미분적분학2을 위한
SageMath™ 기본 사용법

강원대학교 자연과학대학
수학과

목 차

제 1장

매개 변수 방정식과 극 좌표

1.1 매개 변수 곡선의 그래프 그리기

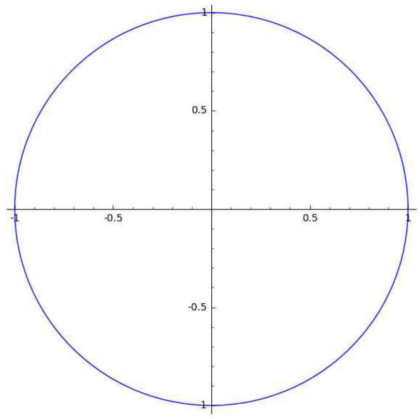
x와 y가 매개 변수라 부르는 제 3의 변수 t에 의해 다음과 같이 정의된다고 하자.

$$x = f(t), \quad y = g(t) \qquad (a \le t \le b)$$

이 방정식을 매개 변수 방정식이라 하고, t가 변함에 따라 좌표 평면에 그리는 곡선을 매개 변수 곡선 또는 매개 곡선이라 한다.

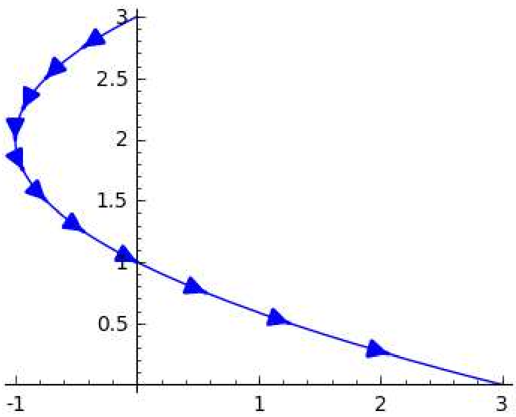
다음 예제를 통하여 매개 변수 곡선을 그려보자. 먼저 매개 변수 t를 **var** 명령어로 변수를 지정하고 **x**와 **y**를 각각 t에 대한 함수로 지정한다.

```
sage : var('t,x,y')
sage : x=cos(t)
sage : y=sin(t)
sage : a=0          # t의 최솟값
sage : b=2*pi       # t의 최댓값
sage : parametric_plot((x, y),(t, a, b))
```



위의 예제는 우리가 많이 접해본 단위 원에 대한 매개 변수 방정식이다. 이번에는 매개 변수 방정식이 $x = t^2 - 2t, \quad y = 3 - t \quad (0 \le t \le 3)$인 곡선을 그려보고, t가 증가함에 따라 곡선 위의 점이 어느 방향으로 이동하는지 표시해보자.

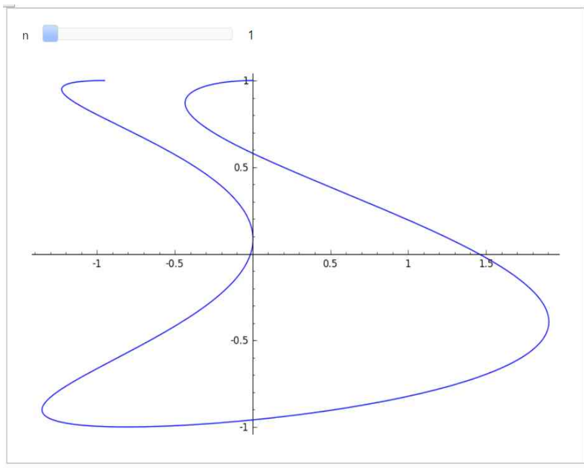
```
sage : var('t,x,y')
sage : x=t^2-2*t
sage : y=3-t
sage : a=0    # t의 최솟값
sage : b=3    # t의 최댓값
sage : p=parametric_plot((x, y),(t, a, b))
sage : small=0.001
sage : step=0.25        # t가 0.25마다 하나의 화살표를 표시
sage : n=(b-a)/step   # 화살표의 총 개수
sage : arr=sum([arrow((x(t=a+i*step), y(t=a+i*step)),
     (x(t=a+i*step+small), y(t=a+i*step+small))) for i in range(1,n) ])
sage : p+arr
```



주어진 매개 변수 방정식에서 매개 변수 t를 소거하여 x, y의 관계식을 구하면 위의 곡선은 포물선임을 쉽게 알 수 있다. 위의 두 예제들은 우리에게 친숙한 곡선들로 손쉽게 그릴 수 있으나 다음 곡선들은 손으로 그리기가 사실상 불가능하다.

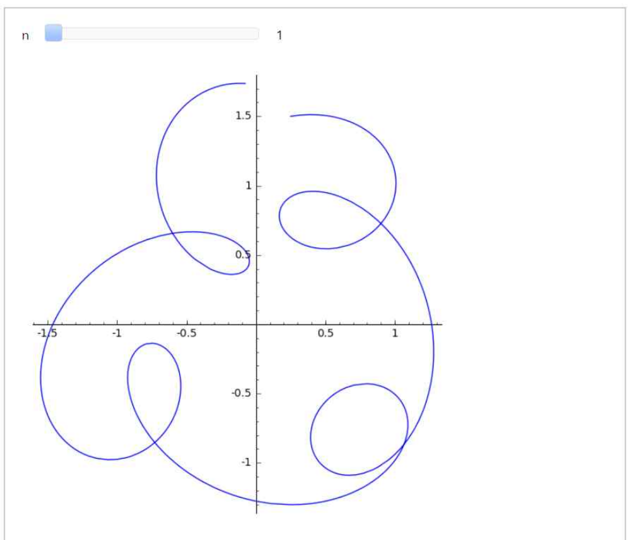
$$x = \sin t - \sin(2.3t), \quad y = \cos t$$

```
sage : var('t,x,y')
sage : x= sin(t)-sin((2.3)*t)
sage : y= cos(t)
sage : a=0     # t의 최솟값
sage : @interact
sage : def _(n=(1..12)):
sage :       b=2*n*pi     # t의 최댓값
sage :       p=parametric_plot((x, y),(t, a, b))
sage :       show(p)
```



$$x = \sin t + \frac{\sin 5t}{2} + \frac{\cos 2.3t}{4}, \quad y = \cos t + \frac{\cos 5t}{2} + \frac{\sin 2.3t}{4}$$

```
sage : var('t,x,y')
sage : x= sin(t)+(1/2)*sin(5*t)+(1/4)*cos((2.3)*t)
sage : y= cos(t)+(1/2)*cos(5*t)+(1/4)*sin((2.3)*t)
sage : a=0     # t의 최솟값
sage : @interact
sage : def _(n=(1..12)):
sage :      b=2*n*pi     # t의 최댓값
sage :      p=parametric_plot((x, y),(t, a, b))
sage :      show(p)
```



1.2 매개 변수 곡선에 대한 미적분

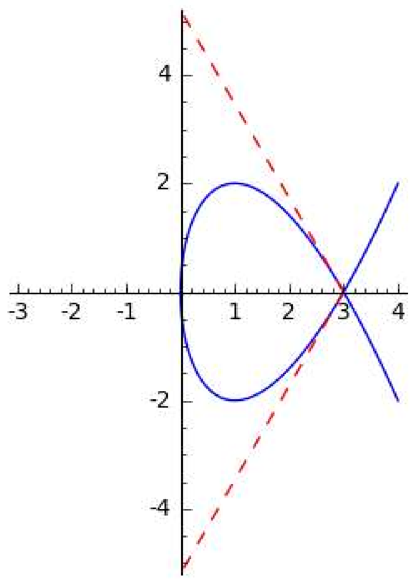
1.2.1 접선

매개 변수 곡선 $x=f(t),\ y=g(t)$에 대한 접선의 기울기는 다음과 같다.

$$\frac{dy}{dx}=\frac{\frac{dy}{dt}}{\frac{dx}{dt}}=\frac{g'(t)}{f'(t)} \qquad (\text{단, } f'(t)\neq 0)$$

이 식을 이용하여 $x=t^2,\ y=t^3-3t$로 정의된 매개 변수 곡선에 대하여 점 (3,0)에서 접선을 구하고 이를 그려보자.

```
sage : var('t, x')
sage : f(t)=t^2
sage : g(t)=t^3-3*t
sage : dydx=diff(g(t), t)/diff(f(t), t)
sage : solve([f(t)==3, g(t)==0],t)
       [[t == sqrt(3)], [t == -sqrt(3)]]
sage : T1=dydx(sqrt(3));T2=dydx(-sqrt(3))
sage : p=parametric_plot((f(t), g(t)), (t, -2, 2))
sage : q=plot(T1*(x-3), (x, -3, 3), linestyle='--', color='red')
sage : r=plot(T2*(x-3), (x, -3, 3), linestyle='--', color='red')
sage : (p+q+r).show(figsize='5',ymin=-5,ymax=5)
```



또한, 매개 변수 곡선에 대한 이계 도함수는 다음과 같이 찾을 수 있다.

$$\frac{d^2y}{dx^2} = \frac{d}{dx}\left(\frac{dy}{dy}\right) = \frac{\dfrac{d}{dt}\left(\dfrac{dy}{dx}\right)}{\dfrac{dx}{dt}}$$

위의 식을 이용하여 매개 변수 곡선 $x = t + e^t$, $y = t - e^t$에 대하여 이계 도함수 $\dfrac{d^2y}{dx^2}$를 구하고, 곡선의 볼록성을 조사해보자.

```
sage : var('t')
sage : x(t)=t+exp(t)
sage : y(t)=t-exp(t)
sage : dy(t)=diff(y(t), t)
sage : dx(t)=diff(x(t), t)
sage : dydx=dy(t)/dx(t)
sage : d2y=(diff(dydx, t)/dx(t)).factor(); d2y
       -2*e^t/(e^t + 1)^3
```

이 때 **d2y**가 항상 음수이므로 위의 매개 변수 곡선은 모든 점에서 위로 볼록이다.

1.2.2 넓이와 호의 길이

$F(x) \geq 0$ 일 때 곡선 $y = F(x)$ $(a \leq x \leq b)$의 아래의 넓이는 $\displaystyle\int_a^b F(x)\,dx$이므로, 매개 변수 곡선 $x = f(t)$, $y = g(t)$ $(\alpha \leq t \leq \beta)$ 아래의 넓이는 $\displaystyle\int_\alpha^\beta g(t)f'(t)\,dt$로 구할 수 있다. 또한 호의 길이도 $\displaystyle\int_\alpha^\beta \sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2}\,dt$로 구할 수 있다.

예를 들어, 싸이클로이드 곡선 $x = r(\theta - \sin\theta)$, $y = r(1 - \cos\theta)$의 한 아치의 넓이를 구해보자. 여기서 한 아치의 범위는 $0 \leq \theta \leq 2\pi$이다.

```
sage : var('x,y,r,t')
sage : x=r*(t-sin(t))
sage : y=r*(1-cos(t))
sage : a=0; b=2* pi
sage : dxdt=diff(x,t);
sage : integral(y*dxdt,t,0,2*pi)
       3*pi*r^2
```

이번에는 구간 $0 \le t \le \pi$에서 곡선 $x = e^t \cos t$, $y = e^t \sin t$의 길이를 구해보자.

```
sage : var('x,y,t')
sage : x=exp(t)*cos(t)
sage : y=exp(t)*sin(t)
sage : dxdt=diff(x,t);
sage : dydt=diff(y,t);
sage : integral(sqrt(dxdt^2+dydt^2),t,0,pi)
       sqrt(2)*e^pi - sqrt(2)
```

1.3 극 좌표, 극 좌표에서 넓이과 길이

1.3.1 극 좌표와 직교 좌표

직교 좌표 (x, y)와 극 좌표 (r, θ)의 관계식은 다음과 같다.

$$x = r\cos\theta,\quad y = r\sin\theta \quad \text{또는} \quad r = \sqrt{x^2 + y^2},\quad \tan\theta = \frac{y}{x}$$

명령어 **def**를 사용하여 극 좌표를 직교 좌표로 바꾸는 함수를 정의해보자.

```
sage : def Polar(r, theta):
sage :     # 극 좌표를 직교 좌표로 바꾸는 함수의 정의
sage :     CartC = ([r*cos(theta), r*sin(theta)]);
sage :     return CartC;
sage : pt=Polar(4, pi/4);
sage : vector(pt)
       (2*sqrt(2), 2*sqrt(2))
```

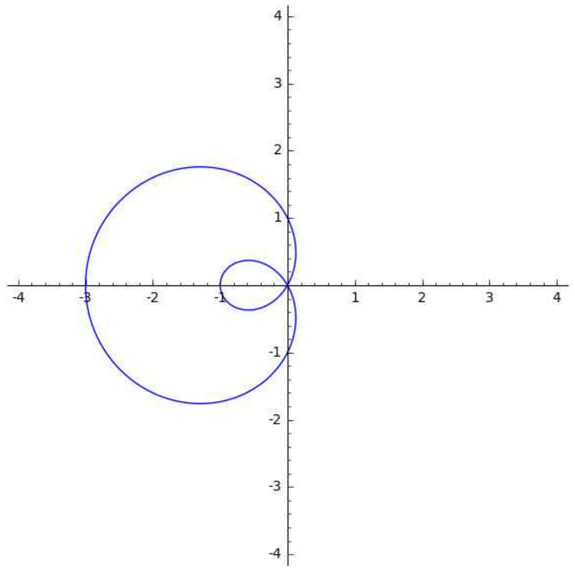
다음은 직교 좌표를 극 좌표로 바꾸는 함수이다.

```
sage : def Cart2Pol(x, y):
sage :     # 직교 좌표를 극 좌표로 바꾸는 함수의 정의
sage :     r=sqrt(x^2+y^2)
sage :     if x<0:
sage :         theta=arctan(y/x)+pi
sage :     else:
sage :         theta=arctan(y/x)
sage :     PolC = ([r, theta])
sage :     return PolC
sage : pt=Cart2Pol(sqrt(3), -1)
sage : vector(pt).simplify_full()
       (2, -1/6*pi)
```

1.3.2 극 곡선의 호의 길이와 넓이

극 방정식 $r=f(\theta)$로 주어지는 극 곡선을 그리는 명령어는 **polar_plot**이다. 곡선을 그릴 때 정의역과 치역의 범위의 차이가 많이 나면 그래프가 실제와 다른 모양으로 보일 수 있는데, 아래와 같이 **aspect_ratio** 옵션을 사용하여 x축과 y축의 비율을 1:1로 맞춰주면 된다.

```
sage : var('theta')
sage : polar_plot(1-2*cos(theta), (0, 2*pi)).show(aspect_ratio=1,
       xmin=-4, xmax=4, ymin=-4, ymax=4)
```

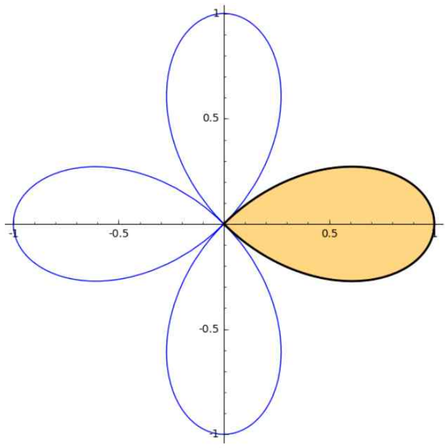


극 곡선 $r=f(\theta)$와 두 반직선 $\theta=a, \theta=b$로 둘러싸인 영역의 넓이는 $\int_a^b \frac{1}{2}r^2 d\theta$이고, 극 곡선 $r=f(\theta)$ $(a \le \theta \le b)$의 길이는 $\int_a^b \sqrt{r^2+\left(\frac{dr}{d\theta}\right)^2}d\theta$이다. 예를 들어, 4엽장미 $r=\cos 2\theta$의 한 고리로 둘러싸인 영역의 넓이를 구해보자.

```
sage : var('r,t')
sage : r=cos(2*t)
sage : p=polar_plot(r, (0, 2*pi))
sage : q=polar_plot(r, (-pi/4, pi/4),color='black', thickness=2,
         fill=True, fillcolor='orange')
sage : (p+q).show(aspect_ratio=1)
```

아래 그림에서 한 고리로 둘러싸인 영역은 그래프의 색 칠해진 부분이며 이에 해당되는 θ의 범위는 $-\frac{\pi}{4} \le \theta \le \frac{\pi}{4}$이다. 따라서 한 고리의 넓이는 다음과 같다.

