13 Bivariate Functions 二元函数

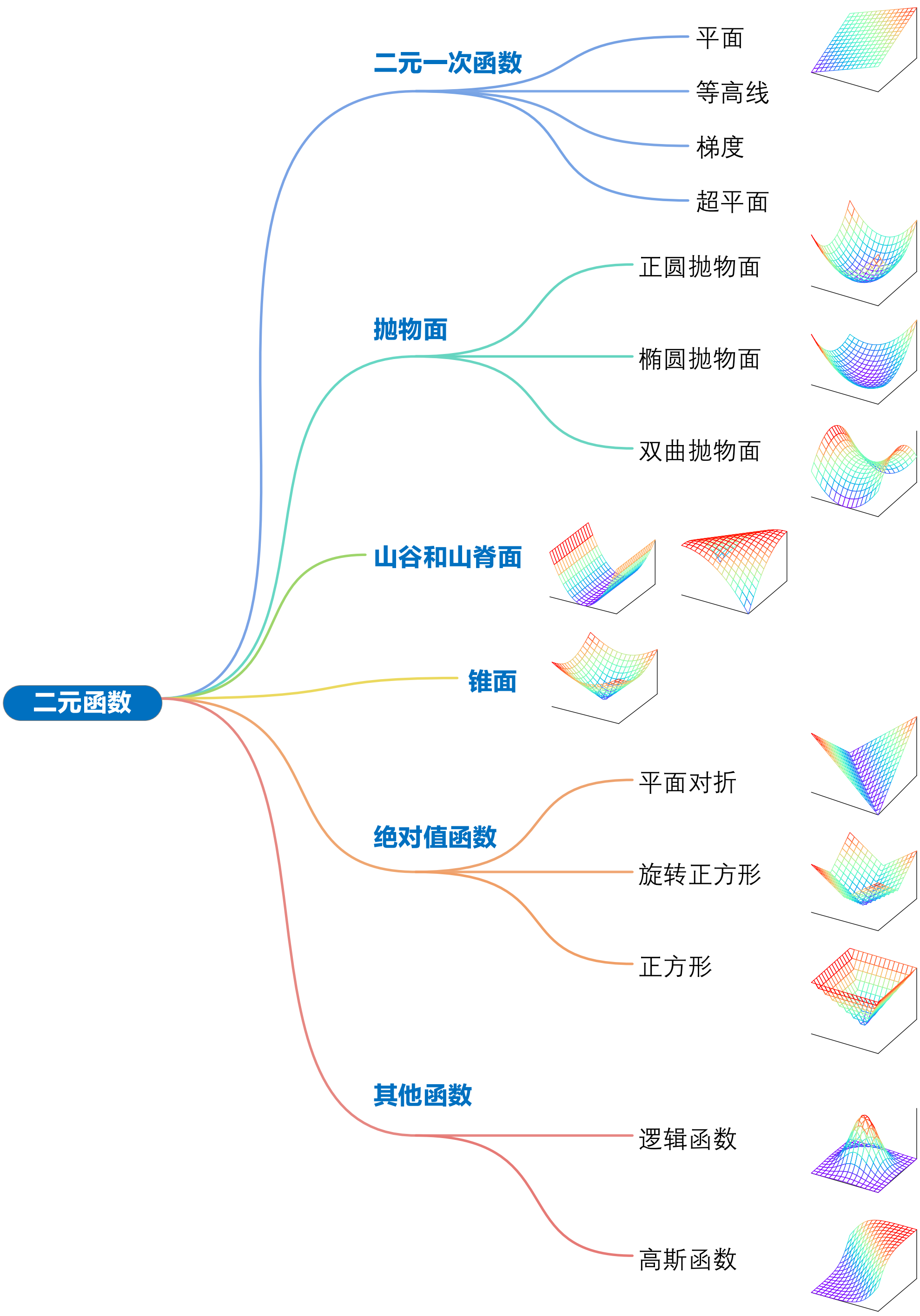
从三维几何图形角度理解

当然，我们可以使用任何需要符号；不要嘲笑符号；发明它们，它们很强大。事实上，很大程度数学就是在发明更好的符号。

We could, of course, use any notation we want; do not laugh at notations; invent them, they are powerful. In fact, mathematics is, to a large extent, invention of better notations.

—— 理查德·费曼 (Richard P. Feynman) | 美国理论物理学家 | 1918 ~ 1988

- ◄ Axes3D.plot_surface() 绘制三维曲面
- ◄ matplotlib.pyplot.contour() 绘制等高线图
- ◄ matplotlib.pyplot.contourf() 绘制填充等高线图
- ◄ numpy.linspace() 在指定的间隔内，返回固定步长的数据
- ◄ numpy.meshgrid() 生成网格数据


二元函数
二元一次函数
平面
等高线
梯度
超平面
抛物面
正圆抛物面
椭圆抛物面
双曲抛物面
山谷和山脊面
锥面
绝对值函数
平面对折
旋转正方形
正方形
其他函数
逻辑函数
高斯函数

13.1 二元一次函数：平面

二元一次函数是一元一次函数的扩展，一般式如下：

$$y = f(x_1, x_2) = w_1 x_1 + w_2 x_2 + b \tag{1}$$

当 w_1 和 w_2 均为 0 时，$f(x_1, x_2) = b$ 为二元常数函数，平行于 x_1x_2 水平面。

用矩阵乘法，(1) 可以写成：

$$y = f(x_1, x_2) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b \tag{2}$$

其中，

$$\boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}, \quad \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \tag{3}$$

当 y 取一定值时，比如 $y = c$，平面蜕化为一条直线：

$$w_1 x_1 + w_2 x_2 + b = c \tag{4}$$

从另外一个角度，c 相当于 $f(x_1, x_2)$ 的某一条等高线；即，$f(x_1, x_2)$ 等高线为直线。

举个例子，图 1 所示图像对应如下解析式：

$$y = f(x_1, x_2) = x_1 + x_2 = \underbrace{\begin{bmatrix} 1 \\ 1 \end{bmatrix}^{\mathrm{T}}}_{\boldsymbol{w}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\boldsymbol{x}} \tag{5}$$

图 1 (a) 所示为 (5) 对应平面，图中黑色直线对应 $x_1 + x_2 = 0$，即 $x_2 = -x_1$；图 1 (b) 所示 $f(x_1, x_2)$ 平面等高线都平行于 $x_1 + x_2 = 0$。由于函数为线性函数，因此等高线平行，且间距相同。

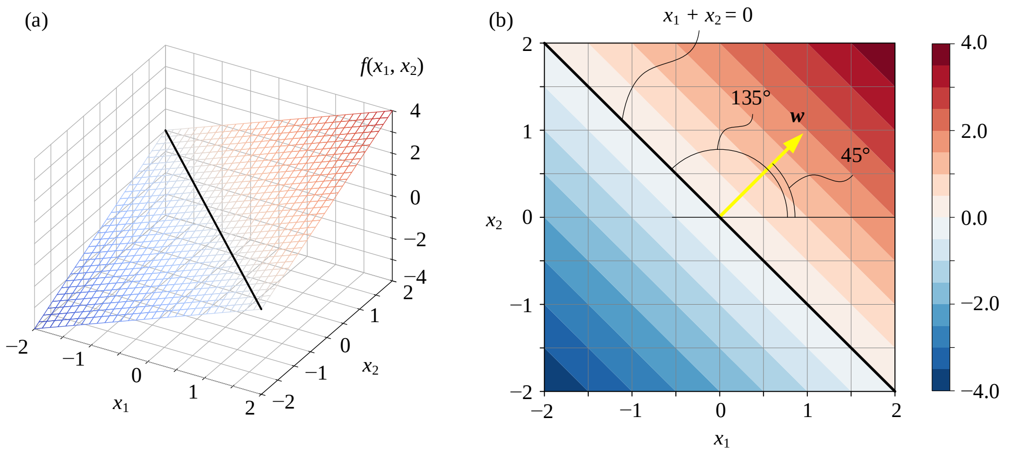


图 1. $f(x_1, x_2) = x_1 + x_2$ 网格图和等高线图

图 1 (b) 中黄色箭头为 $f(x_1, x_2)$ 增大方向，箭头和 x_1 轴正方向夹角为 45°。有心的读者可能发现，黄色箭头对应的向量就是 $\boldsymbol{w}$：

$$\boldsymbol{w} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{6}$$

图 1 (b) 中，$\boldsymbol{w}$ 向量垂直于等高线，指向 $f(x_1, x_2)$ 增大方向。这并非巧合，实际上 $\boldsymbol{w}$ 向量便是**梯度** (gradient)。此外，大家已经意识到向量不仅仅是一列或一行数，还是有方向的线段。

本书在前文讲解不等式时，提到过梯度这个概念，不过当时我们关注的是梯度的反方向，即梯度下降方向。

本系列丛书内容不断深入，大家会理解 $\boldsymbol{w}$ 的几何意义以及梯度向量这一重要概念。这里先给大家留下一个印象。

图 2 所示平面对应解析式如下：

$$y = f\left(x_1, x_2\right) = -x_1 + x_2 = \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\boldsymbol{w}}^{\mathrm{T}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\boldsymbol{x}} \tag{7}$$

图 2 (b) 中黄箭头同样指向 $f(x_1, x_2)$ 增大方向，对应 (7) 中 $\boldsymbol{w}$；箭头和 x_1 轴正方向夹角为 135°。

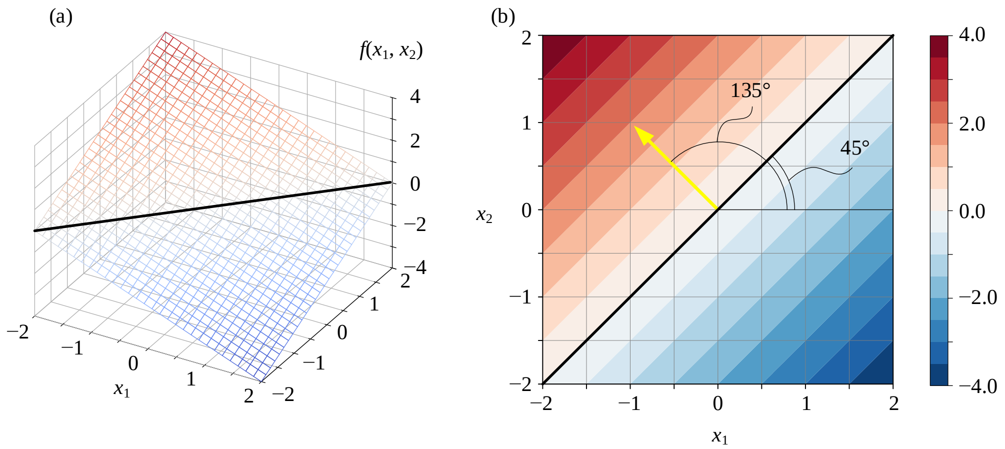


图 2. $f(x_1, x_2) = -x_1 + x_2$ 网格图和等高线图

等高线平行纵轴

当 $w_1 = -1$，$w_2 = 0$，$b = 0$ 时，$f(x_1, x_2)$ 平面高度仅仅受到 x_1 影响。图 3 所示图像对应如下解析式：

$$y = f\left(x_1, x_2\right) = -x_1 = \underbrace{\begin{bmatrix} -1 \\ 0 \end{bmatrix}}_{\boldsymbol{w}}^{\mathrm{T}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\boldsymbol{x}} \tag{8}$$

