),1/sqrt(2),-1/sqrt(2),1/sqrt(2)],'circ');
```

对于半球, 因为$z \geqslant 1$, 所以我们有$z = \sqrt{1-x^2-y^2}+1$. 绘制半球的程序如下:

```
z2 = @(x,y) sqrt(1-x.^2-y.^2)+1;
g2 = ...
    ezmesh(z2,[-1/sqrt(2),1/sqrt(2),-1/sqrt(2),1/sqrt(2)],'circ');
```

冰淇淋图形的绘制结果如**图1**.

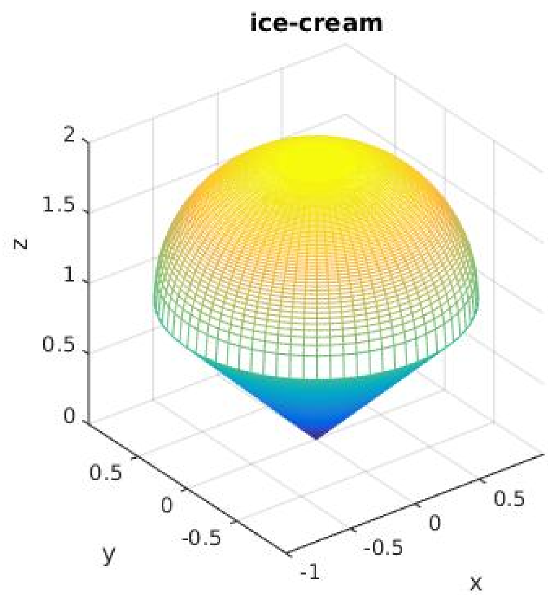

图1 冰淇淋示意图

3 体积计算

3.1 精确解

我们引入球坐标, 则$x=r\sin\theta\cos\phi$, $y=r\sin\theta\sin\phi$, $z=r\cos\theta$. 其中θ为球面上的点与z轴正向的夹角, ϕ为平面xOy上的点与x轴正向的夹角. 利用球面三重积分我们可以得到冰淇淋的体积

$$V=\int_{\omega}\mathrm{d}x\mathrm{d}y\mathrm{d}z=\int_0^{2\pi}\int_0^{\pi/4}\int_0^{2cos\theta}r^2\sin\theta\mathrm{d}r\mathrm{d}\theta\mathrm{d}\phi.$$

下面是求冰淇淋体积的MATLAB程序.

```
V1 = eval(int(int(int(r*r*sin(theta),r,0,2*cos(theta)),...
    theta,0,pi/4),phi,0,2*pi));
```

上面程序的计算得到的体积为3.141593. 事实上, 如果我们去掉上面代码中的eval函数, 在MATLAB软件的控制台中显示的计算结果就是pi.

3.2 近似解

首先, 冰淇淋可以被装在如下的立方体区域内:

$$-1\leqslant x\leqslant 1, -1\leqslant y\leqslant 1, 0\leqslant z\leqslant 2.$$

接下来我们在这个体积为8的立方体区域中生成随机点. 若我们共生成了N个随机点, 其中有m个落在了冰淇淋区域内, 那么冰淇淋的体积V可以被表示为

$$V=\frac{8m}{N}.$$

下面是上述方法的MATLAB实现.

```
countAll = 1000000;
countInIceCream = 0;
for i = 1:countAll
   w = [2*rand(1)-1,2*rand(1)-1,2*rand(1)];
   if (w(3)^2>w(1)^2+w(2)^2 && w(1)^2+w(2)^2+(w(3)-1)^2<1)
       countInIceCream = countInIceCream + 1;
       %S = S + f(w(1),w(2),w(3));
   end
end
V2 = 8 * countInIceCream / countAll;
```

由于我们使用了随机数, 上面的程序的计算结果具有不确定性, 但根据概率统计原理我们知道, countAll越大, 我们的计算结果越倾向于靠近π.本次实验中我们取countAll的值为1000000,得到冰淇淋的近似体积3.141360.

4 质量计算

4.1 精确解

我们再次使用球坐标变换,并把冰淇淋的密度函数$f(x,y,z)=z\mathrm{e}^{-(x^2+y^2+z^2)}$转换为$g(r,\theta,\phi)=r\cos\theta\mathrm{e}^{-r^2}$. 对冰淇淋区域内的密度函数积分, 我们可以得到冰淇

淋的质量

$$M=\int_{\omega}f(x,y,z)\mathrm{d}x\mathrm{d}y\mathrm{d}z=\int_0^{2\pi}\int_0^{\pi/4}\int_0^{2cos\theta}r^2\sin\theta g(r,\theta,\phi)\mathrm{d}r\mathrm{d}\theta\mathrm{d}\phi.$$

下面是求冰淇淋质量的MATLAB程序. 从而我们得到冰淇淋质量的精确解为0.615969.