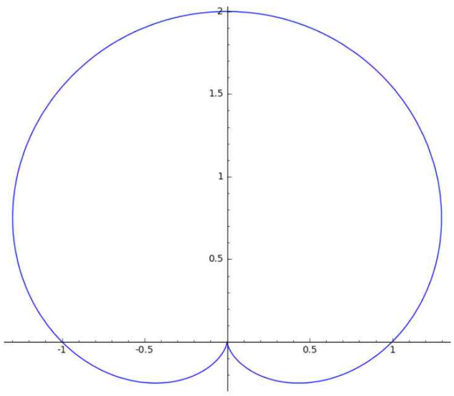
호의 길이를 구할 때는 피적분함수에 제곱근이 포함되므로 대부분의 경우 정확한 적분을 구하는 것이 불가능하다. 이런 경우에는 중점 법칙, 사다리꼴 공식과 같은 근사 적분이나 다음과 같이 테일러 급수를 이용하여 적분의 근삿값을 구할 수 있다.



```
sage : integral((1/2)*r^2, t,-pi/4,pi/4)
     1/8*pi
```

이번에는 심장선 $r=1+\sin\theta$를 그리고, 그 길이를 구해보자.

```
sage : var('r,t')
sage : r=1+sin(t)
sage : polar_plot(r, (0, 2*pi))
```



```
sage : f(t)=sqrt(r^2+diff(r,t)^2).simplify_full();    #피적분함수
sage : show(f)
      t -> sqrt(2*sin(t) + 2)
sage : integral(f(t),t,0,2*pi)    #손으로 적분하면 8이 나오는데...
```

호의 길이를 구할 때는 피적분함수에 제곱근이 포함되므로 대부분의 경우 정확한 적분을 구하는 것이 불가능하다. 이런 경우에는 중점 법칙, 사다리꼴 공식과 같은 근사 적분이나 다음과 같이 테일러 급수를 이용하여 적분의 근삿값을 구할 수 있다.

예를 들어, 테일러 급수를 이용하여 적분 $\int_0^{2\pi} \sqrt{2\sin t + 2}\, dt$의 근사값을 구해보자. 이 때 유의할 점은, 피적분함수 $f(t) = \sqrt{2\sin t + 2}$가 점 $t = \frac{3}{2}\pi$에서 미분 가능하지 않으므로 적분 구간 $[0, 2\pi]$에서 테일러 급수를 이용하면 부정확한 결과를 얻는다는 것이다. ($f(t)$와 테일러 다항식의 그래프를 그려 비교해보라!) 점 $t = \frac{3}{2}\pi$를 피하기 위해 적분을

$$\int_0^{2\pi} \sqrt{2\mathrm{sin}t + 2}\, dt = 2\int_{-\pi/2}^{\pi/2} \sqrt{2\mathrm{sin}t + 2}\, dt$$

로 쓴 다음, 구간 $[-\pi/2, \pi/2]$에서 테일러 급수를 이용해보자. (구간이 원점에 대해 대칭이므로 홀수 차수의 항들은 적분하면 0이 된다. 따라서, 차수를 2씩 증가시켜야 더 정확한 근사값이 얻어진다.)

```
sage : a1=integral(f.taylor(t,0,0),t,-pi/2,pi/2)
sage : n=2
sage : ep=1e-10     #오차 범위
sage : while True:
sage :     a2=integral(f.taylor(t,0,n),t,-pi/2,pi/2)
sage :     if abs(a1-a2)<=ep: break
sage :     n=n+2     # 차수가 2씩 증가함
sage :     a1=a2
sage : N(2*a2)
       8.00000000000023
```

제 2장

수열과 급수

2.1 수열의 극한

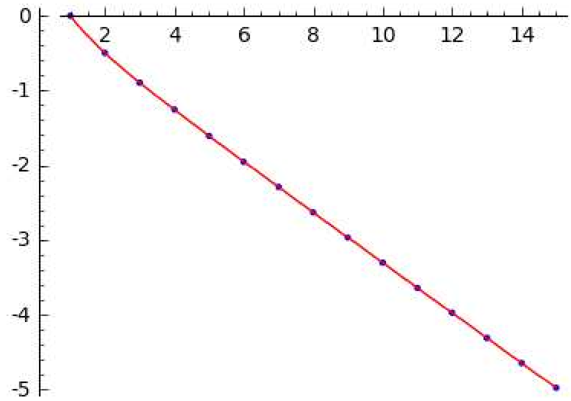
주어진 수열의 처음 몇 개의 항들을 나타내기 위해 다음과 같은 명령어를 사용해 보자. (아래 그림의 n을 옮겨 주면 각각의 항들을 볼 수 있다.)

```
sage : var('n,i')
sage : @interact()
sage : def _(n=slider(1,100,1)):
sage :     for i in srange(1,n+1,1):
sage :       print 5*i-3
```

이번에는 수열 $a_n = \dfrac{1-n^3}{2+3n^2}$을 n의 함수로 보고 그래프로 나타내어 $n \to \infty$일 때 수렴하는지 발산하는지 추정해보자.

```
sage : var('x, i, n')
sage : f(x)=(1-x^3)/(2+3*x^2)
sage : p = plot(f(x), (x, 1, 15), color='red')
sage : q = list_plot([(i,f(i)) for i in range(1, 16, 1)], color='blue')
sage : p+q
```

아래 그림에서 파란색 점은 수열의 각 항을 나타내는데, 이를 살펴보면 수열 a_n은 $-\infty$로 발산하는 것처럼 보인다.

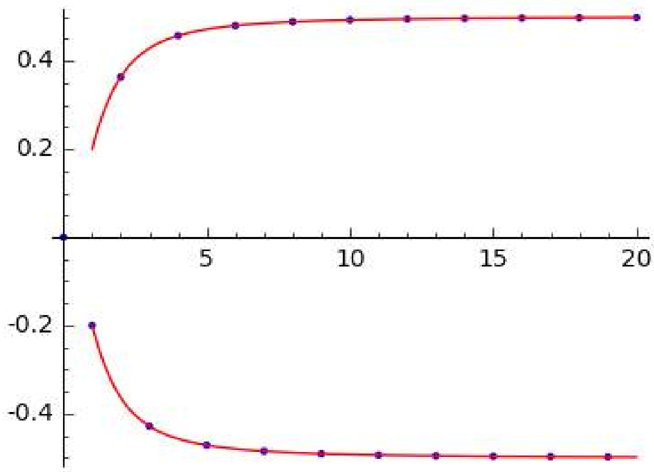


이는 다음과 같은 명령어로 확인할 수 있다.

```
sage : limit((1-n^3)/(2+3*n^2), n=+oo)
  -Infinity
```

다음은 진동하는 수열 $b_n = \dfrac{(-1)n^2}{3+2n^2}$을 그래프로 나타내보자.

```
sage : var('x, i, n')
sage : f(x)=x^2/(3+2*x^2)
sage : p=plot(f(x), (x, 1, 20), color=' red' )
sage : q=plot((-1)*f(x), (x, 1, 20), color=' red' )
sage : r=list_plot([((-1))^i*f(i) for i in range(0, 21, 1)], color=' blue' )
sage : p+q+r
```



2.2 무한 급수

수열 $\{a_n\}_{n=1}^{\infty}$ 의 항들을 더한 것을 무한 급수 또는 급수라 하고, $\sum_{n=1}^{\infty} a_n$으로 나타낸다. 급수의 합은 **sum** 명령어로 구할 수 있다. 예를 들어, 급수

$$\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\frac{1}{16}+\cdots+\frac{1}{2^n}+\cdots$$

의 합을 구한다고 하자.

```
sage : var('n')
sage : sum(1/(2^n),n,1,+oo)
    1
```

```
sage : var('k,n')
sage : sn=sum(1/(2^k),k,1,n)
sage : limit(sn,n=+oo)
    1
```

위와 같이 무한까지의 합을 바로 구하거나 부분합의 극한으로 급수의 합을 찾을 수 있다. 다음의 급수들을 살펴보고 그 합을 구해보자.

$$\sum_{n=1}^{\infty} \frac{1}{n(n+1)}$$

```
sage : sum(1/(n*(n+1)),n,1,+oo)
    1
```

$$\sum_{n=1}^{\infty} 5\left(-\frac{2}{3}\right)^{n-1}$$

```
sage : sum(5*(-2/3)^(n-1),n,1,+oo)
    3
```

$$\sum_{n=1}^{\infty} \frac{1}{n}$$

```
sage : sn=sum(1/k,k,1,n)
sage : limit(sn,n=+oo)
    limit(harmonic_number(n), n, +Infinity)
```

$$\sum_{n=1}^{\infty} \frac{1}{n^2}$$

```
sage : sn=sum(1/(k^2),k,1,n)
sage : limit(sn,n=+oo)
    limit(harmonic_number(n, 2), n, +Infinity)
```

Sage에서는 급수의 정확한 합을 구하기 어려운 경우 위와 같은 메시지가 나온다. 세 번째와 네 번째 예제에서 부분합 s_n의 값들을 살펴보자.

```
sage : var('k')
sage : A=[ ['n','1/n','1/(n^2)'] ]
sage : x=[5,10,50,100,1000,10000]
sage : B=[ (n, N(sum(1/k,k,1,n)),N(sum(1/k^2,k,1,n))) for n in x]
sage : table(A+B,header_row=True, frame=True, align='center')
```

n	1/n	1/(n^2)
5	2.28333333333333	1.46361111111111
10	2.92896825396825	1.54976773116654
50	4.49920533832942	1.62513273362153
100	5.18737751763962	1.63498390018489
1000	7.48547086055035	1.64393456668156
10000	9.78760603604438	1.64483407184806

위의 결과를 살펴보면, 급수 $\sum_{n=1}^{\infty} \frac{1}{n}$은 수렴하지 않는 것처럼 보이고 급수 $\sum_{n=1}^{\infty} \frac{1}{n^2}$은 수렴하는 것처럼 보인다. 실제로 급수 $\sum_{n=1}^{\infty} \frac{1}{n}$은 $+\infty$로 발산하고 급수 $\sum_{n=1}^{\infty} \frac{1}{n^2}$은 $\frac{\pi^2}{6} = 1.6449340668482264 \cdots$으로 수렴한다.

2.3 급수 판정법

일반적으로 급수의 정확한 합을 구하기는 거의 불가능하다. 이번 절에서는 다음과 같이 간단히 사용할 수 있는 급수 판정법으로 급수의 수렴 / 발산을 판정해보자.

▶ 적분 판정법

함수 f는 $[1, \infty)$에서 연속이고 양수이며 감소한다고 하자. 그리고 $a_n = f(n)$라 하자.

그러면 $\sum_{n=1}^{\infty} a_n$이 수렴하기 위한 필요 충분 조건은 $\int_1^{\infty} f(x)dx$가 수렴하는 것이다.

▶ 비 판정법

(i) $\lim_{n \to \infty} \left| \frac{a_{n+1}}{a_n} \right| = L < 1$이면, $\sum_{n=1}^{\infty} a_n$은 수렴한다.

(ii) $\lim_{n \to \infty} \left| \frac{a_{n+1}}{a_n} \right| = L > 1$ 또는 $\lim_{n \to \infty} \left| \frac{a_{n+1}}{a_n} \right| = \infty$이면, $\sum_{n=1}^{\infty} a_n$은 발산한다.

(iii) $\lim_{n \to \infty} \left| \frac{a_{n+1}}{a_n} \right| =1$이면, $\sum_{n=1}^{\infty} a_n$은 비 판정법으로 수렴 / 발산을 결정할 수 없다.

다음은 적분 판정법을 이용하여 급수 $\sum_{n=1}^{\infty} \frac{1}{n^2}$이 수렴함을 보이는 것이다.

```
sage : f(x)=1/(x^2)           # f(x)는 x>1일 때 양수인 함수
sage : df=diff(f(x),x);df     # x>1 일 때 도함수가 음수이므로 감소 함수
      -2/x^3
sage : integral(f(x),x,1,+oo)
      1
```

마찬가지로, 다음 급수들에 대하여 수렴 / 발산을 판단해보자.

$$\sum_{n=2}^{\infty} \frac{\ln n}{n}$$

```
sage : f(x)=log(x)/x          # f(x)는 x>2일 때 양수인 함수
sage : df=diff(f(x),x);df     # x>2 일 때 도함수가 음수이므로 감소 함수
      -log(x)/x^2 + 1/x^2
sage : integral(f(x),x,2,+oo)
      Traceback (click to the left of this block for traceback)
      ...
      ValueError: Integral is divergent.
```

$$\sum_{n=1}^{\infty} \frac{1}{n(\ln n)^2}$$

```
sage : f(x)=1/(x*log(x)^2)    # f(x)는 x>2일 때 양수인 함수
sage : df=diff(f(x),x);df      # x>2 일 때 도함수가 음수이므로 감소함수
      -1/(x^2*log(x)^2) - 2/(x^2*log(x)^3)
sage : integral(f(x),x,2,+oo)
      1/log(2)
```

$$\sum_{n=1}^{\infty} \frac{2^n n!}{n^n}$$

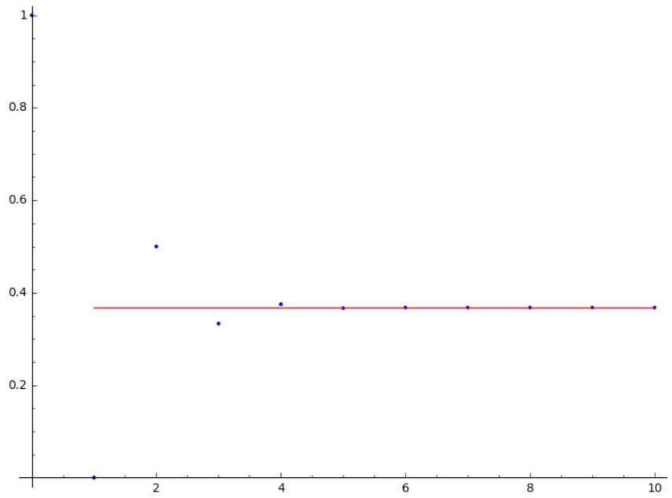
```
sage : var('n')
sage : u(n)=2^n*factorial(n)/n^n
sage : limit(u(n+1)/u(n), n=+oo)
      2*e^(-1)
sage : bool(2*e^(-1)<1)
      True
```

다음으로 교대급수 판정법을 이용하여 급수 $\sum_{n=0}^{\infty} \frac{(-1)^n}{n!}$이 수렴함을 보이자.

```
sage : var('n')
sage : a(n) = (-1)^n/factorial(n)
sage : b(n) = 1/factorial(n)
sage : (b(n)/b(n+1)).simplify_full()
      n + 1
sage : limit(b(n),n=+oo)
      0
```

모든 n에 대해 $b_n \geq b_{n+1}$이고 수열 b_n의 극한값이 0이므로 교대급수 판정법에 의해 급수 $\sum_{n=0}^{\infty} \frac{(-1)^n}{n!}$은 수렴한다. 이때 급수의 합은 다음과 같이 구할 수 있으며, 아래 그래프로부터 급수의 항들을 계속 더해 가면 급수의 합에 근접함을 볼 수 있다.

```
sage : s=sum(a(n),n,0,+oo); s
      e^(-1)
sage : var('i,n')
sage : p=plot(s, (x, 1, 10), color='red')
sage : q=list_plot([sum(a(i),i,0,n) for n in range(0,11,1)], color='blue')
sage : p+q
```



2.4 거듭제곱 급수, 테일러 급수, 매클로린 급수

거듭제곱 급수는 다음과 같은 형태의 급수이다.

$$\sum_{n=0}^{\infty} c_n (x-a)^n = c_0 + c_1(x-a) + c_2(x-a)^2 + \cdots$$

여기서 변수 x에 고정된 값을 대입하면 실수들의 급수이고, 이에 대한 수렴 여부를 판정할 수 있다. 거듭제곱 급수는 x의 일부 값에서는 수렴하고 x의 다른 값에서는 발산할 수 있다. 이 때 다음 세 가지 중에서 하나만 성립한다.

(i) $x=a$일 때만 수렴한다.
(ii) x의 모든 값에 대해 수렴한다.
(iii) 적당한 양수 R이 존재하여 $|x-a|<R$일 때 수렴하고 $|x-a|>R$일 때 발산한다. 이 때 양수 R을 수렴 반지름이라고 한다.

비 판정법을 이용하여 거듭제곱 급수의 수렴 반지름을 구해보자.

$$\sum_{n=0}^{\infty} \frac{(-3)^n x^n}{\sqrt{n+1}}$$

```
sage : var('n')
sage : a(n)=(-3)^n/sqrt(n+1)
sage : R=limit(abs(a(n)/a(n+1)), n=+oo); R    # R은 수렴 반지름이다.
       1/3
```

$$\sum_{n=0}^{\infty} \frac{n(x+2)^n}{3^n}$$

```
sage : var('n')
sage : a(n)=n/3^n
sage : R=limit(abs(a(n)/a(n+1)), n=+oo); R    # R은 수렴 반지름이다.
       3
```

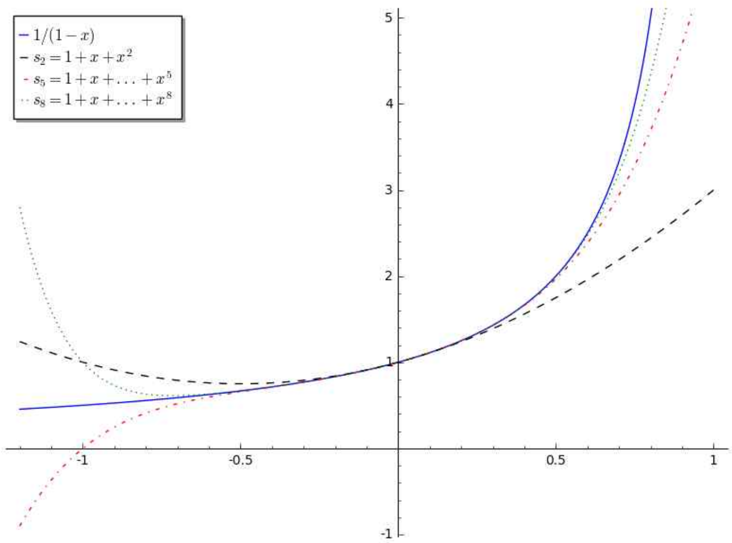
점 $x=a$에서 함수 f의 테일러 급수와 그 부분합은 다음과 같다.

$$f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n, \qquad T_n(x) = \sum_{i=0}^{n} \frac{f^{(i)}(a)}{i!}(x-a)^i$$

이 때 T_n을 점 a에서 f의 n차 테일러 다항식이라고 한다.