图 3 (a) 所示平面平行于 x_2 轴；图 3 (b) 所示 $f(x_1, x_2)$ 平面等高线同样平行于 x_2 轴。图 3 (b) 中，黄色箭头为函数 $f(x_1, x_2)$ 增大方向，箭头平行 x_1 轴，朝左，即朝向 x_1 轴负方向。

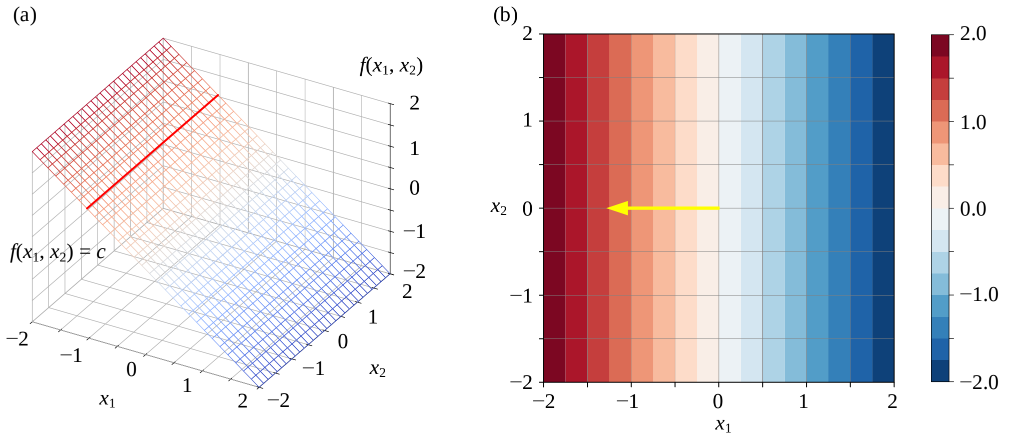


图 3. $f(x_1, x_2) = -x_1$ 网格图和等高线图

等高线平行横轴

当 $w_1 = 0$，$w_2 = 1$，$b = 0$ 时，$f(x_1, x_2)$ 平面仅仅受到 x_2 影响。图 4 所示图像对应如下解析式：

$$y = f\left(x_1, x_2\right) = x_2 = \underbrace{\begin{bmatrix} 0 \\ 1 \end{bmatrix}^{\mathrm{T}}}_{w} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{9}$$

图 4 (a) 所示平面平行于 x_1 轴；图 4 (b) 所示 $f(x_1, x_2)$ 平面等高线同样平行于 x_1 轴。图 4 (b) 中黄色箭头同样为 $f(x_1, x_2)$ 增大方向，箭头指向 x_2 轴正方向。

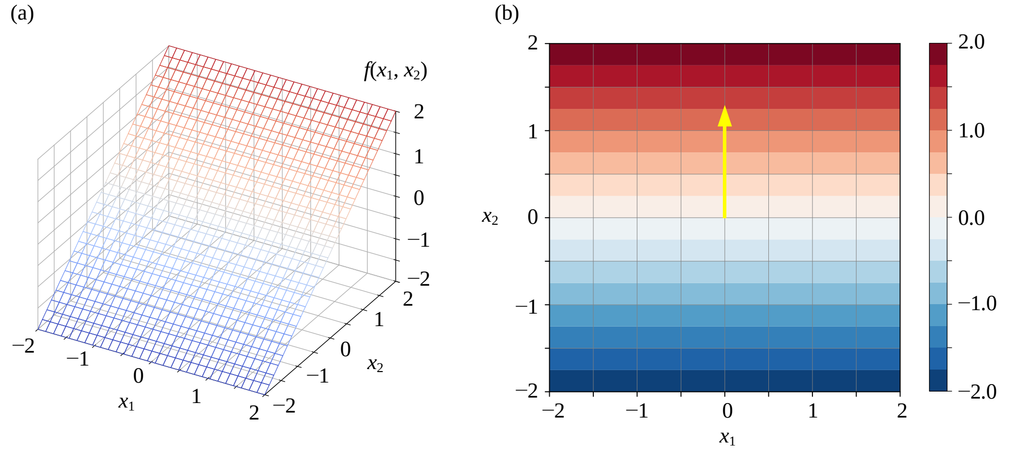


图 4. $f(x_1, x_2) = x_2$ 网格图和等高线图

平面叠加

如图 5 所示，请大家注意，若干平面叠加得到的还是平面；函数 $f_i(x_1, x_2)$ 中下角标 i 为函数序号，不同序号代表不同函数。

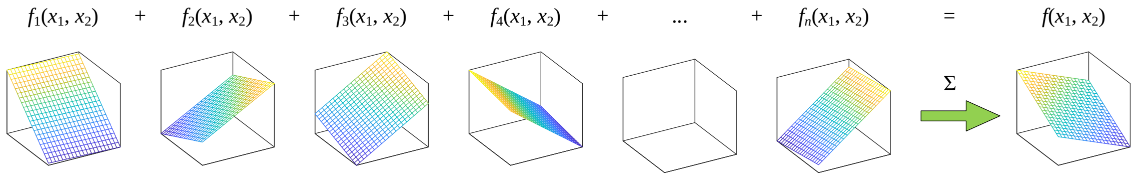


图 5. 若干平面叠加得到的还是平面

超平面

注意，一次函数中变量数量继续增多时，将获得**超平面** (hyperplane)，对应的解析式如下：

$$y = f\left(x_1, x_2, \ldots, x_D\right) = w_1 x_1 + w_2 x_2 + \ldots + w_D x_D + b \tag{10}$$

将 (10) 写成矩阵运算形式：

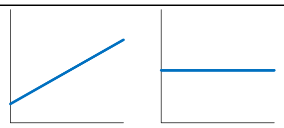
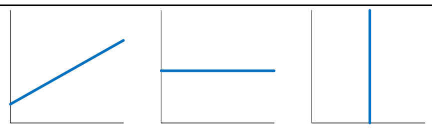
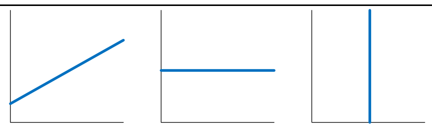
$$y = f\left(\boldsymbol{x}\right) = \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b \tag{11}$$

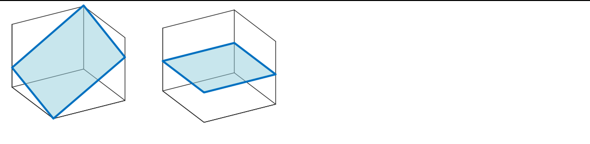
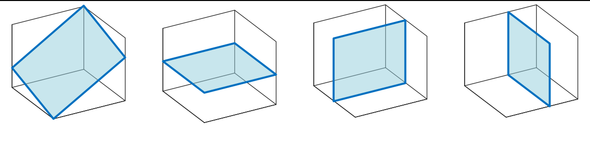
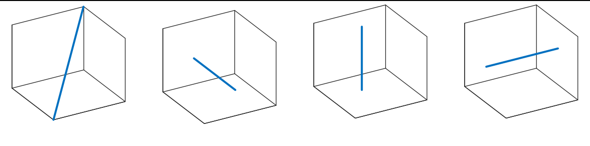
其中，

$$\boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_D \end{bmatrix}, \quad \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_D \end{bmatrix} \tag{12}$$

平面直线、三维空间直线、三维空间平面可以借助不同数学工具进行描述，如表 1 所总结。请读者格外注意区分，代数中函数、方程式、参数方程三个概念之间区别。

表 1. 不同数学工具描绘直线和平面

	类型	图像
$f\left(x_1\right) = w_1 x_1 + b$	函数	
$w_1 x_1 + w_2 x_2 + b = 0$	方程式	
$\begin{cases} x_1 = c_1 + \tau_1 t \\ x_2 = c_2 + \tau_2 t \end{cases}$	参数方程	

$f(x_1, x_2) = w_1x_1 + w_2x_2 + b$	函数	
$w_1x_1 + w_2x_2 + w_3x_3 + b = 0$	方程式	
$\begin{cases} x_1 = c_1 + \tau_1 t \\ x_2 = c_2 + \tau_2 t \\ x_3 = c_3 + \tau_3 t \end{cases}$	参数方程	

本书前文介绍过一元线性回归，回归模型中只含有一个自变量和一个因变量；从图像上来看，一元线性回归模型就是一条直线。

自变量的个数增加到两个，我们便得到二元线性回归。二元线性回归解析式可以写成 $y = b_0 + b_1x_1 + b_2x_2$，这就是我们本节介绍的二元一次函数，对应的图像是一个平面。