```
M1 = eval(int(int(int(r^3*sin(theta)*cos(theta)*exp(-r^2),...
    r,0,2*cos(theta)),theta,0,pi/4),phi,0,2*pi));
```

4.2 近似解

与估算体积时的方法类似,我们在立方体区域中随机生成N个点. 首先我们用落在冰淇淋区域中的点来估计冰淇淋的平均密度, 接着我们估计的密度和之前估算的体积求得冰淇淋的质量.

如果有m个点落在冰淇淋区域内, 记第i个这样的点为P_i, 则$P_i(x_i,y_i,z_i)$处的密度为$f(x_i,y_i,z_i)$. 所以我们可以得到这m个点的平均密度

$$\rho=\frac{\sum_{i=1}^m f(x_i,y_i,z_i)}{m}.$$

用ρ代替冰淇淋的实际密度, 再带入估计得到的冰淇淋体积V, 我们可以得到冰淇淋质量的表达式

$$M=\rho V=\frac{\sum_{i=1}^m f(x_i,y_i,z_i)V}{m}.$$

下面的MATLAB程序实现了上述算法, 并得到0.615819的质量估算结果.

```
S = 0;
f = @(x,y,z) z*exp(-(x^2+y^2+z^2));
for i = 1:countAll
   w = [2*rand(1)-1,2*rand(1)-1,2*rand(1)];
   if (w(3)^2>w(1)^2+w(2)^2 && w(1)^2+w(2)^2+(w(3)-1)^2<1)
       %countInIceCream = countInIceCream + 1;
       S = S + f(w(1),w(2),w(3));
   end
end
M2 = S / countInIceCream * V2;
```

5 实验结论

本实验以一个简单的冰淇淋图形为研究对象, 考察了用蒙特卡罗方法求积分的近似值.

在本次实验中, 使用蒙特卡罗方法得到的计算结果相对误差较小: 当随机点个数达到百万级别时, 相对误差基本控制在0.05%以下.

6 附录

Lab03.m 源代码

```
%Lab03.m
clear;clc;format long;
syms theta r phi;
countAll = 1000000;
countInIceCream = 0;
S = 0;
f = @(x,y,z) z*exp(-(x^2+y^2+z^2));

%draw the figure
z1 = @(x,y) sqrt(x.^2+y.^2);
figure('name','ice-cream figure');
g1 = ...
    ezmesh(z1,[-1/sqrt(2),1/sqrt(2),-1/sqrt(2),1/sqrt(2)],'circ');
hold on;
z2 = @(x,y) sqrt(1-x.^2-y.^2)+1;
g2 = ...
    ezmesh(z2,[-1/sqrt(2),1/sqrt(2),-1/sqrt(2),1/sqrt(2)],'circ');
title('ice-cream');
axis equal;

%generate random points
for i = 1:countAll
   w = [2*rand(1)-1,2*rand(1)-1,2*rand(1)];
   if (w(3)^2>w(1)^2+w(2)^2 && w(1)^2+w(2)^2+(w(3)-1)^2<1)
        countInIceCream = countInIceCream + 1;
        S = S + f(w(1),w(2),w(3));
   end
end

%calculate volume
V1 = eval(int(int(int(r*r*sin(theta),r,0,2*cos(theta)),...
    theta,0,pi/4),phi,0,2*pi));
V2 = 8 * countInIceCream / countAll;
fprintf('The actual volume of the ice-cream is %.6f.\n\n',V1);
fprintf('The estimated volume of the ice-cream is %.6f.\n\n',V2);


%calculate mass
M1 = eval(int(int(int(r^3*sin(theta)*cos(theta)*exp(-r^2),...
    r,0,2*cos(theta)),theta,0,pi/4),phi,0,2*pi));
M2 = S / countInIceCream * V2;
fprintf('The actual mass of the ice-cream is %.6f.\n\n',M1);
fprintf('The estimated mass of the ice-cream is %.6f.\n\n',M2);
```