함수 f의 테일러 급수가 수렴 구간 내에서 f로 수렴하면 f는 테일러 다항식으로 근사할 수 있다. 예를 들어, $f(x) = \dfrac{1}{1-x}$의 테일러 급수를 살펴보자.

```
sage : var('n,x')
sage : assume(abs(x)<1)
sage : sn=sum(x^n, n, 0, oo)
sage : s2=sum(x^n, n, 0, 2)     # 2차 항까지의 합
sage : s5=sum(x^n, n, 0, 5)     # 5차 항까지의 합
sage : s8=sum(x^n, n, 0, 8)     # 8차 항까지의 합
sage : pn=plot(1/(1-x),x,-1.2,1,legend_label='$1/(1-x)$')
sage : p2=plot(s2,x,-1.2,1,color='black',linestyle='--',legend_label='$s_2=1+x+x^2$')
sage : p5=plot(s5,x,-1.2,1,color='red',linestyle='-.',legend_label='$s_5=1+x+...+x^5$')
sage : p8=plot(s8,x,-1.2,1,color='green',linestyle=':',legend_label='$s_8=1+x+...+x^8$')
sage : (pn+p2+p5+p8).show(ymax=5)
```

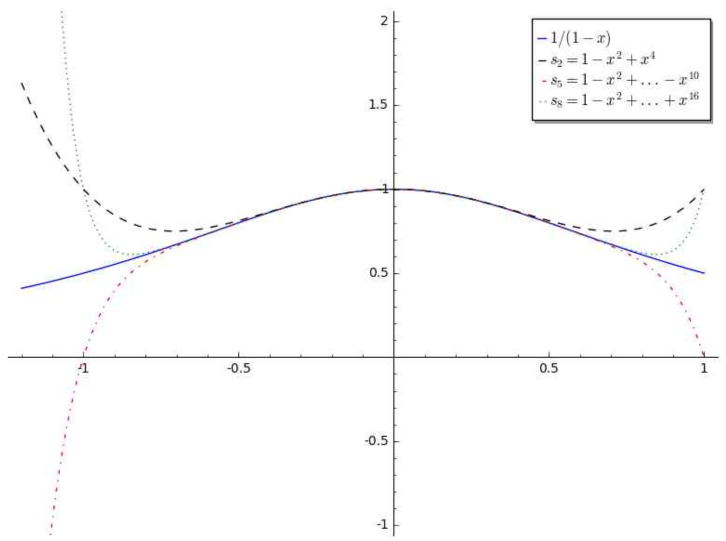


위의 그래프를 살펴보면, 수렴 구간 안에서 테일러 다항식 T_n의 차수가 올라갈수록 T_n이 $f(x)$에 점점 더 가까워짐을 알 수 있다.

다음 급수에 대해서도 비슷한 현상이 발생함을 확인하자.

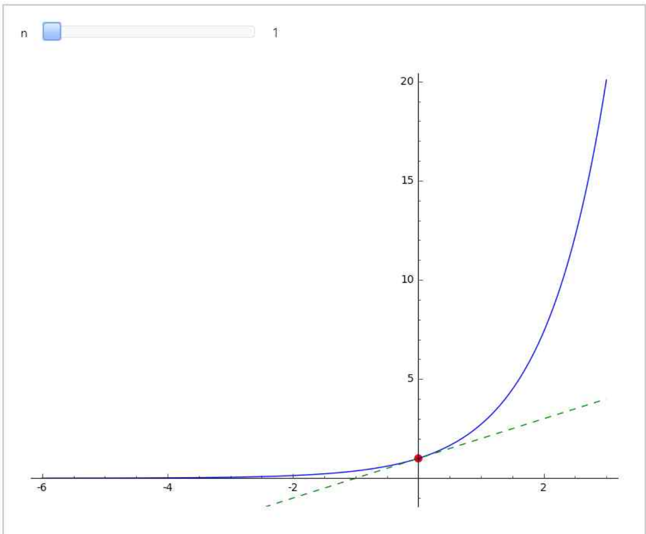
$$\frac{1}{1+x^2}=\frac{1}{1-(-x^2)}=1+(-x^2)+(-x^2)^2+\cdots=1-x^2+x^4+\cdots$$

```
sage : var('n,x')
sage : forget()      # 이전의 assume을 제거한다.
sage : assume(abs(x)<1)
sage : sn=sum((-x^2)^n, n, 0, oo)
sage : s2=sum((-x^2)^n, n, 0, 2)      # 2차 항까지의 합
sage : s5=sum((-x^2)^n, n, 0, 5)      # 5차 항까지의 합
sage : s8=sum((-x^2)^n, n, 0, 8)      # 8차 항까지의 합
sage : pn=plot(1/(1+x^2),x,-1.2,1,legend_label='$1/(1+x^2)$')
sage : p2=plot(s2,x,-1.2,1,color='black',linestyle='--',legend_label='$s_2=1-x^2+x^{4}$')
sage : p5=plot(s5,x,-1.2,1,color='red',linestyle='-.',legend_label='$s_5=1-x^2+...-x^{10}$')
sage : p8=plot(s8,x,-1.2,1,color='green',linestyle=':',legend_label='$s_8=1-x^2+...+x^{16}$')
sage : (pn+p2+p5+p8).show(ymin=-1,ymax=2)
```



함수 $f(x)=e^x$의 그래프를 그려보고, f의 테일러 다항식의 차수가 올라가면서 어떤 현상이 일어나는지 관찰해보자.

```
sage : var('x')
sage : x0 = 0
sage : f = exp(x)
sage : p = plot(f,-6,3)
sage : dot = point((x0,f(x=x0)),pointsize=60,rgbcolor=(1,0,0))
sage : @interact
sage : def _(n=(1..12)):
sage :     ft = f.taylor(x,0,n)
sage :     pt = plot(ft,-6, 3, color='green',linestyle='--')
sage :     show(dot + p + pt, ymin = -1, ymax = 20)
```

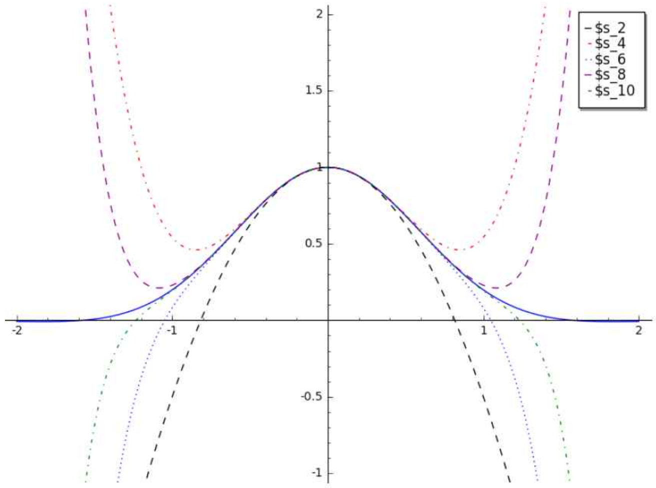


이번에는 조금 더 복잡한 함수 $f(x)=e^{-x^2}\cos x$에 대하여 $f(x)$의 그래프와 테일러 다항식의 그래프를 한 화면에 그려보자.

```
sage : var('x')
sage : x0  = 0
sage : f   = exp(-x^2)*cos(x)
sage : A=[(f).taylor(x,x0,n) for n in range(2,11,2)]
sage : table(A, frame=True, align='center')
```

$-\frac{3}{2}x^2+1$	$\frac{25}{24}x^4-\frac{3}{2}x^2+1$	$-\frac{331}{720}x^6+\frac{25}{24}x^4-\frac{3}{2}x^2+1$	$\frac{1979}{13440}x^8-\frac{331}{720}x^6+\frac{25}{24}x^4-\frac{3}{2}x^2+1$	$-\frac{19093}{518400}x^{10}+\frac{1979}{13440}x^8-\frac{331}{720}x^6+\frac{25}{24}x^4-\frac{3}{2}x^2+1$

```
sage : p=plot(f,-2,2)
sage : p1=plot(A[0],-2,2,color='black',linestyle='--',legend_label='$s_2')
sage : p2=plot(A[1],x,-2,2,color='red',linestyle='-.',legend_label='$s_4')
sage : p3=plot(A[2],x,-2,2,color='blue',linestyle=':',legend_label='$s_6')
sage : p4=plot(A[3],x,-2,2,color='purple',linestyle='--',legend_label='$s_8')
sage : p5=plot(A[4],x,-2,2,color='green',linestyle='-.',legend_label='$s_10')
sage : (p+p1+p2+p3+p4+p5).show(ymin='-1',ymax='2')
```



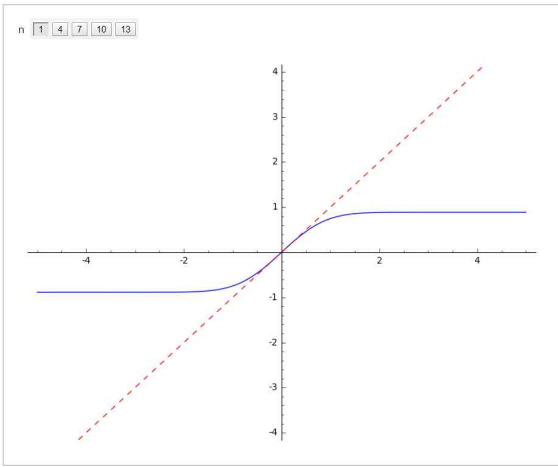
거듭제곱 급수는 직접 적분하기 어려운 함수의 적분을 근사적으로 계산하는데 유용하게 쓸 수 있다. 부정 적분 $\int e^{-x^2}dx$와 정 적분 $\int_0^1 e^{-x^2}dx$의 예를 살펴보자.

```
sage : var('x')
sage : f(x)=exp(-x^2)
sage : integral(exp(-x^2),x)
      1/2*sqrt(pi)*erf(x)
```

부정 적분 $\int e^{-x^2}dx$를 초등 함수들로 표현하지 못하므로, 우리가 알지 못하는 함수 **erf(x)**로 표현된 결과가 나온다. 이 때 테일러 다항식을 이용하여 근사적으로 부정 적분을 구하면, 차수가 높아질수록 $\int e^{-x^2}dx$의 그래프와 비슷해짐을 알 수 있다.

```
sage : x0  = 0
sage : int_f=integral(exp(-x^2))
sage : p= plot(int_f,-5,5)
sage : @interact
sage : def _(n=range(1,15,3)):
sage :     ft = f.taylor(x,x0,n)
sage :     int_ft=integral(ft)
```

```
sage :     pt = plot(int_ft,-5, 5, color='red',linestyle='--')
sage :     show(p + pt, ymin = -4, ymax = 4)
```



다음은 정 적분 $\int_0^1 e^{-x^2}dx$의 근사값을 오차 0.001의 범위에서 찾아보자.

```
sage : f(x)=exp(-x^2)
sage : ft = f.taylor(x,0,6)
sage : A=[(f).taylor(x,x0,n) for n in range(2,8)]
sage : table(A, frame=True, align='center')
```

$x \mapsto -x^2+1$	$x \mapsto -x^2+1$	$x \mapsto \frac{1}{2}x^4-x^2+1$	$x \mapsto \frac{1}{2}x^4-x^2+1$	$x \mapsto -\frac{1}{6}x^6+\frac{1}{2}x^4-x^2+1$	$x \mapsto -\frac{1}{6}x^6+\frac{1}{2}x^4-x^2+1$

위의 결과로부터 $f(x)=e^{-x^2}$의 테일러 다항식들이 항상 짝수 차수임을 알 수 있다. 따라서 근사값의 오차를 줄이기 위해 테일러 다항식의 차수를 높일 때, 차수를 1이 아니라 2만큼 증가시킨다.

```
sage : ep=0.001; n=2
sage : a1=integral(f.taylor(x,0,0),x,0,1)
sage : a2=integral(f.taylor(x,0,n),x,0,1)
sage : while True:
sage :     if abs(a1-a2)<=ep: break
sage :     n=n+2
sage :     a1=a2
sage :     a2=integral(f.taylor(x,0,n),x,0,1)
sage : N(a1, digits=4)
       0.7475
```

제 3장

벡터와 공간 기하학

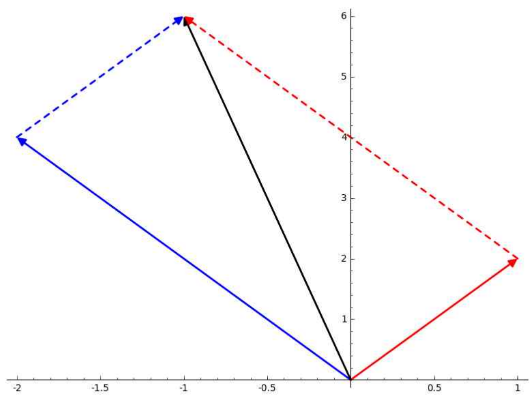
3.1 벡터의 정의와 연산

Sage에서는 n개의 성분들을 갖는 벡터를 **vector([x1, x2, ..., xn])**으로 정의한다.

```
sage : a=vector([1, 2]); b=vector([-2, 4])
sage : a+b     # 벡터의 합
    (-1, 6)
sage : a-b     # 벡터의 차
    (3, -2)
sage : -2*a    # 벡터의 스칼라배
    (-2, -4)
sage : norm(a)     # 벡터의 크기(길이) a.norm( ) 또는 abs(a)로 표현 가능
    sqrt(5)
sage : u=a/norm(a); u     # a와 같은 방향의 단위 벡터
    (1/5*sqrt(5), 2/5*sqrt(5))
```

벡터를 그리려면 명령어 **plot** 또는 **arrow**를 사용한다. **plot**에서는 시점이 항상 원점이지만 **arrow**에서는 시점과 종점을 지정할 수 있다.

```
sage : p=plot(a, color='red') + plot(b) + plot(a+b, color='black')
sage : q=arrow(a,a+b,linestyle='--',color='red')
          + arrow(b,a+b,linestyle='--',color='blue')
sage : p+q
```



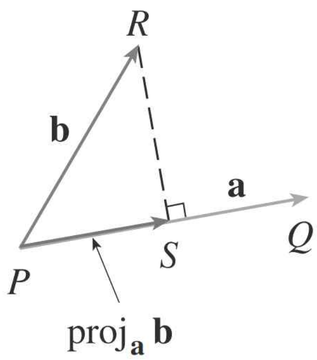
두 벡터 $\vec{a} = < a_1, a_2, \cdots, a_n >$와 $\vec{b} = < b_1, b_2, \cdots, b_n >$의 내적은 다음과 같이 정의된다.

$$\vec{a} \cdot \vec{b} = a_1 b_1 + a_2 b_2 + \cdots + a_n b_n$$

Sage에서는 **a.dot_product(b)** 또는 **a.inner_product(b)**로 계산한다.

```
sage : a=vector([5, -3])
sage : b=vector([4, 6])
sage : a.dot_product(b)
    2
sage : b.dot_product(a)    # 내적은 교환 법칙이 성립한다.
    2
sage : a=vector([2, 2, -1])
sage : b=vector([5, -3, 2])
sage : ab=a.dot_product(b)
sage : an=a.norm()
sage : bn=b.norm()
sage : theta=arccos(ab/(an*bn)); N(theta)    # 벡터 a와 b 사이의 각
    1.46243678131095
```

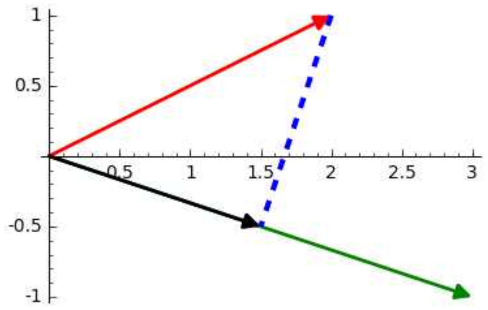
아래 그림에서 두 벡터 $\vec{a}$와 $\vec{b}$의 시점은 모두 P이다. 이 때 R에서 $\overrightarrow{PQ}$를 포함한 직선에 내린 수선의 발을 S라 하면, 벡터 $\overrightarrow{PS}$를 $\vec{a}$ 위로 $\vec{b}$의 정사영(orthogonal projection)이라고 하며 기호로는 $\mathrm{proj}_a b = \left(\dfrac{\vec{a} \cdot \vec{b}}{|\vec{a}|^2}\right)\vec{a}$로 나타낸다.




```
sage : a=vector([3, -1])
sage : b=vector([2, 1])
sage : ab=a.inner_product(b)
sage : aa=a.inner_product(a)
sage : p=ab/aa*a; p
    (3/2, -1/2)
```

이번에는 그래프로 정사영을 표현해 보자.

```
sage : u=plot(a,color='green')+plot(b,color='red')
sage : v=line([b,p],linestyle='--',thickness='3')
sage : w=plot(p,color='black')
```

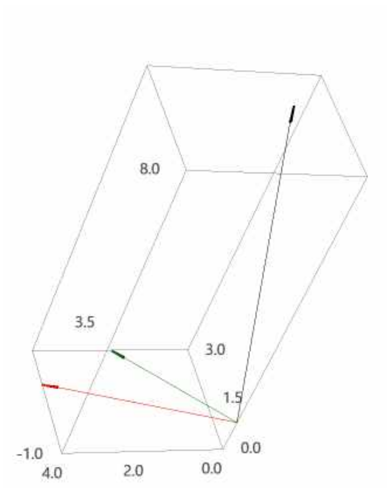


삼차원 벡터 $\vec{a}=< a_1, a_2, a_3 >$와 $\vec{b}=< b_1, b_2, b_3 >$의 외적은 다음과 같이 정의된다.

$$\vec{a}\times\vec{b} =< a_2b_3-a_3b_2,\ a_3b_1-a_1b_3,\ a_1b_2-a_2b_1 >$$

Sage에서는 **a.cross_product(b)**로 계산한다. 아래 그림에서 알 수 있듯이, $\vec{a}\times\vec{b}$는 $\vec{a}$와 $\vec{b}$에 모두 수직이다.

```
sage : a=vector([2, -1, 3]); b=vector([4, -1, 2])
sage : ab=a.cross_product(b);ab
    (1, 8, 2)
sage : ba=b.cross_product(a);ba     # 외적은 교환 법칙이 성립하지 않는다.
    (-1, -8, -2)
sage : u=plot(a,color='green')+plot(b,color='red')
sage : w=plot(ab,color='black')
sage : (u+w).show(aspect_ratio=1)
```



3.2 직선 및 평면의 방정식

3.1.1 직선의 방정식

좌표 평면에서 벡터 $\vec{v} = < a, b >$와 방향이 같고 점 $r_0 = < x_0, y_0 >$를 지나는 직선에 대한 매개 방정식은 다음과 같다.

$$x = x_0 + ta, \quad y = y_0 + tb \qquad (t\text{는 실수})$$

삼차원 공간에서 직선의 매개 방정식은 z성분을 추가하여 다음과 같이 표현한다.