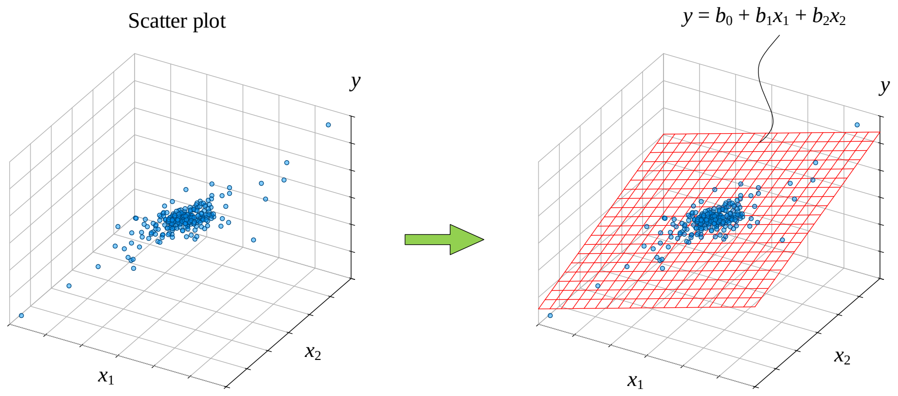


图 6. 从散点图到二元回归平面

图 6 左图是三维直角坐标系散点图。通过观察散点图，我们可以发现因变量随自变量变化的大致趋势。

图 6 右图中红色平面就是二元线性回归模型对应的图形，这个平面试图解释自变量和因变量之间的线性量化关系。

以下代码绘制图 1 到图 4；代码中创建了三个自定义函数，用来可视化。另外，请大家修改如下代码并绘制本章后续图像。

```
# Bk3_Ch13_01

import math
import numpy as np
import matplotlib.pyplot as plt
from matplotlib import cm

def mesh_square(x1_0,x2_0,r,num):

    # generate mesh

    rr = np.linspace(-r,r,num)
    xx1,xx2 = np.meshgrid(rr,rr);

    xx1 = xx1 + x1_0;
    xx2 = xx2 + x2_0;

    return xx1, xx2

def plot_surf(xx1,xx2,ff,caption):

    norm_plt = plt.Normalize(ff.min(), ff.max())
    colors = cm.coolwarm(norm_plt(ff))

    fig = plt.figure()
    ax = fig.gca(projection='3d')
    surf = ax.plot_surface(xx1,xx2,ff,
    facecolors=colors, shade=False)
    surf.set_facecolor((0,0,0,0))
    # z_lim = [ff.min(),ff.max()]
    # ax.plot3D([0,0],[0,0],z_lim,'k')
    plt.show()

    plt.tight_layout()
    ax.set_xlabel('$\it{x_1}$')
    ax.set_ylabel('$\it{x_2}$')
    ax.set_zlabel('$\it{f}$($\it{x_1}$,$\it{x_2}$)')
    ax.set_title(caption)


    ax.xaxis._axinfo["grid"].update({"linewidth":0.25, "linestyle" : ":"})
    ax.yaxis._axinfo["grid"].update({"linewidth":0.25, "linestyle" : ":"})
    ax.zaxis._axinfo["grid"].update({"linewidth":0.25, "linestyle" : ":"})

    plt.rcParams["font.family"] = "Times New Roman"
    plt.rcParams["font.size"] = "10"

def plot_contourf(xx1,xx2,ff,caption):

    fig, ax = plt.subplots()

    cntr2 = ax.contourf(xx1,xx2,ff, levels = 15, cmap="RdBu_r")

    fig.colorbar(cntr2, ax=ax)
    plt.show()

    ax.set_xlabel('$\it{x_1}$')
    ax.set_ylabel('$\it{x_2}$')

    ax.set_title(caption)
    ax.grid(linestyle='--', linewidth=0.25, color=[0.5,0.5,0.5])
```

```
#%% initialization

x1_0  = 0;  # center of the mesh
x2_0  = 0;  # center of the mesh
r     = 2;  # radius of the mesh
num   = 30; # number of mesh grids
xx1,xx2 = mesh_square(x1_0,x2_0,r,num); # generate mesh

#%% Visualizations

plt.close('all')

# f(x1,x2) = -x1

ff = -xx1;
caption = '$\it{f} = -\it{x_1}$';
plot_surf (xx1,xx2,ff,caption)
plot_contourf (xx1,xx2,ff,caption)


#%% f(x1,x2) = x2

ff = xx2;
caption = '$\it{f} = \it{x_2}$';
plot_surf (xx1,xx2,ff,caption)
plot_contourf (xx1,xx2,ff,caption)

#%% f(x1,x2) = x1 + x2

ff = xx1 + xx2;
caption = '$\it{f} = \it{x_1} + \it{x_2}$';
plot_surf (xx1,xx2,ff,caption)
plot_contourf (xx1,xx2,ff,caption)

#%% f(x1,x2) = -x1 + x2

ff = -xx1 + xx2;
caption = '$\it{f} = -\it{x_1} + \it{x_2}$';
plot_surf (xx1,xx2,ff,caption)
plot_contourf (xx1,xx2,ff,caption)
```

13.2 正圆抛物面：等高线为正圆

正圆抛物面 (circular paraboloid) 的是**抛物面** (paraboloid) 的一种特殊形式，它的等高线为正圆。正圆抛物面的最简单的形式为：

$$y = f(x_1, x_2) = a\left(x_1^2 + x_2^2\right) \tag{13}$$

(13) 可以写成如下矩阵运算形式：

$$y = f(x_1, x_2) = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \begin{bmatrix} a & 0 \\ 0 & a \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} = a \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} = a\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x} = a\|\boldsymbol{x}\|^2 \tag{14}$$

向量的模

请大家格外注意，(14) 可以写成 $y = f(x_1, x_2) = a\|\boldsymbol{x}\|^2$ 这种形式。

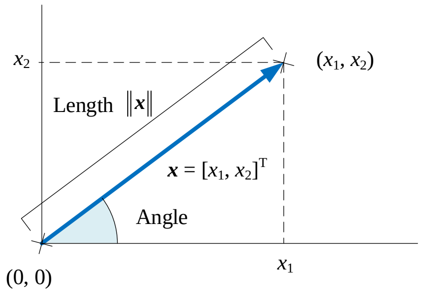


图 7. 向量有大小和方向两个性质

有了坐标系，向量可以理解为平面上有方向的线段，它有大小和方向两个性质。$\|\boldsymbol{x}\|$ 为向量 $\boldsymbol{x}$ 的模 (norm)，就是向量的长度，定义为：

$$\|\boldsymbol{x}\| = \sqrt{x_1^2 + x_2^2} \tag{15}$$

开口朝上

图 8 所示为正圆抛物面开口朝上，对应的解析式如下。

$$y = f(x_1, x_2) = x_1^2 + x_2^2 = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^\mathrm{T}}_{\boldsymbol{x}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\boldsymbol{x}} = \boldsymbol{x}^\mathrm{T}\boldsymbol{x} = \|\boldsymbol{x}\|^2 \tag{16}$$

观察图 8 (b)，曲面等高线为一系列同心正圆。观察等高线变化和曲面，可以发现等高线越密集，曲面变化越剧烈，也就是说曲面坡面越陡峭。图 8 所示曲面最小值点为 (0, 0)。

注意，图 8 (b) 中黄色箭头不再平行；但是，不同位置的黄色箭头都垂直于等高线，并指向函数增大方向。

另外，当 x_1 为定值时，比如 $x_1 = 1$，得到的曲线为抛物线。

$$y = f(x_1, x_2 = 1) = 1 + x_2^2 \tag{17}$$

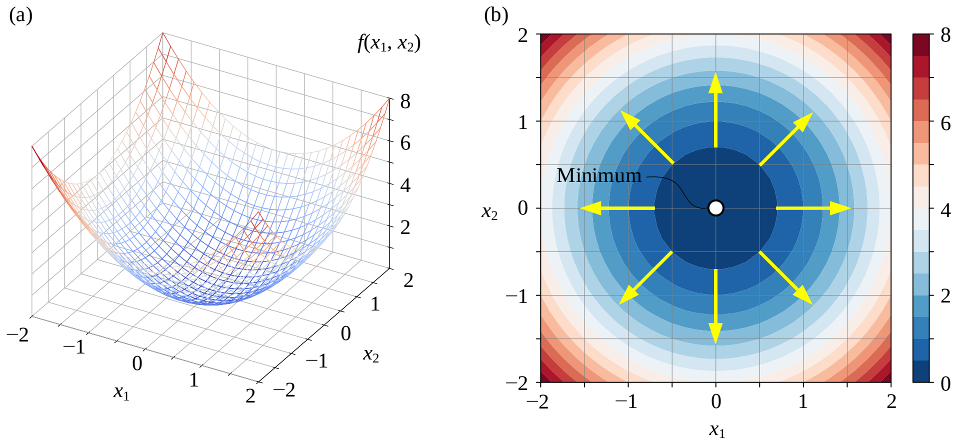


图 8. 开口朝上正圆抛物面，网格图和等高线图

开口朝下

图 9 所示同样为正圆抛物面，但开口朝下；解析式如下所示。

$$y = f(x_1, x_2) = -x_1^2 - x_2^2 = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} = -\boldsymbol{x}^{\mathrm{T}} \boldsymbol{x} \tag{18}$$

图 9 所示曲面在 (0, 0) 处取得最大值点。

图 9 (b) 中不同位置的黄色箭头也都垂直于等高线，并指向函数增大方向。

值得注意的是，图 8 关于 x_1x_2 平面镜像便得到图 9。

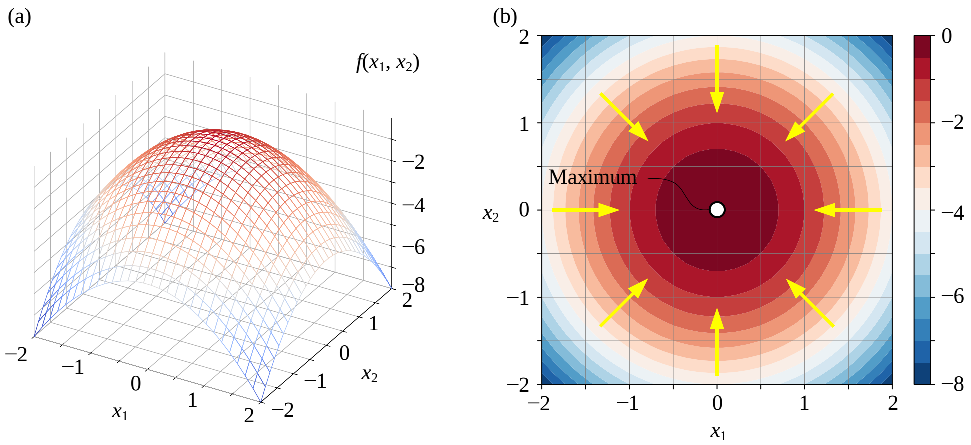


图 9. 开口朝下正圆抛物面，网格图和等高线图

平移

本书前文介绍过函数变换思想，在三维直角坐标系中，将 (14) 中二元函数变量 (x_1, x_2) 平移 (c_1, c_2) 得到：

$$y = f(x_1, x_2) = (x_1 - c_1)^2 + (x_2 - c_2)^2 = (\boldsymbol{x} - \boldsymbol{c})^{\mathrm{T}}(\boldsymbol{x} - \boldsymbol{c}) = \|\boldsymbol{x} - \boldsymbol{c}\|^2 \tag{19}$$

其中，$\boldsymbol{c} = [c_1, c_2]^{\mathrm{T}}$。

举个例子，当 $\boldsymbol{c} = [1, 1]^{\mathrm{T}}$ 时，(19) 对应的抛物面曲面和等高线如图 10 所示。图 9 图像在 x_1x_2 平面平移 $\boldsymbol{c} = [1, 1]^{\mathrm{T}}$，得到图 10 图像。

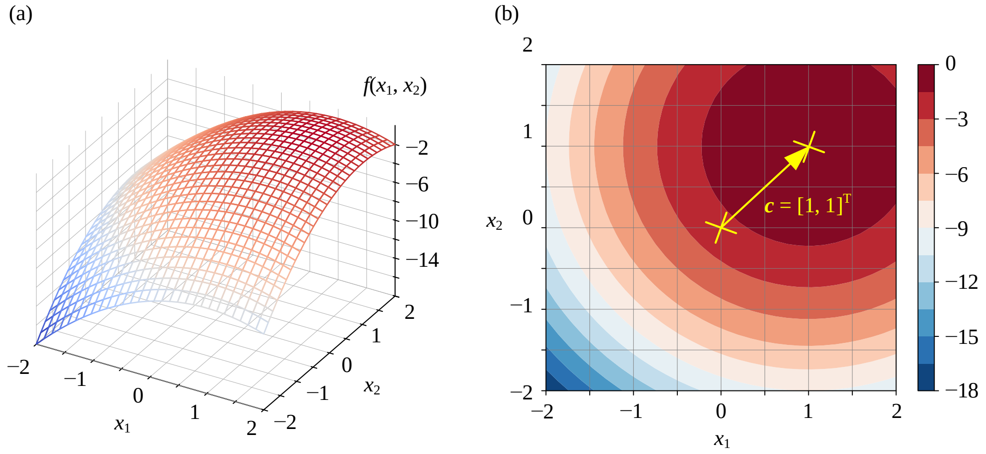


图 10. 抛物面平移

13.3 椭圆抛物面：等高线为椭圆

开口朝上

开口朝上**椭圆抛物面** (elliptic paraboloid) 的一般形式为：

$$y = f(x_1, x_2) = \frac{x_1^2}{a^2} + \frac{x_2^2}{b^2} = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^{\mathrm{T}}}_{\boldsymbol{x}} \begin{bmatrix} 1/a^2 & 0 \\ 0 & 1/b^2 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{\boldsymbol{x}} \tag{20}$$