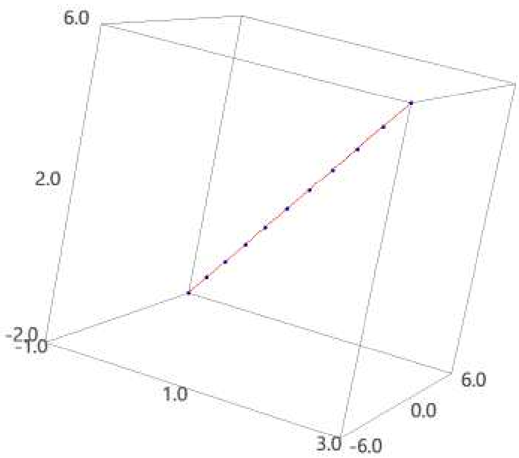
$$x = x_0 + ta, \quad y = y_0 + tb, \quad z = z_0 + tc$$

예를 들어, 점 $(1,0,2)$를 지나고 벡터 $\vec{v} = < 1, 4, -2 >$와 평행한 직선의 매개 방정식을 다음과 같이 구할 수 있다.

```
sage : var('t')
sage : r0=vector([1, 0, 2])
sage : v=vector([1, -3, 2])
sage : L=r0+t*v; L
     (t + 1, -3*t, 2*t + 2)
```

이와 같이 매개 방정식을 구한 다음, **parametric_plot3d**를 사용하여 $-2 \le t \le 2$에 해당하는 선분을 그리고 이를 10등분하여 균일한 간격으로 점들을 찍어본다.

```
sage : p=parametric_plot3d(L, (t, -2, 2),color='red')
sage : x = [-2+4*i/10 for i in [0..10]]     # [-2,2]에서 균일한 간격의 11개 점들
sage : q=point([L(i) for i in x ],color='blue')
sage : (p+q).show(figsize='4')
```



3.2.2 평면의 방정식

점 $P(x_0, y_0, z_0)$를 지나고 벡터 $\vec{n} = < a, b, c >$인 평면의 방정식은 다음과 같다.

$$< a, b, c > \cdot < x - x_0, y - y_0, z - z_0 > = 0$$

또는

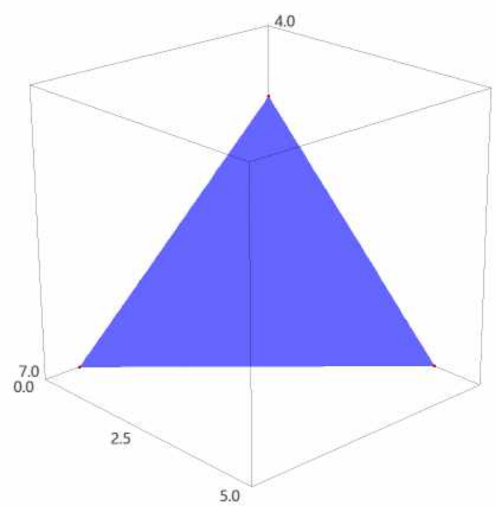
$$a(x - x_0) + b(y - y_0) + c(z - z_0) = 0$$

예를 들어, 점 $(2, 4, -1)$을 지나고 벡터 $\vec{n} = < 2, 3, 4 >$에 수직인 평면의 방정식을 다음과 같이 구할 수 있다.

```
sage : var('x, y, z')
sage : p=vector([x, y, z])
sage : a=vector([2, 4, -1])
sage : n=vector([2, 3, 4])
sage : pl=n.dot_product(p-a); pl     # 평면의 방정식의 좌변
       2*x + 3*y + 4*z - 12
```

또한, 이 평면이 x, y, z축과 만나는 점을 각각 찾고 이를 그림으로 나타내본다.

```
sage : x0=solve((vector([x, 0, 0])-a).dot_product(n)==0,x); x0    # x절편
       [x == 6]
sage : y0=solve((vector([0, y, 0])-a).dot_product(n)==0,y); y0    # y절편
       [y == 4]
sage : z0=solve((vector([0, 0, z])-a).dot_product(n)==0,z); z0    # z절편
       [z == 3]
sage : u=implicit_plot3d(pl==0,(x,0,7),(y,0,5),(z,0,4))
sage : v=point([(6,0,0),(0,4,0),(0,0,3)], color='red', pointsize='10')
sage : (u+v).show(figsize='4')
```



제 4장

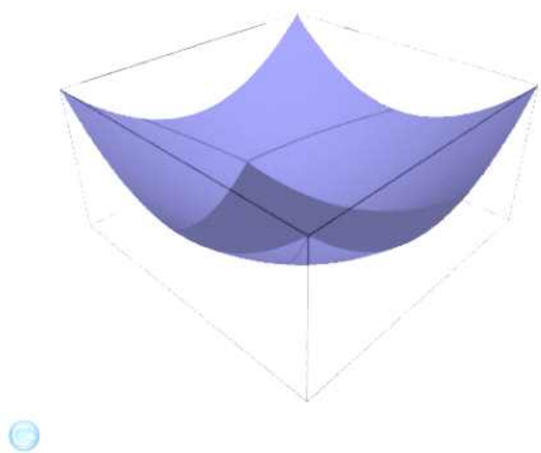
삼차원 공간에서의 그래프

4.1 이변수 함수의 그래프 그리기

직사각형 영역 $[a,b]\times[c,d]$에서 정의된 이변수 함수 $z=f(x,y)$의 그래프를 그리는 명령어는 **plot3d(f(x,y),(x,a,b),(y,c,d))**이다. Sage에서는 기본적으로 x만 변수로 인식하므로, y를 변수로 사용하려면 **var**를 이용하여 y도 변수로 지정해야 한다.

예를 들어, 함수 $f(x,y)=x^2+y^2$의 그래프를 $[-2,2]\times[-2,2]$에서 그리는 명령어와 결과는 다음과 같다.

```
sage : var('x,y')
sage : plot3d(x^2 + y^2, (x,-2,2), (y,-2,2))
```

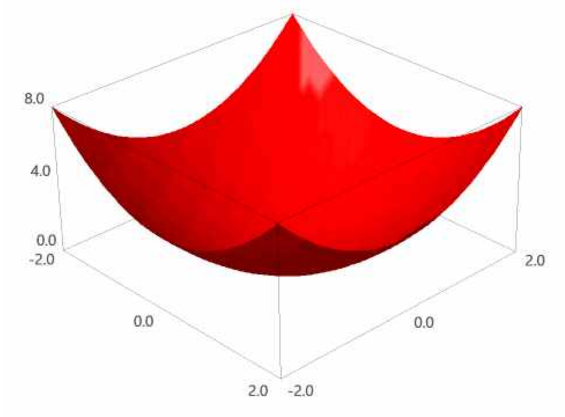


그래의 왼쪽 아랫부분에 있는 클릭하고 그래프에 마우스 우클릭을 하면, 간단한 옵션을 사용할 수 있다.

plot3d는 그래프의 특성과 관련된 다양한 옵션들이 있는데, 이에 대해 알아보자.

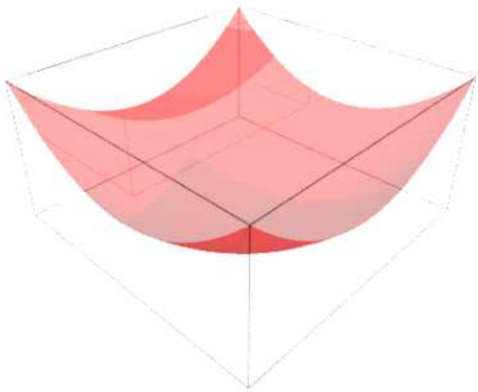
- **color**: 그래프의 색을 지정한다.

```
sage : plot3d(x^2+y^2, (x,-2,2), (y,-2,2), color='red')
```



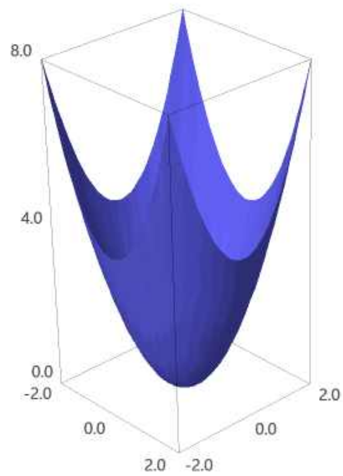
- **opacity**: 0~1 사이의 숫자로 그래프의 투명도를 조절한다.

  sage : **plot3d(x^2+y^2, (x,-2,2), (y,-2,2), color='red', opacity=0.8)**

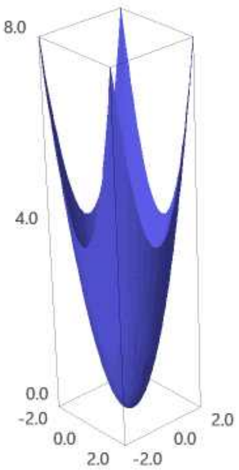


- **aspect_ratio**: (x, y, z)의 비율을 조절한다.

  sage : **plot3d(x^2+y^2, (x,-2,2), (y,-2,2), aspect_ratio=(1,1,1))**

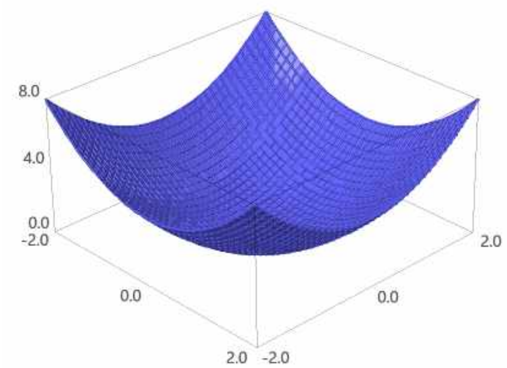


sage : plot3d(x^2+y^2, (x,-2,2), (y,-2,2), aspect_ratio=(1,1,2))



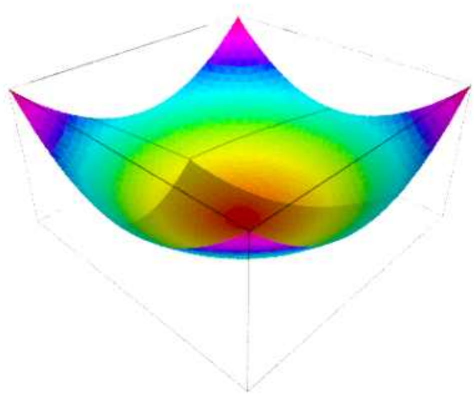
- mesh: 그래프에 격자 모양을 넣는다.

sage : plot3d(x^2 + y^2, (x,-2,2), (y,-2,2), mesh=true)



- adaptive: 그래프의 함숫값이 같은 영역을 같은 색으로 나타낸다.

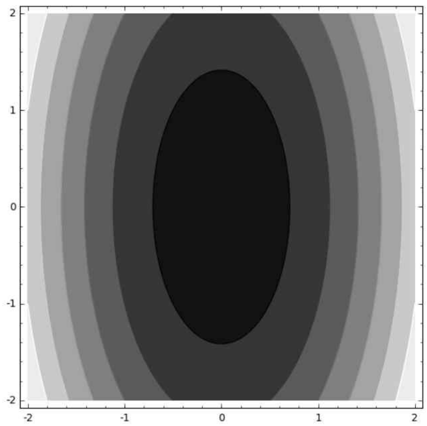
sage : plot3d(x^2 + y^2, (x,-2,2), (y,-2,2), adaptive=true)



4.2 등위 곡선 그리기

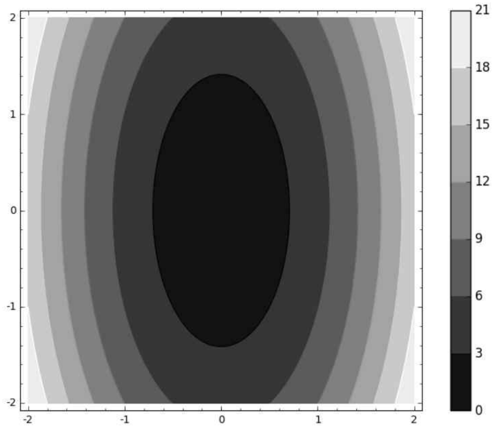
이변수 함수 f의 등위 곡선은 방정식이 $f(x,y)=k$인 곡선이다. 여기에서 k는 f의 치역에 속한 상수이다. sage에서 등위 곡선을 그리는 명령어는 **contour_plot**이다.

```
sage : var('x,y')
sage : f(x,y)=4*x^2+y^2+1
sage : contour_plot(f, (x,-2,2), (y,-2,2))
```



다음은 **contour_plot**과 함께 사용되는 관련된 옵션들이다.

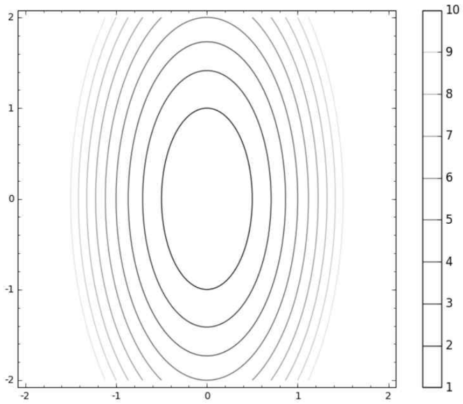
- **colorbar=True** : 음영의 레벨을 나타내어 준다.
  ```
  sage : contour_plot(f, (x,-2,2), (y,-2,2), colorbar=True)
  ```



- **fill=False** : 기본적으로 들어가는 음영색을 제거한다.

● **contours=[list]** : 등위 곡선 $f(x,y)=k$에서 상수 k의 값을 지정한다. 이 때 최소한 세 개 이상의 값을 지정해야 한다.

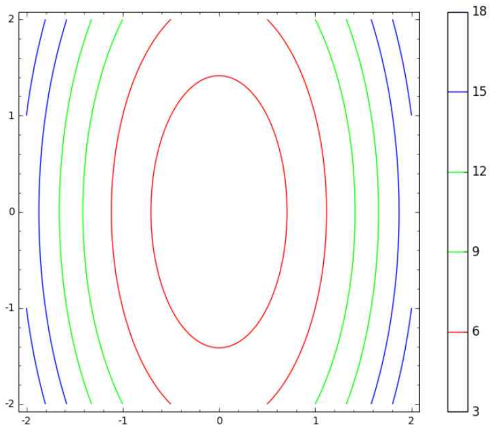
sage : **contour_plot(f, (x,-2,2), (y,-2,2), colorbar=True, fill=False, contours=[1..10])**



옵션 **fill=False**를 사용하여 음영색을 제거하면, 때로는 서로 다른 등위 곡선들을 구분하기 어려우므로 선에 다른 색을 넣는 명령어를 알아보자.

● **cmap=[list]** : 각각의 선에 색을 넣어준다. 이 때 넣어주는 색은 RGB로 지정하거나 내장('hsv', 'cool', 'winter')로 되어있는 명령어를 사용하면 된다.

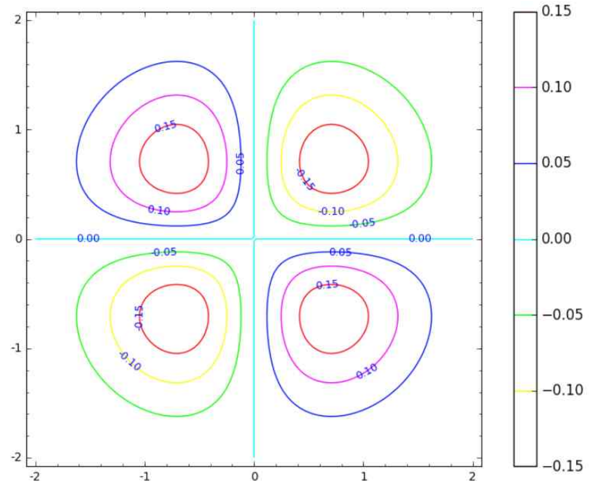
sage : **contour_plot(f, (x,-2,2), (y,-2,2), colorbar=True, fill=False, cmap=[(1,0,0),(0,1,0),(0,0,1)])**



● **labels=True** : 각 등위 곡선 $f(x,y)=k$에 k의 값을 넣어준다.

sage : **f(x,y)=-x*y*exp(-x^2-y^2)**

sage : **contour_plot(f, (x,-2,2), (y,-2,2), colorbar=True, fill=False, cmap='hsv', labels=True)**



이 밖에도 **label_colors='color', label_inline=True** 등 많은 옵션들이 있으니 필요에 따라 추가하면 된다. (sage 명령창에서 **contour_plot?**을 입력해보라.)

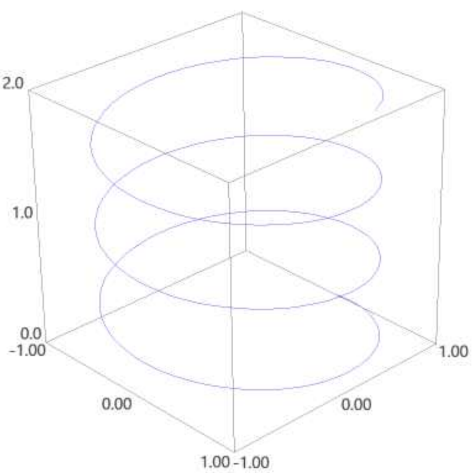
4.3 매개 변수 곡선의 그래프 그리기

삼차원 공간에서의 곡선은 매개 변수 방정식

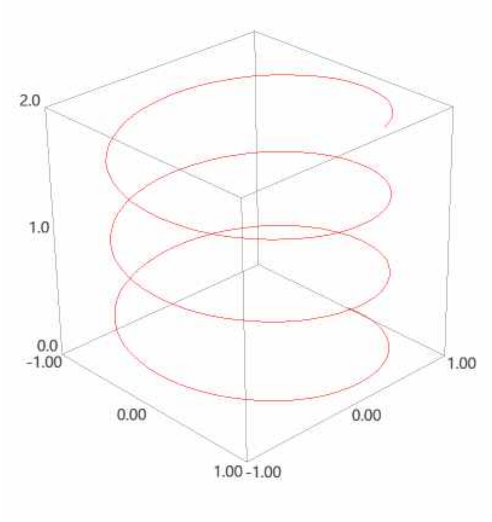
$$x = f(t), \quad y = g(t), \quad z = h(t) \qquad (a \le t \le b)$$

로 나타내어지는데, sage에서는 명령어 **parametric_plot3d**를 사용하여 그릴 수 있다. 이 때 plot3d에서 쓰인 옵션들이 동일하게 적용된다. 특히, **rgbcolor**는 색을 바꾸는 명령어로, **(red,green,blue)**를 0~1사이의 숫자로 지정할 수 있다.

```
sage : var('t')
sage : parametric_plot3d((sin(t), cos(t), t/10), (t,0,20))
```

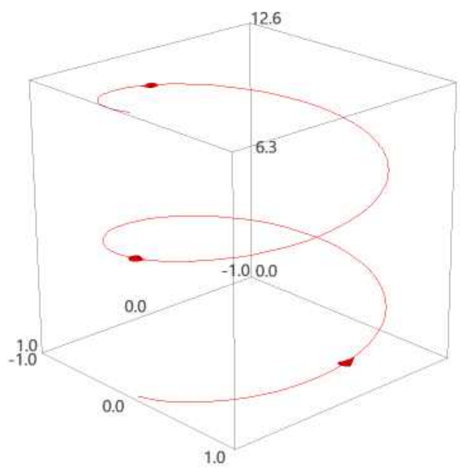


```
sage : parametric_plot3d((sin(t), cos(t), t/10), (t,0,20), rgbcolor=(1,0,0))
```



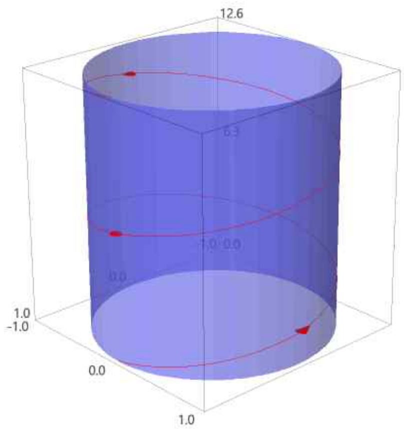
이번에는 매개 변수 곡선 $x = \cos t,\ y = \sin t,\ z = t$의 그래프를 매개 변수 t가 증가하는 방향과 함께 그려보자. (**arrow3d**에 대한 설명은 아래 4.5절을 참조)

```
sage : var('t')
sage : f(t)=cos(t); g(t)=sin(t); h(t)=t
sage : a=0; b=4*pi
sage : r=vector((f,g,h))
sage : C=parametric_plot3d(r, (t, a, b), color='red')
sage : Dt=0.1
sage : Ar1=arrow3d(r(t=a+b*(1/8)),r(t=a+b*(1/8)+Dt),color='red')
sage : Ar2=arrow3d(r(t=a+b*(4/8)),r(t=a+b*(4/8)+Dt),color='red')
sage : Ar3=arrow3d(r(t=a+b*(7/8)),r(t=a+b*(7/8)+Dt),color='red')
sage : Arr=Ar1+Ar2+Ar3
sage : C+Arr
```



이 곡선은 $x^2+y^2=\sin^2 t+\cos^2 t=1$인 원기둥에 놓이는데, $z=t$의 값이 커지면서 반시계 방향으로 소용돌이 모양으로 휘감으면서 위로 올라간다.

```
sage : D=implicit_plot3d(x^2+y^2==1,(x,-1,1),(y,-1,1),(z,0,4*pi),opacity=0.8)
sage : (C+D+Arr).show(figsize='4')
```

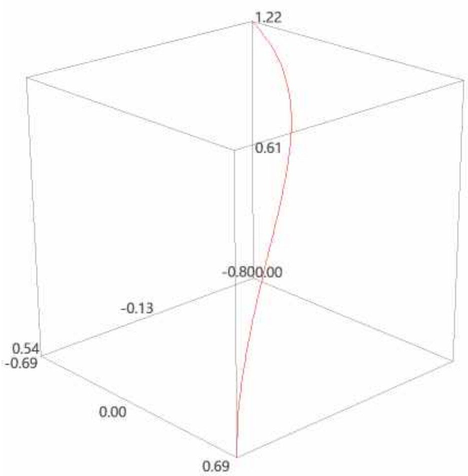


마지막으로, 곡선 $x=\cos t$, $y=\ln(3-t)$, $z=\sqrt{t-1}$의 그래프를 그려보자.

```
sage : var('t')
sage : f(t)=cos(t); g(t)=log(3-t); h(t)=sqrt(t-1)
sage : parametric_plot3d((f(t), g(t), h(t)), (t, 0, 2), color='red')
```

이러면 복잡한 오류 메시지가 나타난다. 함수의 그래프를 그릴 때는 항상 정의역 범위를 생각하면서 그려야 한다. 위의 각 성분 함수들의 정의역의 교집합은 $[1,3)$이므로, 매개 변수 t값의 범위를 이 구간 내로 지정해야한다.

```
sage : parametric_plot3d((f(t), g(t), h(t)), (t, 1, 2.5), color='red')
```

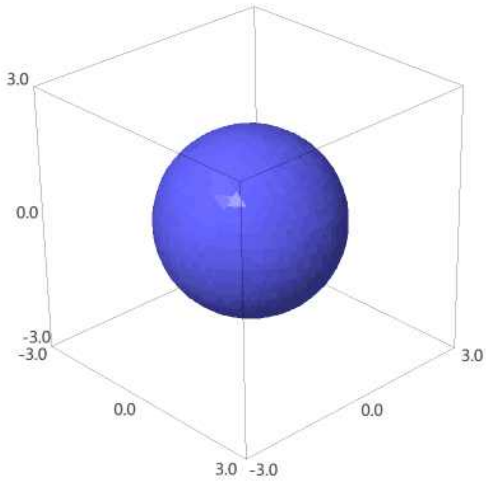


4.4 음함수의 삼차원 그래프 그리기

삼차원에서도 음함수 형태로 되어 있는 식 $F(x,y,z)=k$의 그래프를 그릴 수 있다. (단, k는 상수) 이 때 사용되는 명령어는 **implicit_plot3d**이다.

예를 들어, 구 $x^2+y^2+z^2=4$의 그래프를 그려보자.

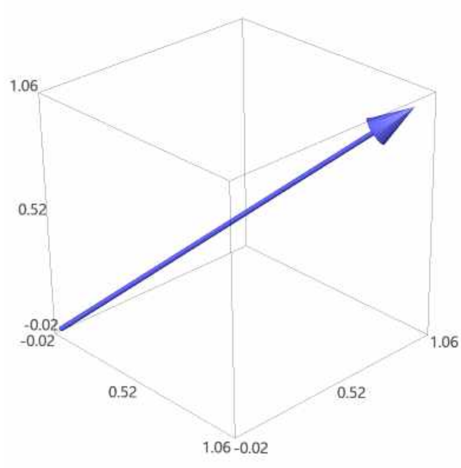
```
sage : var('x,y,z')
sage : implicit_plot3d(x^2+y^2+z^2==4, (x,-3,3), (y,-3,3), (z,-3,3))
```



4.5 그 밖의 삼차원 그래프 그리기

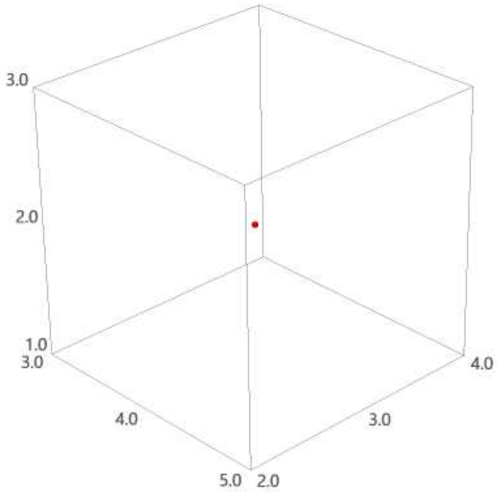
- **arrow3d**: 벡터를 그리는 명령어이다.

```
sage : arrow3d((0,0,0),(1,1,1))
```



● **point3d**: 점을 그리는 명령어이다.

```
sage : point3d((4,3,2), size=10, color='red')
```



제 5장

편도함수

5.1 편도함수

다변수 함수의 미분은 일변수 함수에서 대하여 사용했던 명령어 **diff**를 그대로 쓰면 되는데, 이 때 어떤 변수에 관하여 미분하는지 지정해야 한다.

```
sage : var('x,y')
sage : f(x,y)=4-x^2-2*y^2
sage : fx = diff(f(x,y),x); fx
     -2*x
sage : fx.substitute(x=1,y=1)
     -2
sage : fy = diff(f(x,y),y); fy
     -4*y
sage : fy.substitute(x=1,y=1); fy
     -4
```

이변수 함수 $f(x,y)$를 x와 y에 관하여 여러 번 편미분하여 얻는 고계 편도함수들은 다음과 같이 찾을 수 있다.

```
sage : f(x,y)=exp(x*y^2)
sage : diff(f(x,y),x,2)      # x에 관해 2번 편미분함.
     y^4*e^(x*y^2)
sage : diff(f(x,y),y,y)      # y에 관해 2번 편미분함.
     4*x^2*y^2*e^(x*y^2) + 2*x*e^(x*y^2)
sage : diff(f(x,y),x,y)      # x에 관해 1번, y에 관해 1번 편미분함.
     2*x*y^3*e^(x*y^2) + 2*y*e^(x*y^2)
sage : diff(f(x,y),x,2,y)    # x에 관해 2번, y에 관해 1번 편미분함.
     2*x*y^5*e^(x*y^2) + 4*y^3*e^(x*y^2)
sage : diff(f(x,y),x,2,y,3)  # x에 관해 2번, y에 관해 3번 편미분함.
     8*x^3*y^7*e^(x*y^2) + 60*x^2*y^5*e^(x*y^2) + 96*x*y^3*e^(x*y^2) +
24*y*e^(x*y^2)
```

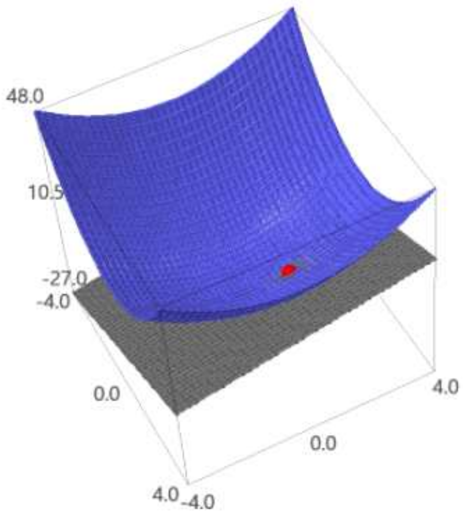
5.2 접평면과 선형 근사

점 $P(x_0, y_0, z_0)$에서 곡면 $z=f(x,y)$에 대한 접평면의 방정식은 다음과 같다.

$$z = f_x(x_0, y_0)(x-x_0)+f_y(x_0, y_0)(y-y_0)+z_0$$

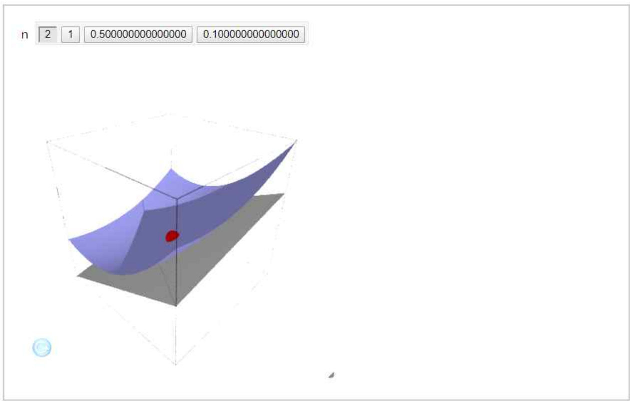
예를 들어, 점 $(1,1,3)$에서 곡면 $z=2x^2+y^2$의 접평면을 찾아보자.

```
sage : f(x,y)=2*x^2+y^2
sage : fx(x,y)=diff(f(x,y),x)
sage : fy(x,y)=diff(f(x,y),y)
sage : g=fx(1,1)*(x-1)+fy(1,1)*(y-1)+3;     # 접평면의 방정식
sage : show(g)
        4x+2y-3
sage : p=plot3d(f,(x,-4,4),(y,-4,4),mesh=true)
sage : q=plot3d(g,(x,-4,4),(y,-4,4),mesh=true,color='gray')
sage : r=point3d((1,1,3),size=20, color= 'red')
sage : (p+q+r).show(figsize='4')
```



다음으로, 접점 $(1,1,3)$근방에서 곡면과 접평면이 얼마나 가까운지 확인해보자.

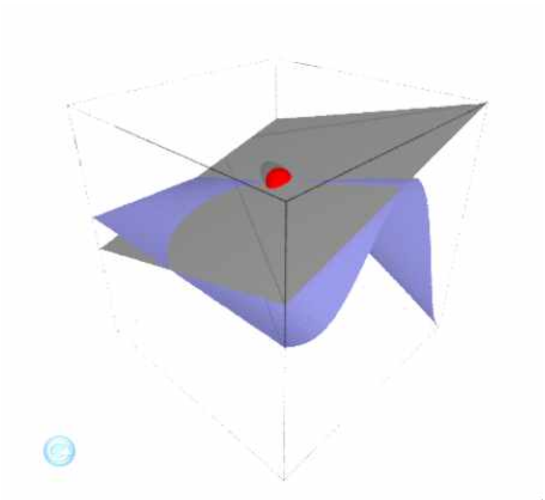
```
sage : @interact
sage : def _(n=[2,1,0.5,0.1]):
sage :     # n은 (x,y)부터 (1,1) 까지의 거리
sage :     p=plot3d(f,(x,1-n,1+n),(y,1-n,1+n),mesh=true)
sage :     q=plot3d(g,(x,1-n,1+n),(y,1-n,1+n),mesh=true,color='gray')
sage :     r=point3d((1,1,3),size=20, color= 'red')
sage :     (p+q+r).show(figsize='4')
```



위의 결과를 보면, 접점에 접근할수록 곡면이 접평면과 가까워지는 것으로 보인다. 즉, 점 (x,y)가 $(1,1)$의 근방에 있을 때 접평면의 방정식 $L(x,y)=4x+2y-3$은 함수 $f(x,y)=2x^2+y^2$에 대한 좋은 근사이다. (이를 f의 선형 근사라고 한다.)