特别地，当 $a^2 = b^2$ 时，椭圆抛物面便是正圆抛物面。

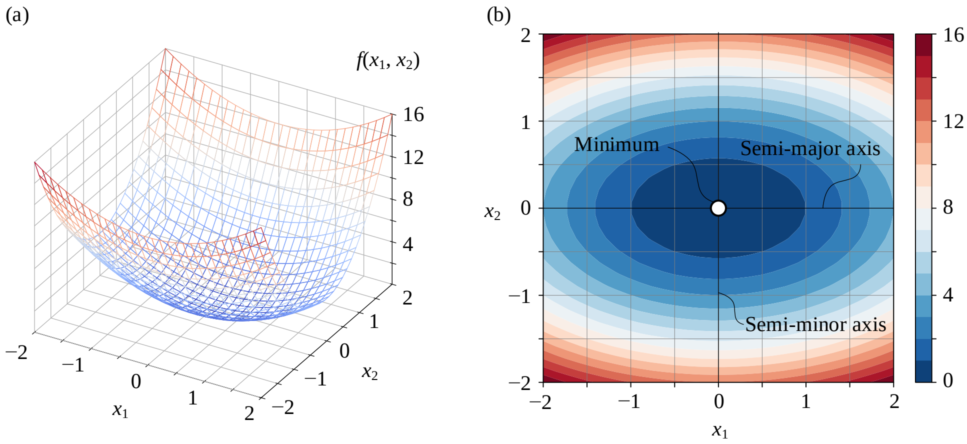


图 11. 开口朝上正椭圆抛物面，等高线为正椭圆，半长轴位于 x_1 轴，网格图和等高线图

图 11 所示椭圆抛物面开口朝上，解析式如下。

$$
y = f(x_1, x_2) = x_1^2 + 3x_2^2 = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^{\mathrm{T}}}_{x} \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{21}
$$

图 11 所示椭圆抛物面的最小值点位于 (0, 0)。图 8 在 x_2 轴方向以一定比例缩放便得到图 11。

如图 11 (b) 所示，曲面等高线为一系列椭圆。这些椭圆为正椭圆，并且半长轴位于 x_1 轴。

回顾一下前文介绍过的椭圆相关概念。**长轴** (major axis) 是过焦点与椭圆相交的线段长，也叫做椭圆最长的直径；**半长轴** (semi-major axis) 是椭圆长轴的一半长。**短轴** (minor axis) 为椭圆最短的直径，**半短轴** (semi-minor axis) 为短轴的一半。

开口朝下

图 12 所示正椭圆抛物面开口朝下，对应解析式如下。

$$
y = f(x_1, x_2) = -3x_1^2 - x_2^2 = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^{\mathrm{T}}}_{x} \begin{bmatrix} -3 & 0 \\ 0 & -1 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{22}
$$

如图 12 (b) 所示，曲面等高线为正椭圆，半长轴位于 x_2 轴。图 12 所示曲面最大值点位于 (0, 0)。

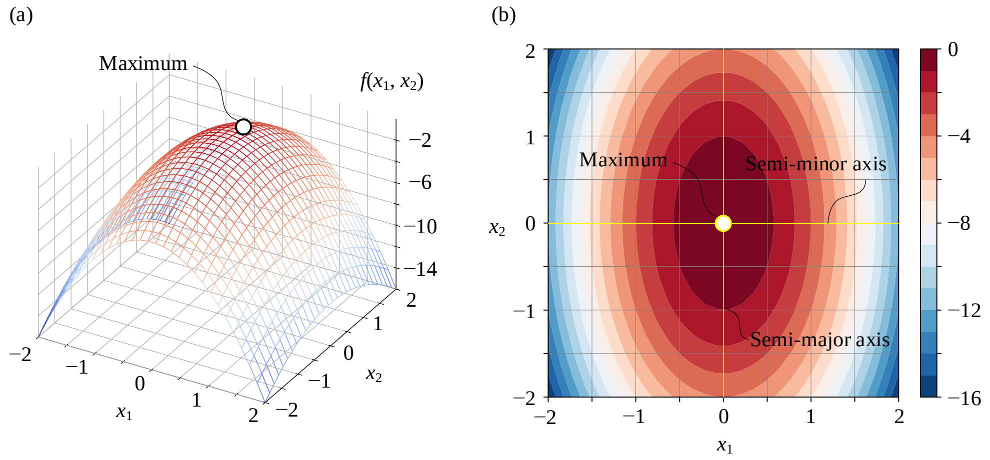


图 12. 开口朝下正椭圆抛物面，半长轴位于 x_2 轴，网格图和等高线图

旋转

图 13 所示旋转椭圆抛物面开口朝上，解析式如下：

$$y = f(x_1, x_2) = x_1^2 + x_1 x_2 + x_2^2 = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \begin{bmatrix} 1 & 1/2 \\ 1/2 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{23}$$

观察图 13 (b) 可以容易发现曲面等高线不再是正椭圆，而是旋转椭圆；旋转椭圆的长半轴和 x_1 轴正方向夹角 135°。

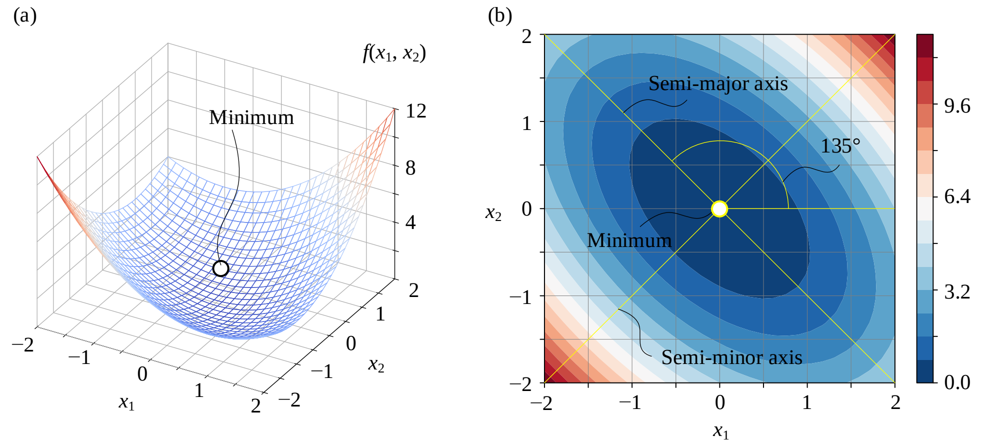


图 13. 开口朝上旋转椭圆抛物面，网格图和等高线图

图 14 所示为旋转椭圆抛物面开口朝下，对应解析式如下：

$$y = f(x_1, x_2) = -x_1^2 + x_1 x_2 - x_2^2 = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \begin{bmatrix} -1 & 1/2 \\ 1/2 & -1 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{24}$$

图 14 曲面等高线椭圆旋转方向和图 13 正好相反。图 14 最大值点也位于 (0, 0)。

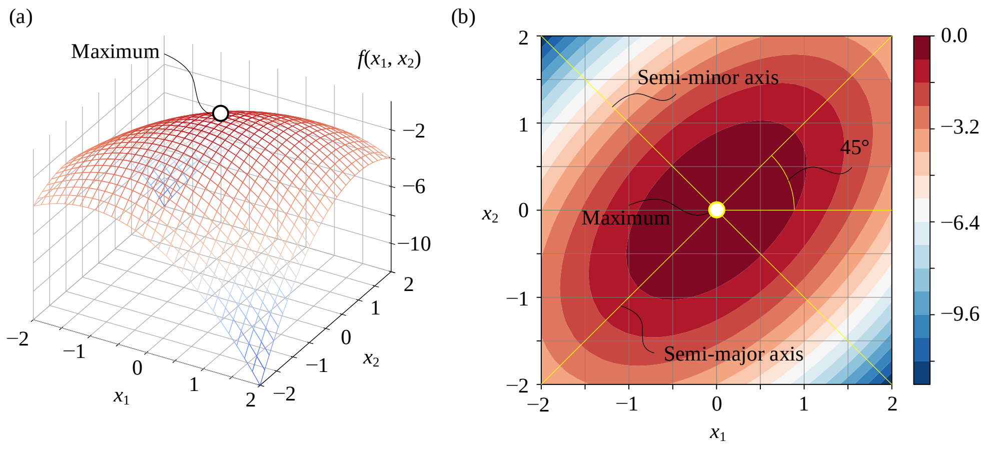


图 14. 开口朝下旋转椭圆抛物面，网格图和等高线图

在多元线性回归中，为了简化模型复杂度，可以引入正则项 (regularizer)。正则项的目的是“收缩”，即让某些估计参数变小，甚至为 0。

L2 正则是常见的正则方法之一。图 15 左上旋转椭圆抛物面上红叉 × 对应的位置就是二元线性回归中最优参数 b_1 和 b_2 (不考虑常数 b_0) 所在位置。

从几何角度，引入 L2 正则项，就相当于在旋转抛物面上叠加一个正圆抛物面。观察图 15 右图，可以发现参数 b_1 和 b_2 位置相对于更靠近原点；这便是 L2 正则项 (正圆曲面) 起到的作用。

L2 正则项权重越大，其影响越大，红叉 × 位置越靠近原点。

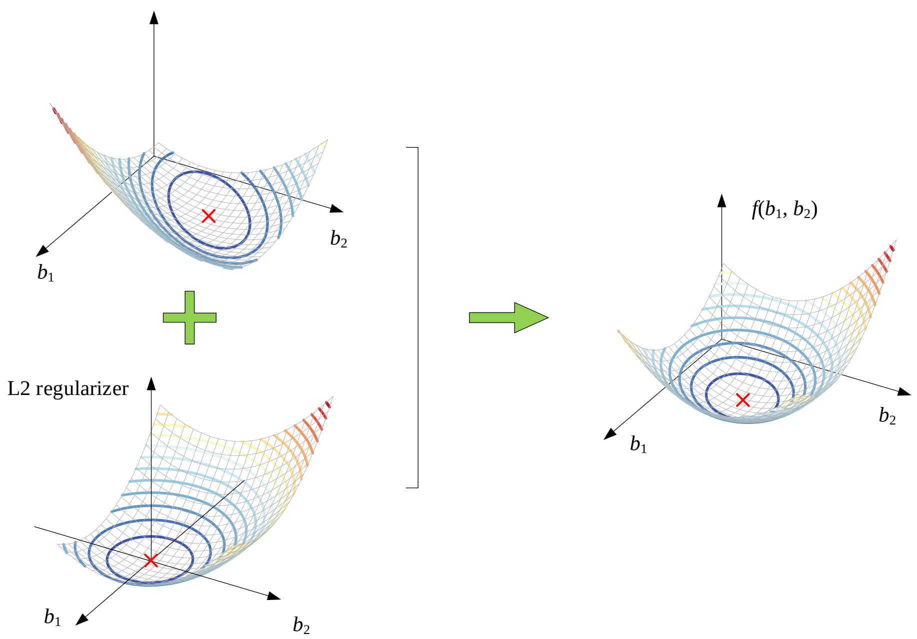


图 15. 线性回归中，L2 正则化相当于椭圆抛物面和正圆抛物面叠加

椭圆相关性质对于数据科学和机器学习很多算法相关重要，本系列丛书后续将继续探讨椭圆和其他数学知识的联系。

13.4 双曲抛物面：马鞍面

双曲抛物面 (hyperbolic paraboloid)，也叫**马鞍面** (saddle surface)，其形状酷似马鞍而得名。双曲抛物面的一般形式为：

$$y = f(x_1, x_2) = \frac{x_1^2}{a^2} - \frac{x_2^2}{b^2} = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \begin{bmatrix} 1/a^2 & 0 \\ 0 & -1/a^2 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{25}$$

图 16 所示为双曲抛物面解析式为。

$$y = f(x_1, x_2) = x_1^2 - x_2^2 = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{26}$$

观察图 16 (b)，可以发现曲面等高线为一系列双曲线。而曲面中心点，也称作**鞍点** (saddle point)，即不是曲面的最大值点也不是最小值点。有关鞍点的性质，本系列丛书会逐步介绍。