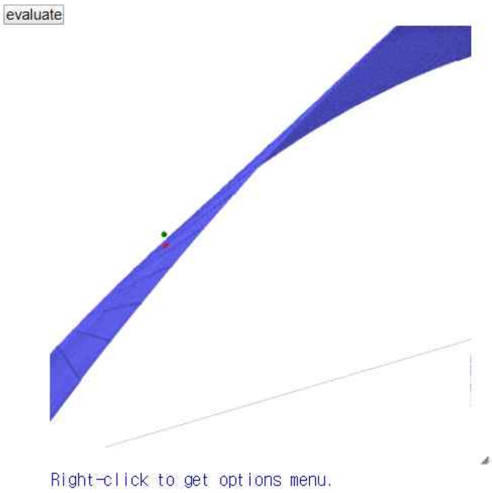
이번에는 점 $(1,1)$에서 함수 $f(x,y)=1-xy\cos\pi y$의 선형 근사를 구하고, 이를 이용하여 $f(1.02,0.97)$의 근사값을 구해보자.

```
sage : f(x,y)=1-x*y*cos(pi*y)
sage : fx(x,y)=diff(f(x,y),x)
sage : fy(x,y)=diff(f(x,y),y)
sage : L(x,y)=fx(1,1)*(x-1)+fy(1,1)*(y-1)+f(1,1)     # 접평면의 방정식
sage : p=plot3d(f,(x,0,2),(y,0,2),mesh=true)
sage : q=plot3d(L,(x,0,2),(y,0,2),mesh=true,color='gray')
sage : r=point3d((1,1,3),size=20, color= 'red')
sage : (p+q+r).show(figsize='4')
```



```
sage : p=plot3d(f,(x,0.9,1.1),(y,0.9,1.1),mesh=true)
sage : s=point3d((1.02,0.97,L(1.02,0.97)),size=10, color= 'green')
```

```
sage : t=point3d((1.02,0.97,f(1.02,0.97)),size=10, color= 'red')
sage : (p+s+t).show(figsize='4')
```




```
sage : L(1.02,0.97); N(f(1.02,0.97))
      1.99000000000000
      1.98500900777829
```

위 그래프는 점을 자세히 보기 위해 옵션을 통해 확대한 것이고, 수치 결과로부터 $f(1.02, 0.97)$의 실제값이 선형 근사에 의한 근사값과 비슷한 것을 확인할 수 있다.

5.3 방향 도함수와 기울기 벡터

점 (x, y)에서 단위 벡터 $\vec{u} = < a, b >$ 방향으로 함수 $f(x, y)$의 방향 도함수는

$$D_{\vec{u}} f(x,y) = f_x(x, y)a + f_y(x, y)b$$

로 나타낼 수 있다. 예를 들어, $f(x, y) = x^3 - 3xy + 4y^2$이고 $\vec{u} = < \frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}} >$일 때 방향 도함수 $D_{\vec{u}} f(2,1)$을 찾아보자.

```
sage : var('x,y')
sage : f = x^3-3*x*y+4*y^2
sage : dx(x,y) = diff(f,x)
sage : dy(x,y) = diff(f,y)
sage : d = vector([dx,dy]);
sage : u = vector([1/sqrt(2),-1/sqrt(2)]);
sage : d(2,1).dot_product(u)
      7/2*sqrt(2)
```

이변수 함수 $f(x,y)$의 기울기 벡터는 다음과 같이 정의된다.

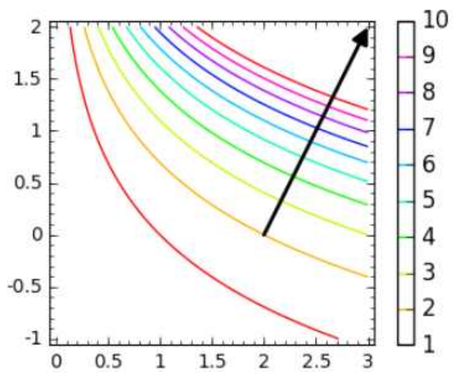
$$\nabla f(x,y) = \ < f_x(x,y),\ f_y(x,y) >$$

함수 $f(x,y)=x^2y^3-4y$의 기울기 벡터를 구하고, 점 $(2,-1)$에서 벡터 $\vec{v}=<2,5>$ 방향으로 f의 방향 도함수를 찾아보자.

```
sage : var('x,y')
sage : f = x^2*y^3-4*y
sage : gradf=vector([diff(f,x),diff(f,y)])
sage : A=vector([2,5])
sage : v=A/A.norm()
sage : B=gradf.substitute(x=2,y=-1).dot_product(v); B
      32/29*sqrt(29)
```

이번에는 곡면 $z=xe^y$의 등위 곡선들과 점 $P(2,0)$에서의 기울기 벡터를 그려 보고, 이를 통해 기울기 벡터가 어떤 의미를 갖는지 알아보자.

```
sage : var('x,y')
sage : f = x*exp(y)
sage : gradf=vector([diff(f,x),diff(f,y)])
sage : B=gradf.substitute(x=2,y=0)
sage : show(B)            # (2,0)에서의 기울기 벡터
sage : show(abs(B))       # 기울기 벡터의 크기가 최대 변화율이다.
      (1,2)
      √5
sage : p=contour_plot(f, (x,0,3), (y,-1,2), colorbar=True, cmap='hsv',
        fill=False, contours=[1..10])
sage : q=arrow((2,0),(3,2))
sage : (p+q).show(figsize='4')
```



위의 그림에서 보듯이, 기울기 벡터는 등위 곡선에 수직이고 점 $P(2,0)$에서의 최대 변화율은 기울기 벡터 방향으로 $\sqrt{5}$이다.

삼변수 함수 F의 기울기 벡터는 등위 곡면 $F(x,y,z)=k$에 수직이다. 따라서, 등위 곡면 위의 점 $P(x_0,y_0,z_0)$에서 접평면의 방정식은

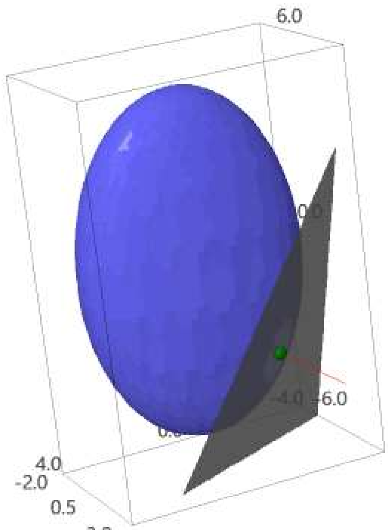
$$\nabla F(x_0,y_0,z_0)\cdot(x-x_0,y-y_0,z-z_0)=0$$

이고, 법선의 매개 변수 방정식은

$$x=F_x(x_0,y_0,z_0)t+x_0,\quad y=F_y(x_0,y_0,z_0)t+y_0,\quad z=F_z(x_0,y_0,z_0)t+z_0$$

이다. 예를 들어, 타원면 $\frac{x^2}{4}+y^2+\frac{z^2}{9}=3$ 위의 점 $(-2,1,-3)$에서 접평면과 법선의 방정식을 찾아보고, 이를 타원면과 함께 그려보자.

```
sage : var('x,y,z,t')
sage : F=x^2/4+y^2+z^2/9
sage : gradF=vector([diff(F,x),diff(F,y),diff(F,z)])    # 기울기 벡터
sage : u=gradF.substitute(x=-2,y=1,z=-3)    # (-2,1,-3)에서의 기울기 벡터
sage : v=vector([x+2,y-1,z+3])
sage : A=u.dot_product(v)==0; A    # 접평면의 방정식
       -x + 2*y - 2/3*z - 6 == 0
sage : L=vector(u*t )+vector((-2,1,-3) ); L    # 법선의 매개 변수 방정식
       (-t - 2, 2*t + 1, -2/3*t - 3)
sage : p=implicit_plot3d(F==3, (x,-4,4), (y,-2,2), (z,-6,6),opacity=0.8)
sage : q=point3d((-2,1,-3),color='green',size=20)
sage : r=implicit_plot3d(A, (x,-4,4), (y,-2,2), (z,-6,6), color='gray',
       opacity=0.8)
sage : s=parametric_plot3d(L,(t,0,1),color='red')
sage : (p+q+r+s).show(aspect_ratio=1)
```

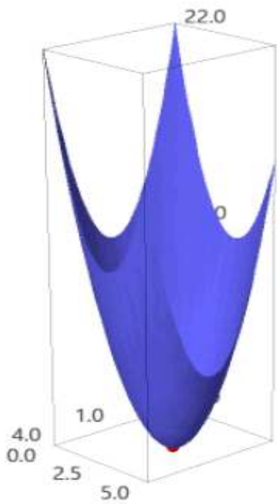


5.4 최댓값과 최솟값

5.4.1 극댓값과 극솟값

먼저 함수 $f(x,y)=x^2+y^2-2x-6y+14$의 임계점들을 구하고, 이를 극소, 극대 또는 안장점으로 분류해보자.

```
sage : f(x,y)=x^2+y^2-2*x-6*y+14
sage : fx(x,y)=diff(f,x)
sage : fy(x,y)=diff(f,y)
sage : solve([fx==0,fy==0],x,y)
      [[x == 1, y == 3]]
sage : f(1,3)
      4
sage : p=plot3d(f,(x,-2,4),(y,0,5),aspect_ratio=1)
sage : q=point3d((1,3,4),size=15,color='red')
sage : p+q
```



함수 f의 임계점은 $(1,3)$뿐인데, 위의 그림에서 확인할 수 있듯이 이는 극소이다.

$f_x(a,b)=f_y(a,b)=0$일 때, 이변수 함수 f의 극값을 결정하는 이계 편도함수 판정법은 다음과 같다. (단, $D=f_{xx}(a,b)f_{yy}(a,b)-\{f_{xy}(a,b)\}^2$)

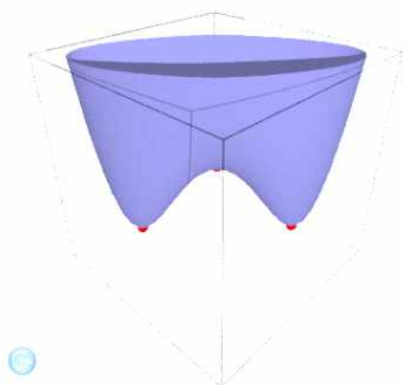
(i) $D>0$ 이고 $f_{xx}(a,b)>0$이면, $f(a,b)$는 극솟값이다.
(ii) $D>0$ 이고 $f_{xx}(a,b)<0$이면, $f(a,b)$는 극댓값이다.
(iii) $D<0$ 이면, $f(a,b)$는 극값이 아니다.

다음과 같이 **def** 명령어를 이용하여 이계 편도함수 판정법을 정의해보자.

```
sage : def Det(f, a, b):
sage :     fxx=diff(f,x,2).substitute(x=a,y=b)
sage :     fyy=diff(f,x,2).substitute(x=a,y=b)
sage :     fxy=diff(diff(f,x),y).substitute(x=a,y=b)
sage :     D=fxx*fyy-fxy^2
sage :     if D>0 and fxx > 0 :
sage :         print( '극솟값이다.' )
sage :     elif D>0 and fxx < 0 :
sage :         print( '극댓값이다.' )
sage :     else:
sage :         print( '안장점이다.' )
```

이를 함수 $f(x,y)=x^4+y^4-4xy+1$에 적용하면 다음 결과를 얻는다. (**solve**를 이용하여 임계점을 구하면 실수해와 허수해가 모두 나오므로 직접 실수해만 찾는다.)

```
sage : f(x,y)=x^4+y^4-4*x*y+1
sage : solve([diff(f,x)==0,diff(f,y)==0],x,y)
    [[x == sqrt(-I), y == -(-1)^(1/4)], [x == -sqrt(-I), y == (-1)^(1/4)],
    [x == (-1)^(1/4), y == (-1)^(3/4)], [x == -I, y == I], [x == I, y ==-I
    ], [x == -1, y == -1], [x == 1, y == 1], [x == 0, y == 0]]
sage : Det(f,0,0)
    안장점이다.
sage : Det(f,1,1)
    극솟값이다.
sage : Det(f,-1,-1)
    극솟값이다.
sage : p=implicit_plot3d(z==x^4+y^4-4*x*y+1,(x,-2,2),(y,-2,2),(z,-2,4),
       mesh=True)
sage : q=point3d([(0,0, f(0,0)),(1,1, f(1,1)),(-1,-1, f(-1,-1))],
       color='red',size=10)
sage : p+q
```



5.4.2 최댓값과 최솟값

R^2에서 폐집합은 모든 경계점을 포함하는 집합이고, 유계 집합은 어떤 원판 내부에 포함되는 집합이다. 일반적으로, 유계인 폐집합 D에서 연속인 함수 f의 최댓값과 최솟값은 다음과 같이 찾을 수 있다.

(i) D의 내부에 속한 f의 임계점들에서 f의 값을 구한다.
(ii) D의 경계에서 f의 최댓값과 최솟값을 구한다.
(iii) (i),(ii)에서 구한 값들 중 가장 큰 것이 최댓값이고, 가장 작은 것이 최솟값이다.

예를 들어, 직사각형 영역 $D=[0,3]\times[0,2]$에서 함수 $f(x,y)=x^2-2xy+2y$의 최댓값과 최솟값을 찾아보자.

(i) D의 내부에 속한 f의 임계점들에서 f의 값을 구한다.

```
sage : var('x,y')
sage : f(x,y) = x^2-2*x*y+2*y
sage : solve([diff(f,x)==0,diff(f,y)==0],x,y)
     [[x == 1, y == 1]]
sage : fv=[]
sage : fv.append(f(1,1)); fv
     [1]
```

(ii) D의 왼쪽 경계 $y=0$에서 $g1(x)=f(x,0)$의 최댓값과 최솟값을 구한다.

```
sage : g1(x)=f(x,0); g1
     x |--> x^2
sage : solve(diff(g1,x)==0,x)
     [x == 0]
sage : fv.append(g1(0))
sage : fv.append(g1(3))
```

D의 오른쪽 경계 $y=2$에서 $g2(x)=f(x,2)$의 최댓값과 최솟값을 구한다.

```
sage : g2(x)=f(x,2); g2
     x |--> x^2 - 4*x + 4
sage : solve(diff(g2,x)==0,x)
     [x == 2]
sage : fv.append(g2(2))
sage : fv.append(g2(0))
```

```
sage : fv.append(g2(3))
sage : fv
    [1, 0, 9, 0, 4, 1]
```

D의 아래쪽 경계 $x=0$에서 $h1(x)=f(0,y)$의 최댓값과 최솟값을 구한다.

```
sage : h1(y)=f(0,y);h1
    y |--> 2*y
sage : fv.append(h1(0))    # 함수 h1가 일차 함수이므로 임계점이 없다.
sage : fv.append(h1(2))
```

D의 윗쪽 경계 $x=3$에서 $h2(x)=f(3,y)$의 최댓값과 최솟값을 구한다.

```
sage : h2(y)=f(3,y);h2
    y |--> -4*y + 9
sage : fv.append(h2(0))    # 함수 h2가 일차 함수이므로 임계점이 없다.
sage : fv.append(h2(2)); fv
    [1, 0, 9, 0, 4, 1, 0, 4, 9, 1]
```

(iii) (i),(ii)에서 찾은 값들 중에서 가장 큰 것과 가장 작은 것을 찾는다.

```
sage : print(max(fv), min(fv))
    (9, 0)
```

그러므로 직사각형 영역 $D=[0,3]\times[0,2]$에서 함수 $f(x,y)=x^2-2xy+2y$의 최댓값은 9이고 최솟값은 0이다.

제 6장

다중 적분

6.1 이중 적분

6.1.1 직사각형 영역에서 이중 적분

직사각형 영역 $R=[a,b]\times[c,d]$에서의 이중 적분은 다음 두 가지의 반복 적분으로 바꾸어 계산할 수 있다.

$$\int\int_R f(x,y)\,dA = \int_a^b\int_c^d f(x,y)\,dy\,dx = \int_c^d\int_a^b f(x,y)\,dx\,dy$$

우변의 반복 적분은 명령어 **integral** 또는 **integrate**를 반복해서 사용하여 구한다. 몇 가지 예를 살펴보자.

$$\int\int_R (x-3y^2)\,dA, \qquad R=[0,2]\times[1,2],$$

```
sage : var('x,y')
sage : integral(integral(x-3*y^2,x,0,2),y,1,2)
    -12
sage : integral(integral(x-3*y^2,y,1,2),x,0,2)
    -12
```

$$\int\int_R y\sin(xy)\,dA, \qquad R=[1,2]\times[0,\pi]$$

```
sage : var('x, y')
sage : f=y*sin(x*y)
sage : integral(integral(f,x,1,2),y,0,pi)
    0
sage : integral(integral(f,y,0,pi),x,1,2)
     -1/2*pi*Ei(2*I*pi) + 1/2*pi*Ei(I*pi) + 1/2*pi*Ei(-I*pi) -
    1/2*pi*Ei(-2*I*pi) + 1/2*pi*gamma(-1, 2*I*pi) - 1/2*pi*gamma(-1, I*pi) -
    1/2*pi*gamma(-1, -I*pi) + 1/2*pi*gamma(-1, -2*I*pi)
```

위와 같은 경우는 같은 값이 나오는 것이 아니라 전혀 다른 형태의 결과가 나온다. 왜 그런지 알아보기 위해 주어진 함수를 y에 대해 적분한 결과를 살펴보자.

```
sage : integral(f,y,0,pi)
    -(pi*x*cos(pi*x) - sin(pi*x))/x^2
```

위의 두 항들 중 앞의 항을 적분한 결과는 다음과 같다.

```
sage : integral(-pi*cos(pi*x)/x,x,1,2)
    -1/2*pi*(Ei(2*I*pi) - Ei(I*pi) - Ei(-I*pi) + Ei(-2*I*pi))
```

여기에서 나오는 Ei(x) 함수는 우리가 아는 초등 함수들에는 속하지 않는다.

```
sage : Ei?
```

이와 같이 동일한 이중 적분이라도 적분 순서에 따라 계산이 훨씬 더 어려워질 수 있기 때문에 적분 순서를 잘 택하는 것이 바람직하다.

6.1.2 일반적인 영역에서 이중 적분

Ⅰ형 : 함수 f가 다음과 같은 평면 영역 D에서 연속 함수라고 하자.

$$D=\{(x,y)\mid a \le x \le b,\ g_1(x) \le y \le g_2(x)\}$$

그러면 $\int\int_D f(x,y)\,dA = \int_a^b \int_{g_1(x)}^{g_2(x)} f(x,y)\,dy\,dx$가 성립한다.

Ⅱ형 : f가 다음과 같은 평면 영역 D에서 연속 함수라고 하자.

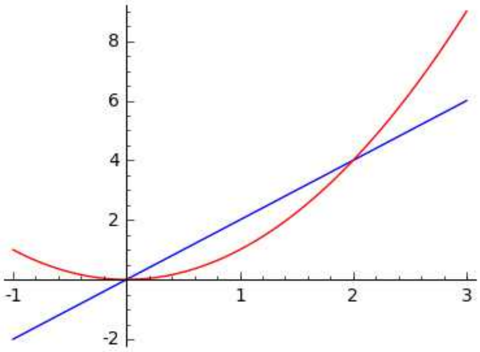
$$D=\{(x,y)\mid c \le y \le d,\ h_1(y) \le x \le h_2(y)\}$$

그러면 $\int\int_D f(x,y)\,dA = \int_c^d \int_{h_1(y)}^{h_2(y)} f(x,y)\,dx\,dy$가 성립한다.

Ⅰ형과 Ⅱ형의 구분은 컴퓨터 프로그램이 하는 것이 아니라 영역 D의 형태를 보고 직접 선택해야 한다.

예를 들어, 포물선 $z=x^2+y^2$ 아래 및 xy 평면에서 직선 $y=2x$와 포물선 $y=x^2$으로 둘러싸인 영역 D 위에 놓여있는 입체의 부피를 찾아보자.

```
sage : var('x, y, z')
sage : f=2*x ; g=x^2
sage : solve(f==g,x)     # 교점을 찾는다.
     [x == 0, x == 2]
sage : p=plot(f,x,-1,3)
sage : q=plot(g,x,-1,3,color=' red' )
sage : (p+q).show(figsize='4')
```

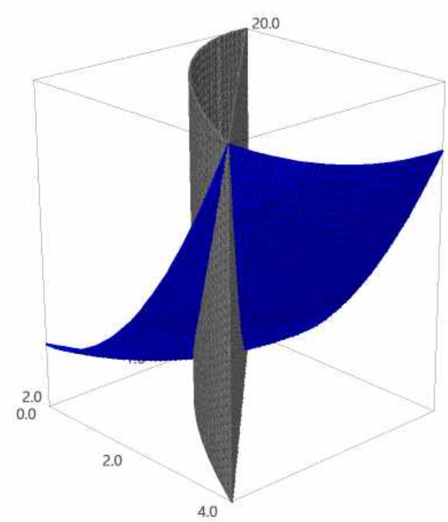


이 그래프에서 볼 수 있듯이, 영역 D는 Ⅰ형도 되고 Ⅱ형도 된다. 여기서는 Ⅰ형만 실습해보고 Ⅱ형은 연습으로 남기도록 한다.

```
sage : integral(integral(x^2+y^2, y,x^2,2*x),x,0,2)
     216/35
```

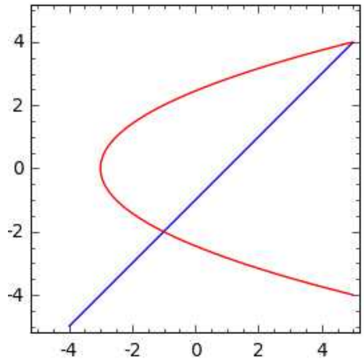
참고로 영역 D 위에 놓여있는 입체를 그리면 다음과 같다.

```
sage : p=implicit_plot3d(y==2*x, (x,0,2),(y,0,4),(z,0,20), color='gray')
sage : q=implicit_plot3d(y==x^2, (x,0,2),(y,0,4),(z,0,20), color='gray')
sage : r=plot3d(x^2+y^2, (x,0,2),(y,0,4), color='blue')
sage : (p+q+r).show(aspect_ratio=(8,4,1),opacity='0.6',mesh=True)
```



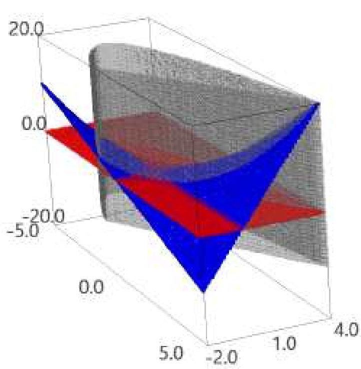
이번에는 D가 직선 $y=x-1$과 포물선 $y^2=2x+6$으로 둘러싸인 영역일 때, 적분 $\int\int_D xy\,dA$를 계산해보자.

```
sage : solve([y==x-1,y^2==2*x+6],x,y)
     [[x == -1, y == -2], [x == 5, y == 4]]
sage : p=implicit_plot(y==x-1,(x,-5,5),(y,-5,5))
sage : q=implicit_plot(y^2==2*x+6,(x,-5,5),(y,-5,5), color='red')
sage : (p+q).show(figsize='4')
```



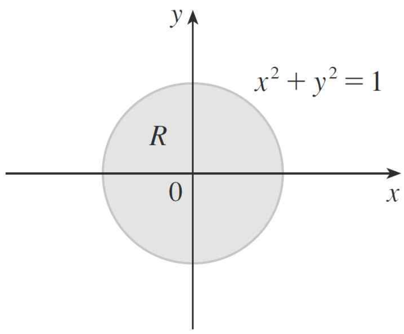
이 예제의 경우는 Ⅰ형보다는 Ⅱ형으로 적분하는 것이 훨씬 쉬워 보인다.

```
sage : integral(integral(x*y, x, (y^2-6)/2,y+1),y,-2,4)
     36
sage : p=implicit_plot3d(y==x-1, (x,-5,5), (y,-2,4), (z,-10,20),
       color='gray', opacity='0.4')
sage : q=implicit_plot3d(y^2==2*x+6, (x,-5,5), (y,-2,4), (z,-10,20),
       color='gray', opacity='0.4')
sage : r=plot3d(x*y, (x,-5,5), (y,-2,4), color='blue')
sage : s=implicit_plot3d(z==0, (x,-5,5), (y,-2,4), (z,-10,20),
       color='red', opacity='0.6')
sage : (p+q+r+s).show(aspect_ratio=(8,4,1),mesh=True)
```

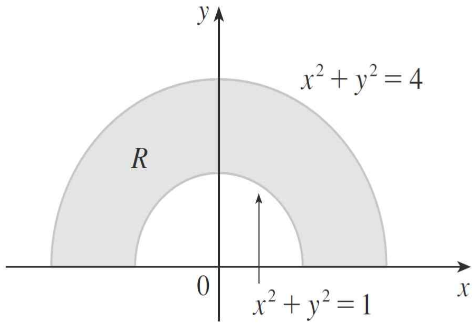


6.1.3 극 좌표에서 이중 적분

이중 적분 $\int\int_R f(x,y)\,dA$에서 영역 R이 아래의 그림들과 같으면, 직교 좌표보다 극 좌표를 이용하여 적분의 영역을 나타내는 것이 편리하다.



(a) $R=\{(r,\theta)\mid 0\le r\le 1,\ 0\le\theta\le 2\pi\}$



(b) $R=\{(r,\theta)\mid 1\le r\le 2,\ 0\le\theta\le\pi\}$

이 때 이중 적분에서 직교 좌표에서 극 좌표로의 변환은 다음과 같다.

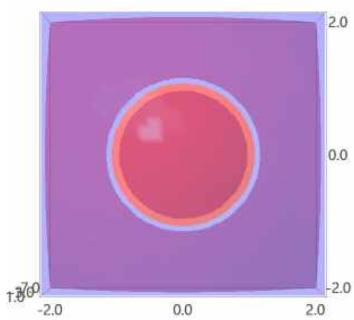
$$\int\int_R f(x,y)\,dA = \int_\alpha^\beta\int_a^b f(r\cos\theta, r\sin\theta)\,r\,dr\,d\theta$$

예를 들어, 위의 오른쪽 그림처럼 R이 위쪽 반평면에서 두 원 $x^2+y^2=1$과 $x^2+y^2=4$로 갇힌 영역일 때, $\int\int_R (3x+4y^2)\,dA$의 값을 찾아보자.

```
sage : var('r, t')
sage : integral(integral((3*r*cos(t)+4*(r*sin(t))^2)*r,r,1,2),t,0,pi)
       15/2*pi
```

이번에는 평면 $z=0$과 포물면 $z=1-x^2-y^2$으로 갇힌 입체의 부피를 찾아보자.

```
sage : p=plot3d(0, (x,-2,2), (y,-2,2), opacity='0.4')
sage : q=plot3d(1-x^2-y^2, (x,-2,2), (y,-2,2), color='red',opacity='0.4')
sage : p+q
```

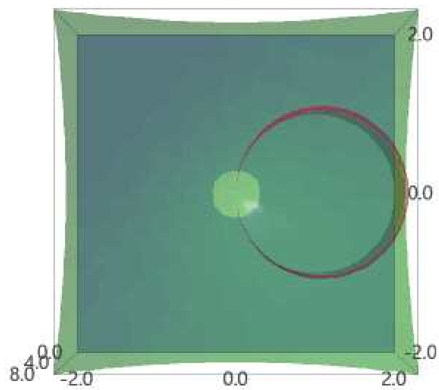


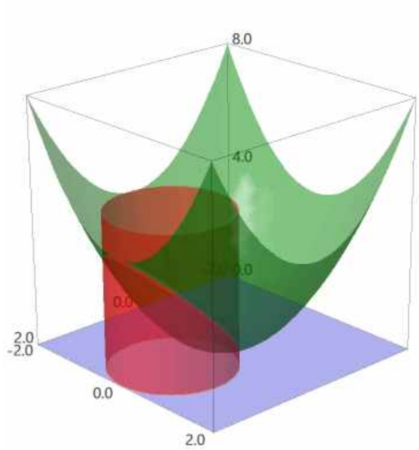
위 그림은 입체를 xy평면 위에서 본 것이다. 원 위에 갇힌 입체의 모양이 나오므로, 극 좌표를 이용하면 부피를 쉽게 구할 수 있다.

```
sage : var('x, y, z, r, t')
sage : f(x,y)=1-x^2-y^2
sage : integral(integral(f(r*cos(t),r*sin(t))*r,r,0,1),t,0,2*pi)
       1/2*pi
```

마지막으로, 포물면 $z=x^2+y^2$ 아래와 xy 평면 위 및 원기둥 $x^2+y^2=2x$의 안쪽에 놓여있는 입체의 부피를 찾아보자.

```
sage : p=plot3d(x^2+y^2, (x,-2,2), (y,-2,2), color='green', opacity='0.4')
sage : q=implicit_plot3d(x^2+y^2==2*x, (x, -2,2), (y, -2,2),(z, 0, 5),
       color='red',opacity='0.4')
sage : p+q
```





그래프를 살펴보면, 적분 영역은 다음과 같음을 알 수 있다.

$$D=\{(r,\theta)|-\pi/2\le\theta\le\pi/2, 0\le r\le 2\cos\theta\}$$

따라서 입체의 부피는 다음과 같은 이중 적분으로 계산된다.

```
sage : f(x,y)=x^2+y^2
sage : integral(integral(f(r*cos(t),r*sin(t))*r,r,0,2*cos(t)),t,-pi/2,pi/2)
       3/2*pi
```

6.2 삼중 적분

6.2.1 직육면체 영역에서 삼중 적분

이중 적분과 비슷하게, 직육면체 $E=\{(x,y,z)\,|\,a\le x\le b,\ c\le y\le d,\ r\le z\le s\}$ 에서의 삼중 적분은 총 6가지의 반복 적분으로 바꾸어 계산할 수 있다.

$$\iiint_E f(x,y,z)\,dV=\int_r^s\int_c^d\int_a^b f(x,y,z)\,dx\,dy\,dz=\cdots\cdots$$

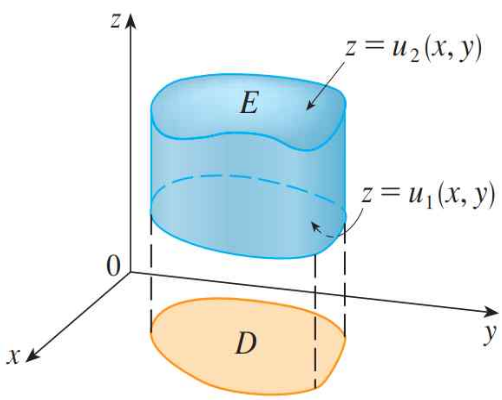
예를 들어, $E=[0,1]\times[-1,2]\times[0,3]$일 때 $\iiint_E xyz^2\,dV$의 값을 찾아보자.

```
sage : var('x,y,z')
sage : f(x,y)=integral(x*y*z^2,(z,0,3)); f      # z 범위에서의 적분
    (x, y) |--> 9*x*y
sage : g(x)=integral(f(x,y),(y,-1,2)); g        # y 범위에서의 적분
    x |--> 27/2*x
sage : h=integral(g,(x,0,1)); h                 # x 범위에서의 적분
    27/4
```

6.2.2 일반적인 영역에서 삼중 적분

Ⅰ형 : 그림과 같이 영역 E가 xy평면으로 정사영이 D로 나타나는 경우이다.

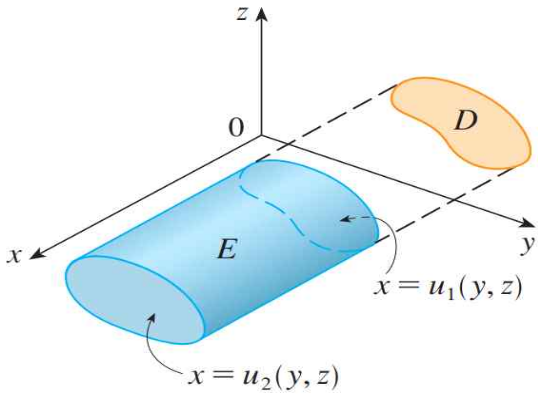
$$E=\{(x,y,z)\mid (x,y)\in D,\ u_1(x,y)\le z\le u_2(x,y)\}$$



여기에서 $\iiint_E f(x,y,z)\,dV=\iint_D\left(\int_{u_1(x,y)}^{u_2(x,y)} f(x,y,z)\,dz\right)dA$가 성립하는데, 영역 D에서의 적분은 이중 적분에서의 방법으로 해결할 수 있다.

Ⅱ형 : 그림과 같이 영역 E가 yz평면으로 정사영이 D로 나타나는 경우이다.

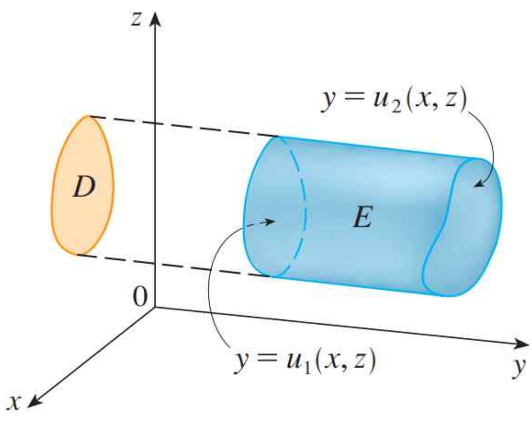
$$E=\{(x,y,z) \mid (y,z)\in D,\ u_1(y,z) \le x \le u_2(y,z)\}$$



이 때 $\iiint_E f(x,y,z)\,dV = \iint_D \left(\int_{u_1(y,z)}^{u_2(y,z)} f(x,y,z)\,dx \right) dA$가 성립한다. 마찬가지로, 영역 D에서의 적분은 이중 적분에서의 방법으로 해결할 수 있다.