正圆和椭圆抛物面的等高线为闭合曲线，而图 16 (b) 中等高线不再闭合。此外请大家自行在图 16 (b) 中四条黑色等高线不同点处，画出前文介绍的黄色箭头 (即梯度)；要求是，箭头垂直该点处等高线，并指向函数增大方向 (朝向暖色系)。

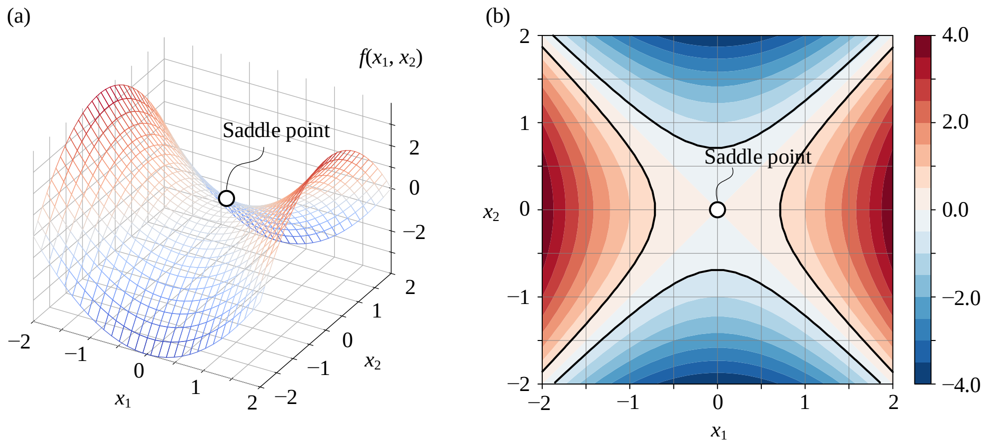


图 16. 双曲抛物面，网格图和等高线图

旋转

图 17 所示为旋转双曲线抛物面，解析式如下：

$$y = f(x_1, x_2) = x_1 x_2 = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \begin{bmatrix} 0 & 1/2 \\ 1/2 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{27}$$

(27) 图 17 (b) 所示等高线实际上是一系列反比例函数曲线。

比较图 16 (b)，可以发现图 17 (b) 中双曲线旋转 45 度。

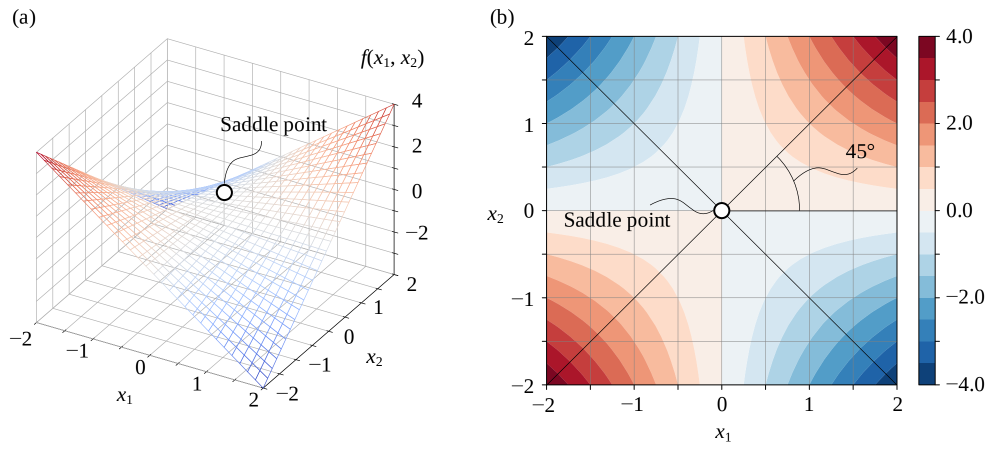


图 17. 旋转双曲抛物面，网格图和等高线图

13.5 山谷和山脊：无数极值点

本节介绍山谷面和山脊面，和它们的几何特征。

山谷面

图 18 所示为**山谷面** (valley surface)，对应一般解析式如下：

$$y = f(x_1, x_2) = x_1^2 = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{28}$$

观察图 18 (b) 可以发现，山谷面存在无数极小值点；并且，这些极小值点均在一条直线上。

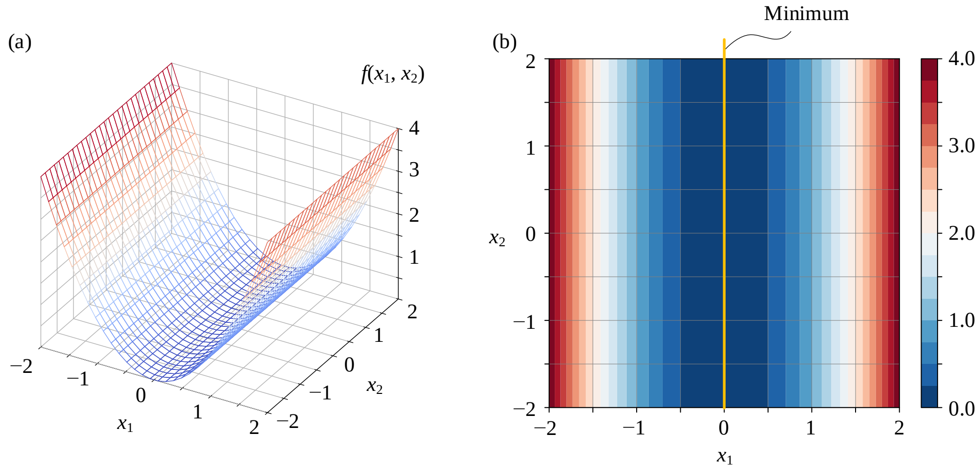


图 18. 山谷面，网格图和等高线图

叠加

如图 19 所示，下正圆抛物面可以看做由两个山谷面叠加得到。

$$y = f(x_1, x_2) = x_1^2 + x_2^2 \tag{29}$$

很多曲面都可以看做是若干不同类型曲面叠加而成；这个几何视角对于理解一些机器学习和数据科学算法非常重要。

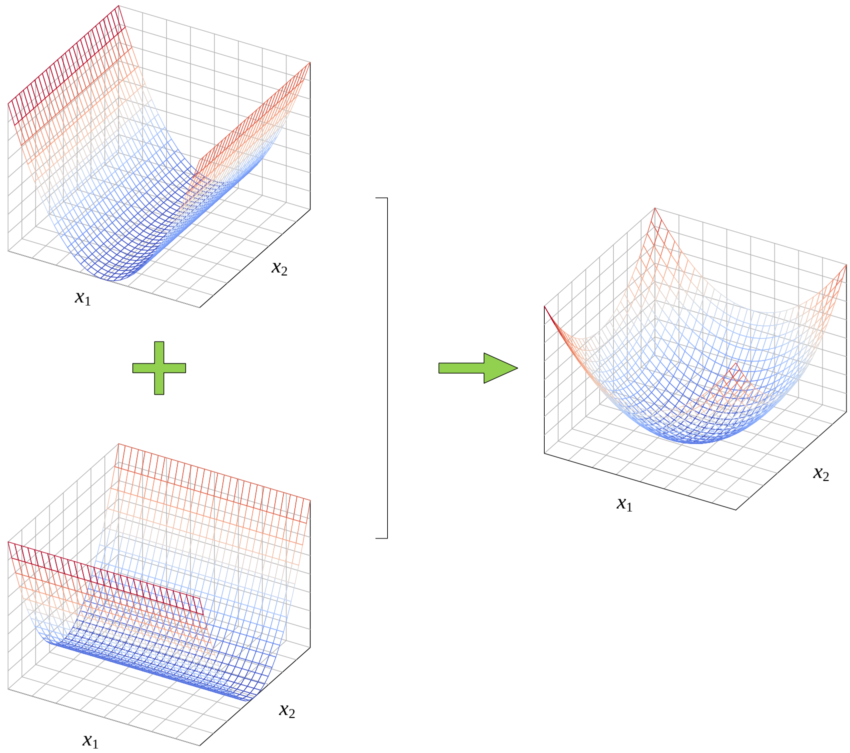


图 19. 两个山谷面合成得到正圆面

山脊面

图 20 所示为旋转**山脊面** (ridge surface)，解析式如下：

$$y = f(x_1, x_2) = -\frac{x_1^2}{2} + x_1 x_2 - \frac{x_2^2}{2} = \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x}^{\mathrm{T}} \begin{bmatrix} -1/2 & 1/2 \\ 1/2 & -1/2 \end{bmatrix} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} \tag{30}$$

图 20 (b) 告诉我们，山脊面有一系列极大值点，它们在同一条斜线上。

也请大家在图 20 (b) 中黑色等高线不同点绘制梯度方向箭头。

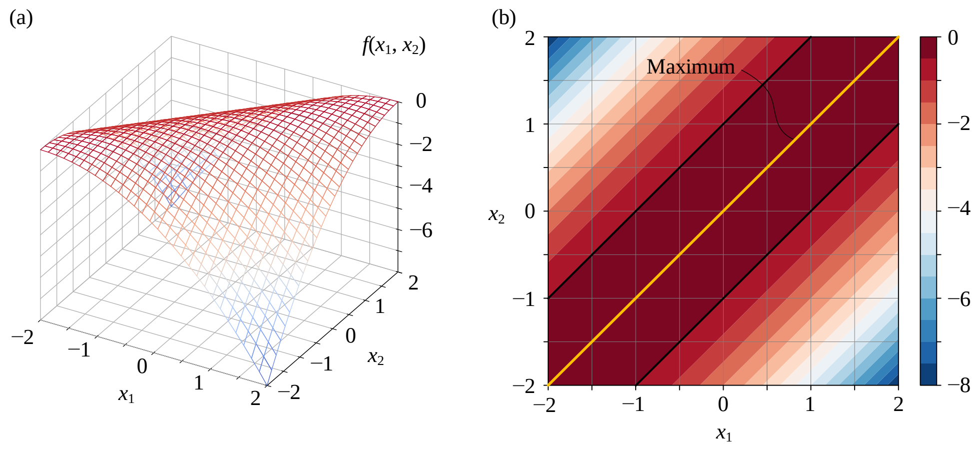


图 20. 旋转山脊面，网格图和等高线图

13.6 锥面：正圆抛物面开方

开口朝上

开口朝上正圆抛物面解析式开平方取正便得到锥面。图 21 所示**锥面** (cone surface) 开口朝上，对应解析式如下：

$$y = f(x_1, x_2) = \sqrt{x_1^2 + x_2^2} = \sqrt{\boldsymbol{x}^{\mathrm{T}} \boldsymbol{x}} = \|\boldsymbol{x}\| \tag{31}$$

观察图 21 (b) 可以发现，锥面的等高线为一系列同心圆。

另外需要注意的是，图 21 (b) 中不同等高线之间均匀渐变，显然不同于图 8 (b)。图 21 所示曲面在 (0, 0) 处取得最小值。但是 (0, 0) 并不光滑，该点为尖点。

前文说过，向量模 $\|\boldsymbol{x}\|$ 代表向量长度，也就相当于距离；图 21 (b) 中不同等高线代表和 (0, 0) 距离相同，前文介绍过这些等高线也可以叫做“等距线”。

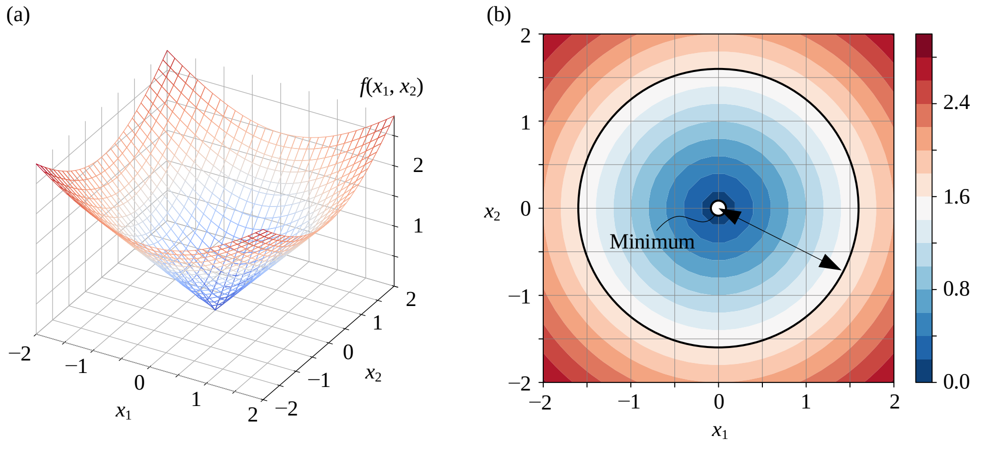


图 21. 正圆锥面，开口朝上，网格图和等高线图

开口朝下

(31) 解析式加个负号便得到如图 22 所示开口向下锥面，解析式如下。

$$y = f(x_1, x_2) = -\sqrt{x_1^2 + x_2^2} \tag{32}$$

图 22 (b) 锥面等高线同样为一系列均匀渐变同心圆，锥面在 (0, 0) 取得最大值。最大值点也是尖点。

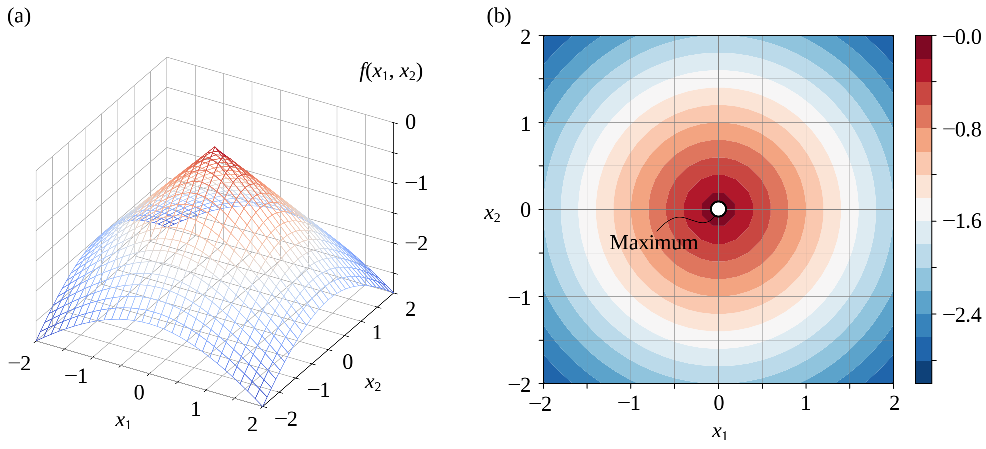


图 22. 正圆锥面，开口朝下，网格图和等高线图

对顶圆锥

将图 21 和图 22 两个圆锥面在顶点处拼接在一起变获得如图 23 所示**对顶圆锥** (double cone 或 vertically opposite circular cone)。大家在前文已经看到看到对顶圆锥和圆锥曲线之间的关系。

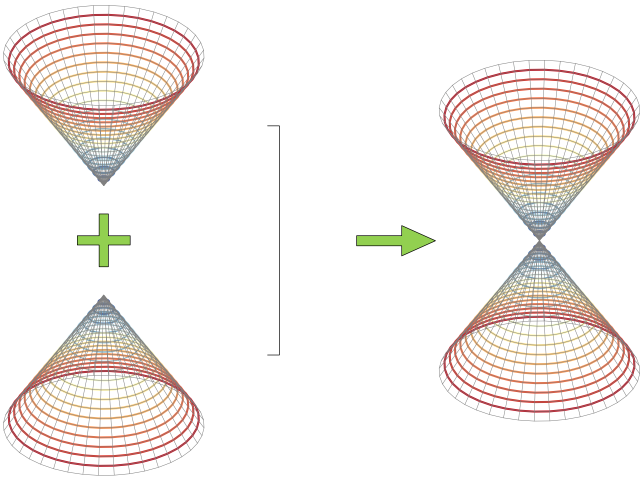


图 23. 对顶圆锥

以下代码绘制图 23 中开口朝上的圆锥面。注意，图 23 中网格面在极坐标系中生成。

```
# Bk3_Ch13_02

import numpy as np
import matplotlib.pyplot as plt

def mesh_circ(c1, c2, r, num):

    theta = np.arange(0,2*np.pi+np.pi/num,np.pi/num)
    r     = np.arange(0,r,r/num)
    theta,r = np.meshgrid(theta,r)
    xx1 = np.cos(theta)*r + c1
    xx2 = np.sin(theta)*r + c2

    return xx1, xx2

xx1, xx2 = mesh_circ(0, 0, 4, 20)

ff_x = np.sqrt(xx1**2 + xx2**2)

# 3D visualization
ax = plt.figure().add_subplot(projection='3d')
ax.plot_wireframe(xx1, xx2, ff_x, rstride=1,
                  cstride=1, color = [0.5,0.5,0.5],
                  linewidth = 0.2)

ax.contour3D(xx1, xx2, ff_x, 20, cmap = 'RdYlBu_r')

ax.xaxis.set_ticks([])
ax.yaxis.set_ticks([])
ax.zaxis.set_ticks([])
plt.xlim(xx1.min(),xx1.max())
plt.ylim(xx2.min(),xx2.max())
ax.set_proj_type('ortho')
# ax.view_init(30, -125)
ax.set_xlabel('$x_1$')
ax.set_ylabel('$x_2$')
ax.set_zlabel('$f(x_1,x_2)$')
plt.tight_layout()
```

13.7 绝对值函数：和超椭圆有关

本节将绝对值函数扩展到二元，本节构造三个不同绝对值函数。

平面对折

第一个例子，$x_1 + x_2$取绝对值，具体解析式如下：

$$y = f(x_1, x_2) = |x_1 + x_2| \tag{33}$$

如图 24 所示，(33) 相当于 $f(x_1, x_2) = x_1 + x_2$ 平面对折。

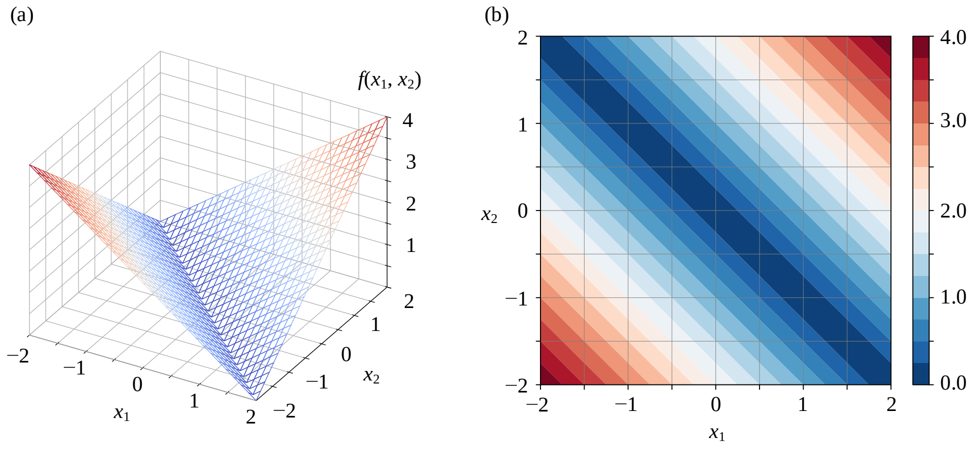


图 24. $f(x_1, x_2) = |x_1 + x_2|$ 空间形状

此外，(33) 相当于旋转山谷面，开平方取正：

$$y = f(x_1, x_2) = \sqrt{(x_1 + x_2)^2} \tag{34}$$

旋转正方形

第二个例子，x_1 和 x_2 分别取绝对值，再求和，解析式如下：

$$y = f(x_1, x_2) = |x_1| + |x_2| \tag{35}$$

图 25 所示 $f(x_1, x_2) = |x_1| + |x_2|$ 等高线图像为一系列旋转正方形。

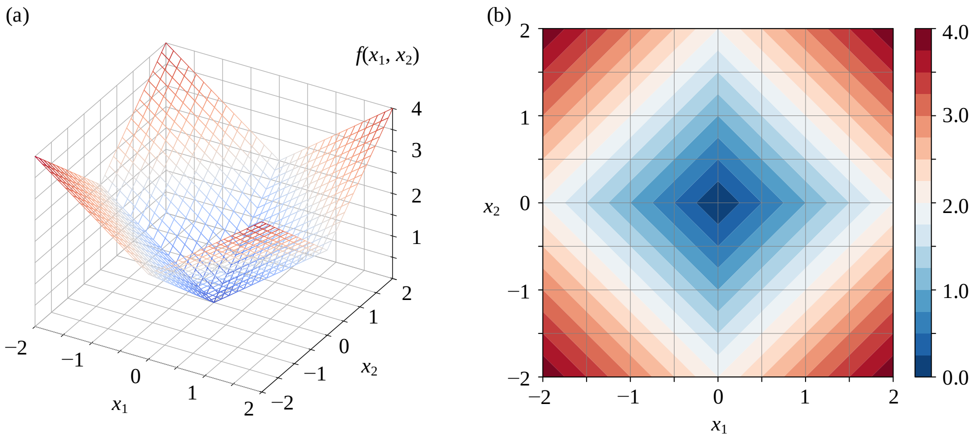


图 25. $f(x_1, x_2) = |x_1| + |x_2|$ 空间形状

正方形

第三个绝对值函数的例子为，x_1 和 x_2 分别取绝对值，比大小后、取两者中最大值，如下：

$$y = f(x_1, x_2) = \max(|x_1|, |x_2|) \tag{36}$$

如图 26 所示，$\max(|x_1|, |x_2|)$ 对应曲面等高线为正方形。

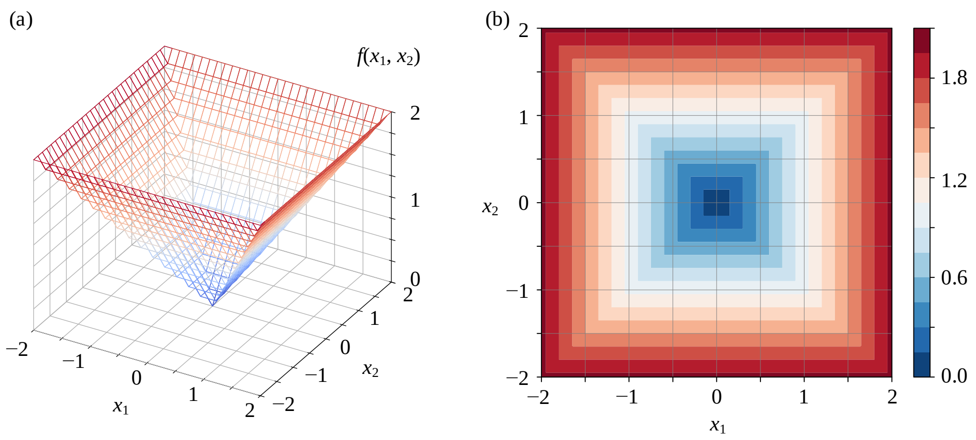


图 26. $f(x_1, x_2) = \max(|x_1|, |x_2|)$ 空间形状

本节介绍的三个绝对值函数和本书前文介绍的超椭圆存在联系；此外，本系列丛书后续还会介绍它们和 L^p 范数、距离度量之间的联系：

实际上，上一节介绍的锥面也可以看做是一种绝对值函数。

$$y = f(x_1, x_2) = \sqrt{|x_1|^2 + |x_2|^2} = \|\boldsymbol{x}\| \tag{37}$$

请大家注意区分绝对值和向量模这两个数学概念。

本章前文介绍过，引入正则项可以简化在多元线性回归。

除了 L2 正则项，L1 正则项也经常使用。

如图 27 所示，引入 L1 正则项，相当于在旋转抛物面上叠加一个 $f(b_1, b_2) = \alpha(|b_1| + |b_2|)$ 绝对函数对应的曲面。