Ⅲ형 : 그림과 같이 영역 E가 zx평면으로 정사영이 D로 나타나는 경우이다.

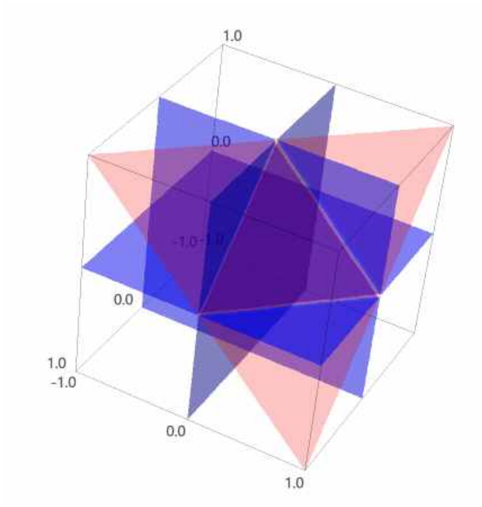
$$E=\{(x,y,z) \mid (x,z)\in D,\ u_1(x,z) \le y \le u_2(x,z)\}$$



이 때 $\iiint_E f(x,y,z)\,dV = \iint_D \left(\int_{u_1(x,z)}^{u_2(x,z)} f(x,y,z)\,dy \right) dA$가 성립한다. 마찬가지로, 영역 D에서의 적분은 이중 적분에서의 방법으로 해결할 수 있다.

예를 들어, E가 네 평면 $x=0$, $y=0$, $z=0$ 및 $x+y+z=1$로 갇힌 사면체일 때, $\iiint_E z\,dV$의 값을 찾아보자. 우선, 사면체가 나타내는 부분을 그려본다.

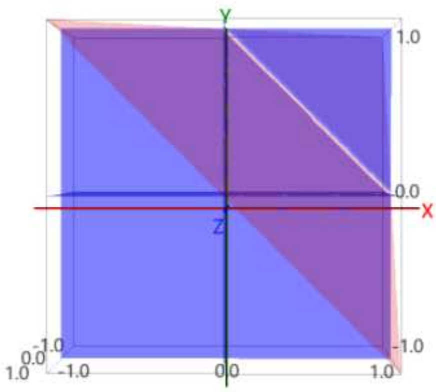
```
sage : p1=implicit_plot3d(x==0, (x,0,1), (y,0,1), (z,-1,1), opacity=0.4,
       color='blue')
sage : p2=implicit_plot3d(y==0, (x,-1,1), (y,-1,1), (z,-1,1), opacity=0.4,
       color='blue')
sage : p3=implicit_plot3d(z==0, (x,-1,1), (y,-1,1), (z,-1,1), opacity=0.4,
       color='blue')
sage : p4=implicit_plot3d(x+y+z==1, (x,-1,1), (y,-1,1), (z,-1,1),
       opacity=0.2, color='red')
sage : p1+p2+p3+p4
```



이 문제는 I, II, III형을 모두 적용할 수 있는데, 여기서는 Ⅰ형만 해본다. 위의 그림을 살펴보면, z의 범위는 $0 \le z \le 1-x-y$임을 확인할 수 있다.

```
sage : var('x,y,z')
sage : f(x,y)=integral(z,(z,0,1-x-y)); f     # z 범위에서의 적분
    (x, y) |--> 1/2*x^2 + (x - 1)*y + 1/2*y^2 - x + 1/2
```

이제 영역 D를 찾아 이 결과를 적분하자. 위에서 얻은 그래프에 마우스 우클릭하여 보기 →XY plane을 실행하면, 아래의 그림을 확인 할 수 있다.



따라서 $D=\{(x,y)\mid 0\le x\le 1,\ 0\le y\le 1-x\}$이고 D 위에서의 이중 적분은 다음과 같이 계산된다.

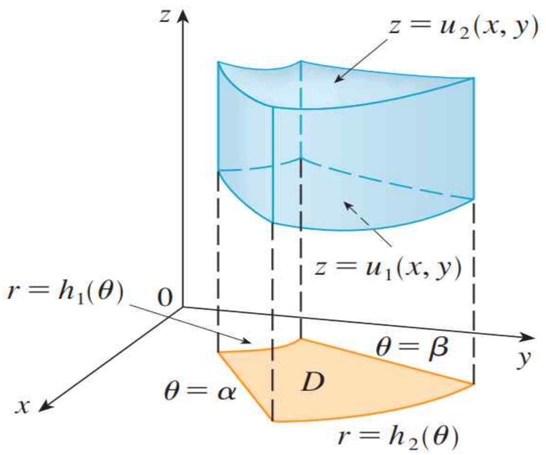
```
sage : g(x)=integral(f(x,y),(y,0,1-x)); g     # y 범위에서의 적분
     x |--> -1/6*x^3 + 1/2*x^2 - 1/2*x + 1/6
sage : h=integral(g,(x,0,1)); h     # x 범위에서의 적분
     1/24
```

6.2.3 원기둥 좌표에서 삼중 적분

원기둥 좌표는 삼차원 공간에서의 점 P를 (r,θ,z)로 표현하는데, 여기에서 r과 θ는 xy 평면 위로의 사영의 극 좌표이고 z는 xy평면에서 점 P까지의 거리이다. 원기둥 좌표와 직교 좌표의 관계는 다음과 같다.

$$x=r\cos\theta,\ \ y=r\sin\theta,\ \ z=z$$

영역 E가 다음 그림과 같은 형태로 나타난다고 하자.



그러면 영역 E에서 x와 y의 범위는 극 좌표로

$$D=\{(r,\theta)\mid \alpha\le\theta\le\beta,\ h_1(\theta)\le r\le h_2(\theta)\}$$

로 나타내어지고, z의 범위는 $u_1(x,y)\le z\le u_2(x,y)$이다. 그러므로, 영역 E 위에서의 삼중 적분은 다음과 같이 쓸 수 있다.

$$\int\int\int_E f(x,y,z)\,dV=\int\int_D\left(\int_{u_1(x,y)}^{u_2(x,y)} f(x,y,z)\,dz\right)dA$$

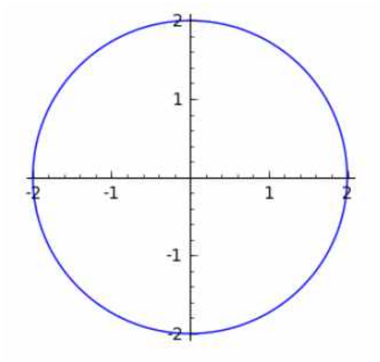
우변의 적분을 구하려면, 먼저 z에 관해 적분한 후 직교 좌표 (x,y)를 극 좌표 (r,θ)로 바꾼 후 그 결과를 극 좌표에 관해 적분하면 된다.

예를 들어, 원기둥 좌표를 이용하여 다음 주어진 삼중 적분의 값을 찾아보자.

$$\int_{-2}^{2}\int_{-\sqrt{4-x^2}}^{\sqrt{4-x^2}}\int_{\sqrt{x^2+y^2}}^{2}(x^2+y^2)\,dzdydx$$

우선 x와 y의 범위를 나타내는 영역 D를 그림으로 그려보자.

```
sage : p=plot(sqrt(4-x^2), x,-2,2)
sage : q=plot(-sqrt(4-x^2), x,-2,2)
sage : (p+q).show(figsize='4',aspect_ratio=1)
```



이로부터 xy 평면에 나타나는 영역이 원 $D=\{(r,\theta)\mid 0\le\theta\le 2\pi,\ 0\le r\le 2\}$임을 알 수 있다. 또한, z의 범위는 $\sqrt{x^2+y^2}=r\le z\le 2$이다. 따라서, 주어진 삼중 적분은 다음과 같이 계산된다.

```
sage : var('r, t')
sage : u(x,y)=x^2+y^2
sage : v(r,t)=u(r*cos(t),r*sin(t))
sage : v=v.simplify_full(); v
      r^2
sage : f(r,t)=integral(v*r, (z, r, 2)); f
      (r, t) |--> -(r - 2)*r^3
sage : g(r)=integral(f(r,t),(t,0,2*pi)); g
      r |--> -2*pi*(r - 2)*r^3
sage : h=integral(g,(r,0,2)); h
       16/5*pi
```

원기둥 좌표에서의 삼중 적분은 직교 좌표로는 계산하기 어려운 적분을 계산하기 쉽게 하기 위해 쓰이는 방법이다. 하지만 Sage에서는 원기둥 좌표로 바꾸지 않고도 직접 계산하는 것이 가능하다.

```
sage : var('x, y, z')
sage : f(x,y)=integral(x^2+y^2,(z,sqrt(x^2+y^2),2)); f
      (x, y) |--> -(x^2 + y^2)*(sqrt(x^2 + y^2) - 2)
```

```
sage : g(x)=integral(f,(y,-sqrt(4-x^2),sqrt(4-x^2))); dg
    x |--> -3/4*x^4*arcsinh(sqrt(x^2)*sqrt(-x^2 + 4)/x^2) + 1/6*(7*x^2
    +8)*sqrt(-x^2 + 4)
sage : h(x)=integral(g,(x,-2,2)); h
    16/5*pi
```

6.2.4 구면 좌표에서 삼중 적분

구면 좌표는 삼차원 공간의 점 P를 (ρ, θ, ϕ)로 나타내는데, ρ는 원점 O와 P 사이의 거리를 나타내고, θ는 원기둥 좌표와 같은 각이며 ϕ는 양의 z축과 선분 $\overline{OP}$ 사이의 각이다. 구면 좌표와 직교 좌표의 관계는 다음과 같다.

$$x = \rho\sin\phi\cos\theta,\ \ y = \rho\sin\phi\sin\theta,\ \ z = \rho\cos\phi$$

영역 E가 구면 쐐기 $E = \{(\rho, \theta, \phi) \mid a \le \rho \le b,\ \alpha \le \theta \le \beta,\ c \le \phi \le d\}$의 형태일 때, E 위에서의 삼중 적분은 다음과 같이 쓸 수 있다.

$$\int\int\int_E f(x, y, z)\,dV = \int_c^d\int_\alpha^\beta\int_a^b f(\rho\sin\phi\cos\theta, \rho\sin\phi\sin\theta, \rho\cos\phi)\rho^2\sin\phi\, d\rho\, d\theta\, d\phi$$

예를 들어, $B = \{(x, y, z) \mid x^2 + y^2 + z^2 \le 1\}$일 때 삼중 적분 $\int\int\int_B e^{(x^2+y^2+z^2)^{3/2}}\,dV$의 값을 찾아보자. 영역 B를 구면 좌표로 나타내면 다음과 같다.

$$B = \{(\rho, \theta, \phi) \mid 0 \le \rho \le 1,\ 0 \le \theta \le 2\pi,\ 0 \le \phi \le \pi\}$$

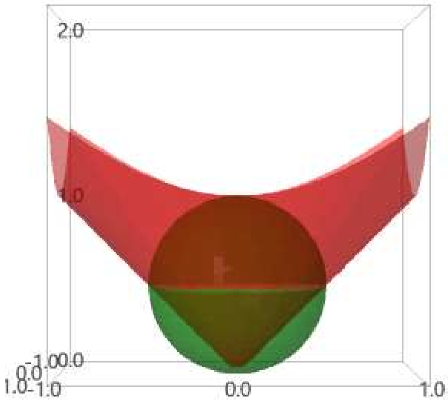
따라서 위의 공식에 의해 주어진 삼중 적분은 다음과 같이 계산된다.

```
sage : var('r,s,t')
sage : f=exp((x^2+y^2+z^2)^(3/2))
sage : F=f(r*sin(s)*cos(t),r*sin(s)*sin(t),r*cos(s)).simplify_full()
sage : int_rho=integral(F*r^2*sin(s),r,0,1)
sage : int_theta=integral(int_rho,t,0,2*pi)
sage : int=integral(int_theta,s,0,pi); int
    4/3*pi*(e - 1)
```

다음으로, 원뿔 $z = \sqrt{x^2 + y^2}$ 위와 구면 $x^2 + y^2 + z^2 = z$ 아래 놓여 있는 입체의 부피를 찾아보자. 먼저 이 입체를 그림으로 그리면 다음과 같다.

```
sage : p=implicit_plot3d(z==sqrt(x^2+y^2), (x,-1,1), (y,-1,1), (z,0,2),
       opacity='0.4', color='red')
```

```
sage : q=implicit_plot3d(x^2+y^2+z^2==z, (x,-1,1), (y,-1,1), (z,0,2),
        opacity='0.4', color='green')
sage : (p+q).show(figsize='4')
```



또한, 구면 좌표로 입체를 다음과 같이 나타낼 수 있다 .

$$E=\{(\rho,\theta,\phi)\mid 0\le\rho\le\cos\phi,\ 0\le\theta\le 2\pi,\ 0\le\phi\le\pi/4\}$$

따라서 입체의 부피는 삼중 적분으로 다음과 같이 계산된다.

```
sage : var('r,s,t')
sage : int_rho=integral(1*r^2*sin(s),r,0,cos(s))
sage : int_theta=integral(int_rho,t,0,2*pi)
sage : int=integral(int_theta,s,0,pi/4);int
    1/8*pi
```

6.3 다중 적분에서 변수 변환

직교 좌표를 극 좌표로 변환하는 것과 같이, 적절한 변수 변환

$$T:\quad x=g(u,v),\quad y=h(u,v)$$

을 통해 이중 적분을 간단한 형태로 바꿀 수 있다. 변환 T가 uv 평면의 영역 S를 xy 평면의 영역 R 위로 보내는 사상이고 변환 T의 야코비 행렬식

$$\frac{\partial(x,y)}{\partial(u,v)}=\begin{vmatrix}\frac{\partial x}{\partial u} & \frac{\partial x}{\partial v}\\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v}\end{vmatrix}=\frac{\partial x}{\partial u}\frac{\partial y}{\partial v}-\frac{\partial x}{\partial v}\frac{\partial y}{\partial u}$$

가 영이 아니면, 영역 R 위에서의 이중 적분을 다음과 같이 나타낼 수 있다.

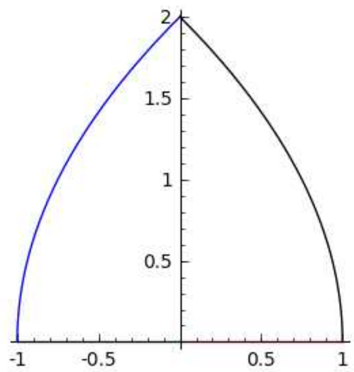
$$\int\int_R f(x,y)\,dA=\int\int_S f(x(u,v),y(u,v))\left|\frac{\partial(x,y)}{\partial(u,v)}\right|du\,dv$$

예를 들어, 변수 변환 $x = u^2 - v^2,\ y = uv$에 대하여 uv 평면의 정사각형 영역

$$S = \{(u, v) \mid 0 \le u \le 1,\ 0 \le v \le 1\}$$

은 xy 평면의 어떤 영역 R로 변환되는지 찾아보자.

```
sage : var('u, v, x, y')
sage : x(u,v)=u^2-v^2; y(u,v)=2*u*v
sage : x1(u)=x(u,0); y1(u)=y(u,0)
sage : x2(u)=x(u,1); y2(u)=y(u,1)
sage : x3(v)=x(0,v); y3(v)=y(0,v)
sage : x4(v)=x(1,v); y4(v)=y(1,v)
sage : p=parametric_plot((x1,y1), (u,0,1), color='red')
sage : q=parametric_plot((x2,y2), (u,0,1), color='blue')
sage : r=parametric_plot((x3,y3), (v,0,1), color='green')
sage : s=parametric_plot((x4,y4), (v,0,1), color='black')
sage : (p+q+r+s).show(figsize='4')
```



위의 그림을 보면, 영역 R은 네 개의 곡선으로 갇혀 있음을 알 수 있다. 검은색과 파란색으로 나타난 곡선들의 방정식을 다음과 같이 찾아보자.

```
sage : var('x, y')
sage : x2; y2
    u |--> u^2 - 1
    u |--> 2*u
sage : x2(y/2)
    1/4*y^2 - 1
sage : x4; y4
    v |--> -v^2 + 1
    v |--> 2*v
sage : x4(y/2)
    -1/4*y^2 + 1
```

검은색과 파란색 곡선들은 방정식이 각각 $x = \frac{y^2}{4} - 1$과 $x = -\frac{y^2}{4} + 1$인 포물선임을 알 수 있다. 이제 R이 x축과 두 포물선 $y^2 = 4 + 4x$, $y^2 = 4 - 4x$로 갇힌 영역일 때, 변수 변환 $x = u^2 - v^2$, $y = uv$를 이용해서 이중 적분 $\int\int_R y\,dA$의 값을 찾아보자.

```
sage : var('u, v, x, y')
sage : x(u,v)=u^2-v^2
sage : y(u,v)=2*u*v
sage : Ja=diff(x,u)*diff(y,v)-diff(x,v)*diff(y,u)
sage : integral(integral(y*Ja,u,0,1),v,0,1)
      2
```