引入 L1 正则项，参数 b_1 和 b_2 位置更靠近原点。特别地，当 L1 正则项权重增大到一定程度，b_1 或 b_2 优化解可能为 0。也就是说，红叉 × 位置可能在横轴或者纵轴上。这种特性是 L2 正则项不具备的。

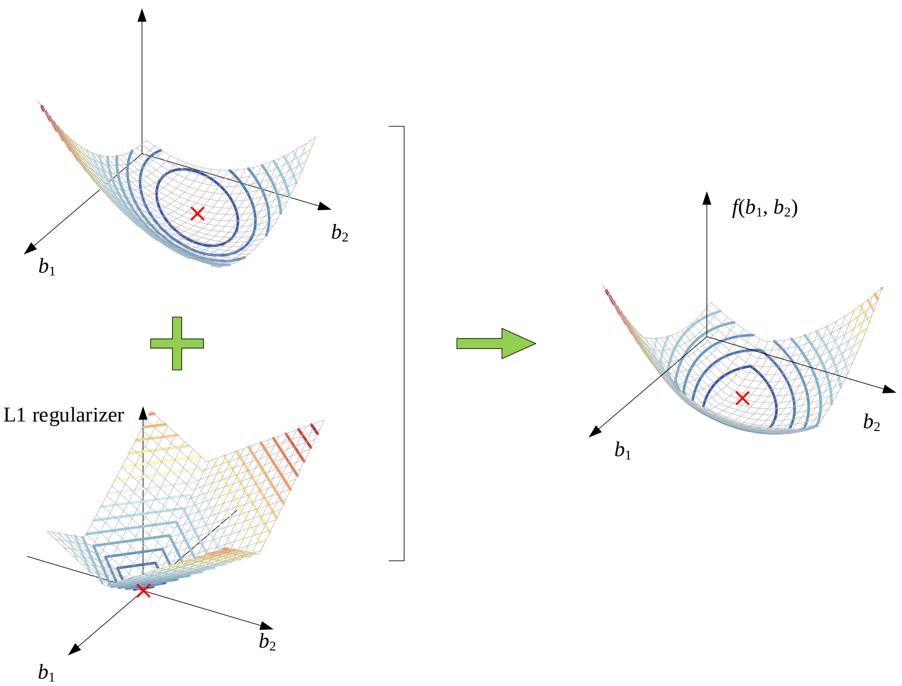


图 27. 线性回归中 L1 正则化相当于椭圆抛物面和绝对值曲面叠加

13.8 逻辑函数：从一元到二元

本节将一元逻辑函数推广到二元；二元逻辑函数对应的一般解析式如下。

$$y = f(x_1, x_2) = \frac{1}{1 + \exp\left(-\left(w_1 x_1 + w_2 x_2 + b\right)\right)} = \frac{1}{1 + \exp\left(-\left(\boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + b\right)\right)} \tag{38}$$

举个例子，当 $w_1 = 1, w_2 = 1, b = 0$ 时。

$$y = f\left(x_1, x_2\right) = \frac{1}{1 + \exp\left(-\left(x_1 + x_2\right)\right)} \tag{39}$$

观察图 28 曲面可以发现，当 $x_1 + x_2$ 趋近正无穷时，(39) 趋近 1，却无法达到 1；当 $x_1 + x_2$ 趋向于负无穷时，(39) 趋近 0，却无法达到 0。

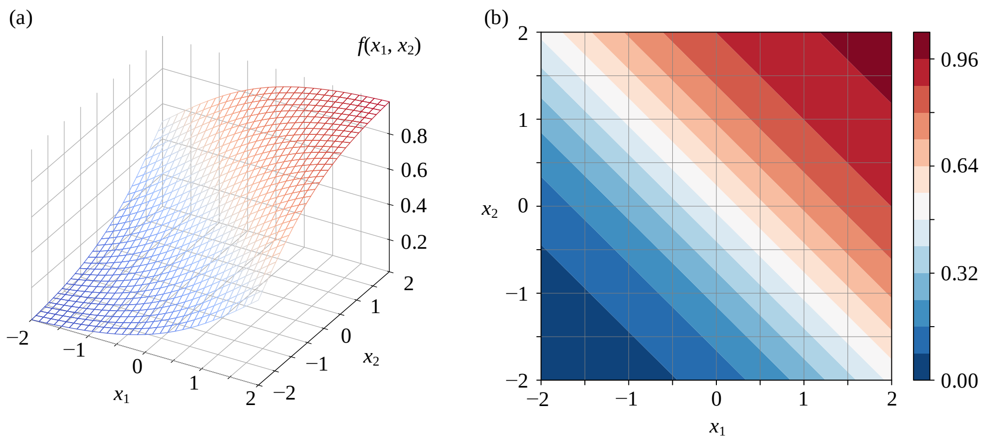


图 28. $f(x_1, x_2) = 1/(1 + \exp(-(x_1+ x_2)))$ 空间形状

再举个例子，当 $w_1 = 4, w_2 = 4, b = 0$ 时

$$y = f\left(x_1, x_2\right) = \frac{1}{1 + \exp\left(-4\left(x_1 + x_2\right)\right)} \tag{40}$$

图 29 所示为 (40) 对应的曲面；对比图 28 和 图 29，不难发现，当 w_1 和 w_2 增大时，坡面变得陡峭。

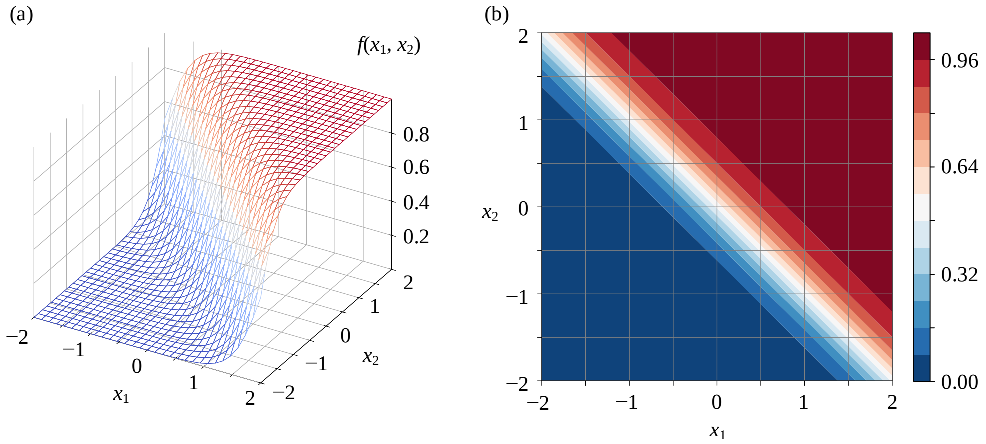


图 29. $f(x_1, x_2) = 1/(1 + \exp(-4(x_1+ x_2)))$ 空间形状

正方形

上一章提到，逻辑函数是 S 型函数的一种；而机器学习中，sigmoid 函数很多时候特指 tanh() 函数。二元 tanh() 函数形式如下：

$$y = f\left(x_1, x_2\right) = \tanh\left(\gamma\left(w_1 x_1 + w_2 x_2\right) + r\right) = \tanh\left(\gamma \boldsymbol{w}^{\mathrm{T}} \boldsymbol{x} + r\right) \tag{41}$$

举个例子，$\gamma = 1$，$w_1 = 0$，$w_2 = 0$，$r = 0$ 时。

$$y = f\left(x_1, x_2\right) = \tanh\left(x_1 + x_2\right) \tag{42}$$

图 30 所示为 (42) 对应的曲面以及平面等高线。当 γ 增大时，曲面也变得陡峭；比如，图 31 对应 $\gamma = 4$，$w_1 = 0$，$w_2 = 0$，$r = 0$ 函数曲面。

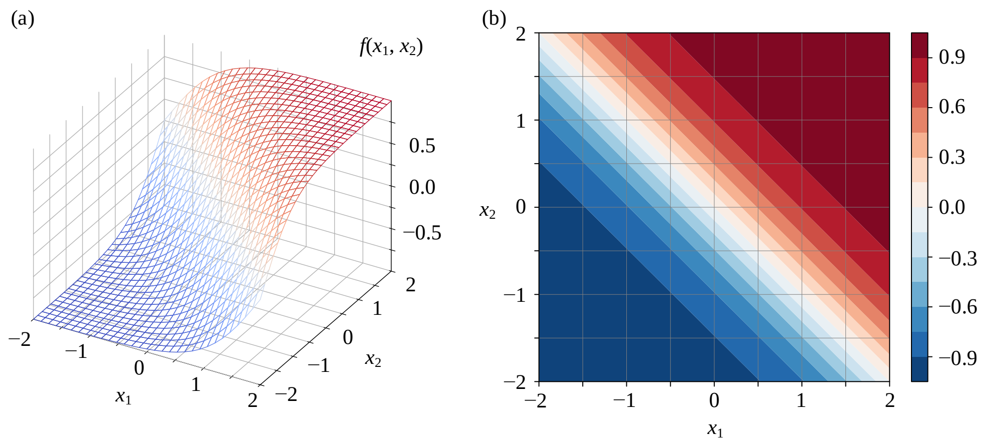


图 30. 二元 Sigmoid 核函数，$\gamma = 1$，$w_1 = 0$，$w_2 = 0$，$r = 0$

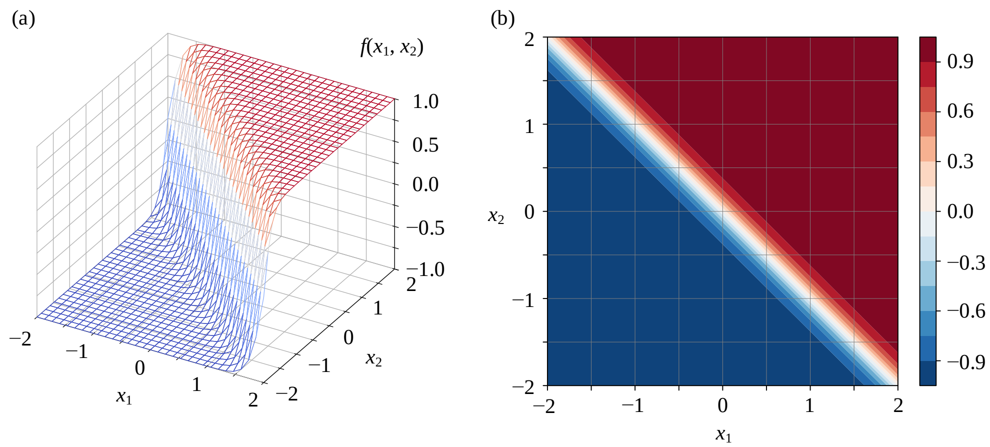


图 31. 二元 Sigmoid 核函数，$\gamma = 4$，$w_1 = 0$，$w_2 = 0$，$r = 0$

13.9 高斯函数：机器学习的多面手

将一元高斯函数推广到二元；二元高斯函数的一般形式为。

$$y = f(x_1, x_2) = \exp\left(-\gamma\left((x_1 - c_1)^2 + (x_2 - c_2)^2\right)\right) \tag{43}$$

举个例子，$\gamma = 1$，$c_1 = 0$，$c_2 = 0$ 时，二元高斯函数函数解析式为。

$$y = f(x_1, x_2) = \exp\left(-\left(x_1^2 + x_2^2\right)\right) = \exp\left(-\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}\right) = \exp\left(-\|\boldsymbol{x}\|^2\right) \tag{44}$$

图 32 所示为 (44) 对应的曲面和平面等高线。

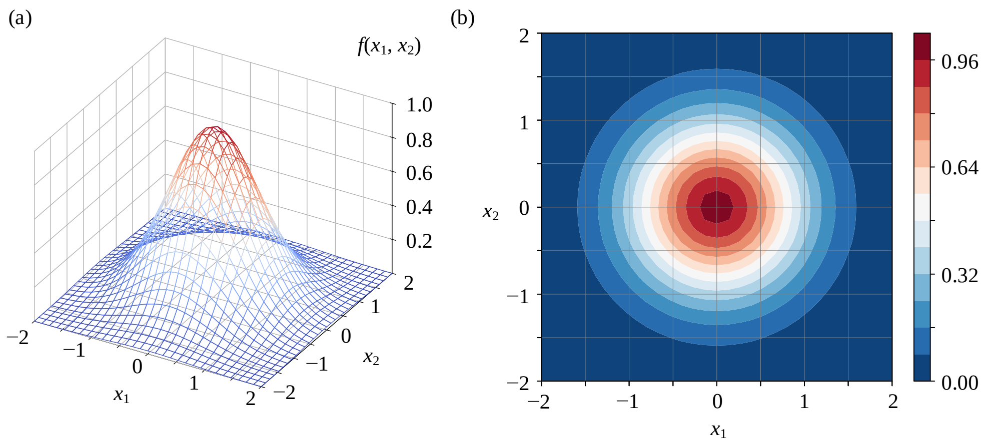


图 32. 高斯核曲面，$\gamma = 1$，$c_1 = 0$，$c_2 = 0$

再举个例子，$\gamma = 2$，$c_1 = 0$，$c_2 = 0$ 时，二元径向基函数为。

$$y = f(x_1, x_2) = \exp\left(-2\left(x_1^2 + x_2^2\right)\right) = \exp\left(-2\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}\right) = \exp\left(-2\|\boldsymbol{x}\|^2\right) \tag{45}$$

图 33 所示为 (45) 对应的曲面和平面等高线。

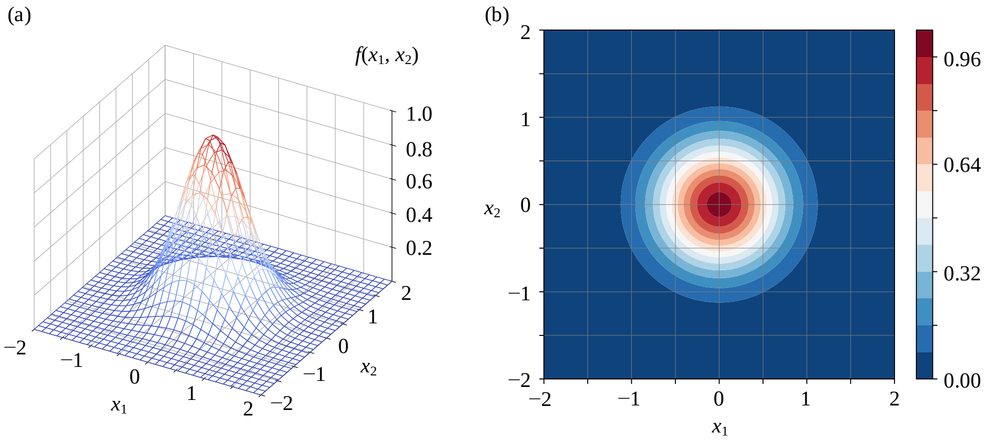


图 33. 高斯核曲面，$\gamma = 2$，$c_1 = 0$，$c_2 = 0$

图 34 所示为 $\gamma = 4$，$c_1 = 0$，$c_2 = 0$ 时，二元径向基函数的曲面和等高线。比较图 32、图 33、图 34，可以发现随着 γ 增大，曲面变得更尖，更陡峭。

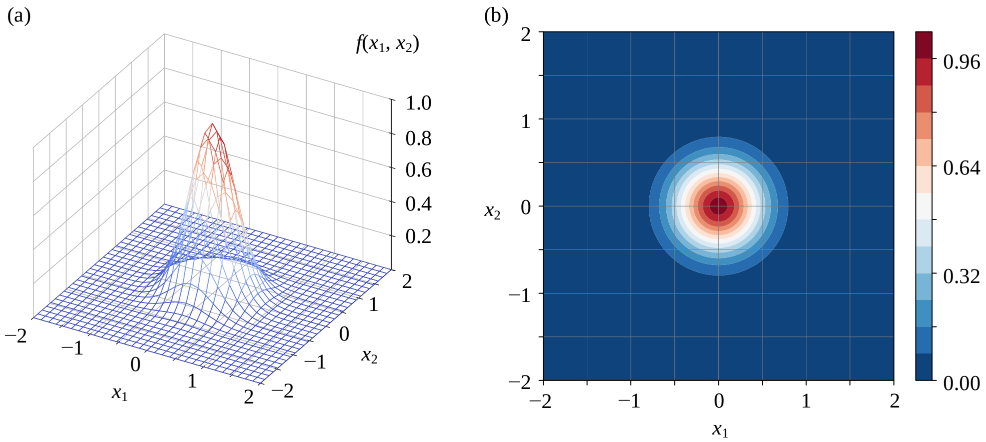


图 34. 高斯核曲面，$\gamma = 4$，$c_1 = 0$，$c_2 = 0$

本书前文简单介绍过一种重要的机器学习方法——支持向量机 SVM (Support Vector Machine)。如图 35 所示，SVM 基本原理是找到一条灰色“宽带”，将绿色点和蓝色点分开，并让灰色“间隔 (margin)”最宽。

灰色“间隔”中心线 (图 35 中红色直线) 便是分割边界，即分类决策边界 (decision boundary)。

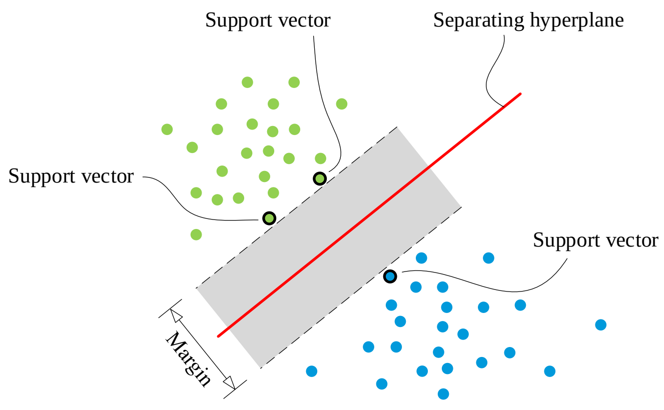


图 35. 支持向量机原理

但是，实际情况却是，很多数据并不能用一条直线将不同标签样本分类，比如图 36 所示情况。

对于这种情况，我们需要采用 SVM 核技巧 (kernel trick)。SVM 核技巧的基本思路就是将数据映射到高维空间中，让数据在这个高维空间中线性可分。

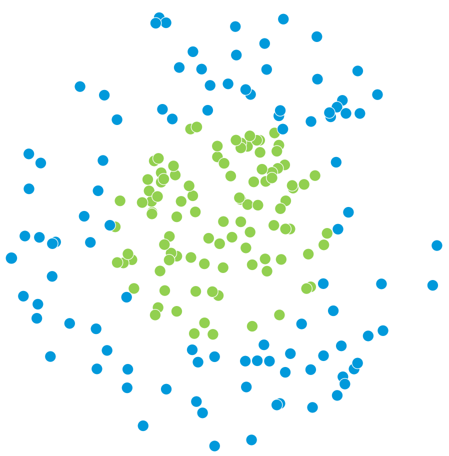


图 36. 线性不可分数据

核技巧如图 37 所示，原数据线性不可分，显然不能用一条直线将数据分成两类。但是，将原来二维数据投射到三维空间之后，就可以用一个平面将数据轻易分类。这个投射规则便是核函数 (kernel function)；而高斯函数是重要的核函数之一。图 37 (b) 右图是由若干高斯函数叠加而成。

红色等高线便是分类决策边界。

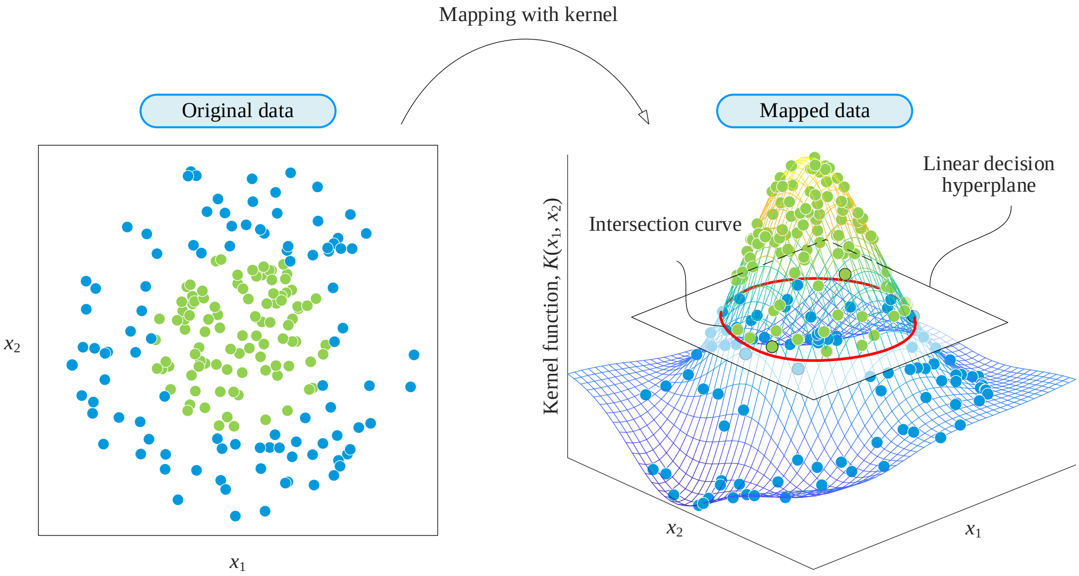
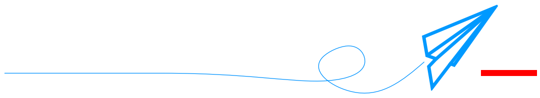


图 37. SVM 核技巧



本节将一元函数推广到二元情况，并将它们和几何、优化、机器学习联系起来；虽然，这样显得“急功近利”，但是必须承认带着“学以致用”目标学习数学将大大提高学习效率。