2024年1月8日 15:01

1.2.1一阶导数

1.导数定义

$$f'(x)=\lim_{\Delta x\to 0}\frac{f(x+\Delta x)-f(x)}{\Delta x}$$

$$f'(x_0)=\lim_{x\to x_0}\frac{f(x)-f(x_0)}{x-x_0}$$

左导数

$$f'_-(x)=\lim_{\Delta x\to 0_-}\frac{f(x+\Delta x)-f(x)}{\Delta x}$$

右导数

$$f'_+(x)=\lim_{\Delta x\to 0_+}\frac{f(x+\Delta x)-f(x)}{\Delta x}$$

2.左导数=右导数→导数可导→函数连续

$f(x)=|x|$ 在x=0处不可导

3.中心差分公式

$$f'(x)\approx\frac{f(x+\Delta x)-f(x-\Delta x)}{2\Delta x}$$

4.重要公式

基本函数	求导公式
幂函数	$(x^a)'=ax^{a-1}$
指数函数	$(e^x)'=e^x$
指数函数	$(a^x)'=a^x\ln a$
三角函数	$(\sin x)'=\cos x$
三角函数	$(\cos x)'=-\sin x$
三角函数	$(\tan x)'=\sec^2 x$
三角函数	$(\cot x)'=-\csc^2 x$
对数函数	$(\ln x)'=\frac{1}{x}$
对数函数	$(\log_a x)'=\frac{1}{\ln a}\frac{1}{x}$
反三角函数	$(\arcsin x)'=\frac{1}{\sqrt{1-x^2}}$
反三角函数	$(\arccos x)'=-\frac{1}{\sqrt{1-x^2}}$
反三角函数	$(\arctan x)'=\frac{1}{1+x^2}$

基本运算	求导公式
加法	$(f(x)+g(x))' = f'(x)+g'(x)$
减法	$(f(x)-g(x))' = f'(x)-g'(x)$
数乘	$(cf(x))' = cf'(x)$
乘法	$(f(x)g(x))' = f'(x)g(x)+f(x)g'(x)$
除法	$\left(\frac{f(x)}{g(x)}\right)' = \frac{f'(x)g(x)-f(x)g'(x)}{g^2(x)}$
倒数	$\left(\frac{1}{f(x)}\right)' = -\frac{f'(x)}{f^2(x)}$

函数与反函数的导数呈倒数关系 $g(y_0)$

$$g'(y) = \frac{1}{f'(g(y))}$$

下面计算反三角函数的导数。如果令 $g(y)=\arcsin(y)$，其反函数为 $f(x)=\sin(x)$。因此有

$$g'(y) = \frac{1}{f'(g(y))} = \frac{1}{\cos(\arcsin(y))} = \frac{1}{\sqrt{1-y^2}}$$

令 $g(y)=\arccos(y)$，其反函数为 $f(x)=\cos(x)$。从而有

$$g'(y) = \frac{1}{f'(g(y))} = \frac{1}{-\sin(\arccos(y))} = -\frac{1}{\sqrt{1-y^2}}$$

令 $g(y)=\arctan(y)$，其反函数为 $f(x)=\tan(x)$。从而有

$$g'(y) = \frac{1}{f'(g(y))} = \frac{1}{\frac{1}{\cos^2(\arctan(y))}} = \cos^2(\arctan(y)) = \frac{1}{1+\tan^2(\arctan(y))} = \frac{1}{1+y^2}$$

5.机器学习常用函数

$$\boldsymbol{Sigmoid(x) = \frac{1}{1+e^x}}$$

在机器学习中广泛使用的 logistic 函数（也称为 sigmoid 函数）定义为

$$f(x) = \frac{1}{1+\mathrm{e}^{-x}} \tag{1.17}$$

它可以看作是如下函数的复合

$$f(u) = u^{-1}, u = 1+\mathrm{e}^v, v=-x$$

根据复合函数与基本函数的求导公式，其导数为

$$f'(x) = -\frac{1}{(1+\mathrm{e}^{-x})^2}(1+\mathrm{e}^{-x})' = -\frac{1}{(1+\mathrm{e}^{-x})^2}(\mathrm{e}^{-x})' = -\frac{1}{(1+\mathrm{e}^{-x})^2}(\mathrm{e}^{-x})(-x)' = \frac{\mathrm{e}^{-x}}{(1+\mathrm{e}^{-x})^2}$$

而

$$\frac{\mathrm{e}^{-x}}{(1+\mathrm{e}^{-x})^2} = \frac{1}{1+\mathrm{e}^{-x}}\frac{\mathrm{e}^{-x}}{1+\mathrm{e}^{-x}} = \frac{1}{1+\mathrm{e}^{-x}}\left(1-\frac{1}{1+\mathrm{e}^{-x}}\right)$$

因此

$$f'(x) = f(x)(1-f(x))$$

1.2.2机器学习中的常用函数

1.ReLU函数

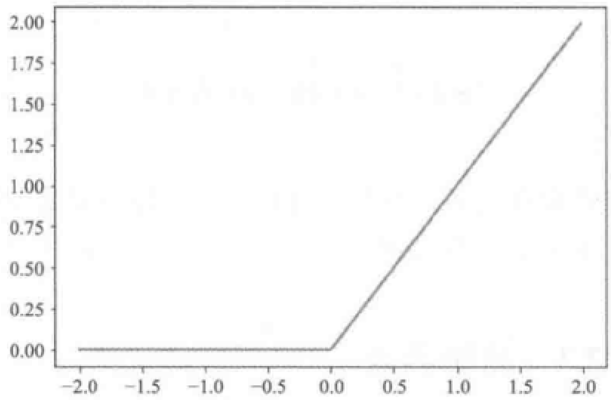

图 1.13 ReLU 函数的曲线

$$f(x)=\begin{cases}x, & x>0\\ 0, & x<0\end{cases}$$

2.softplus函数，是ReLU函数的光滑近似

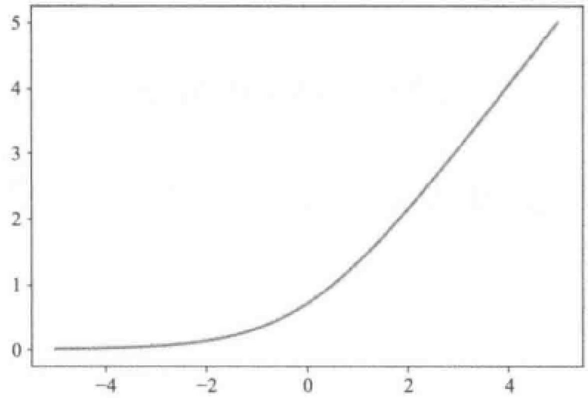

图 1.12 softplus 函数的曲线

$$f(x)=\ln(1+e^{x})$$

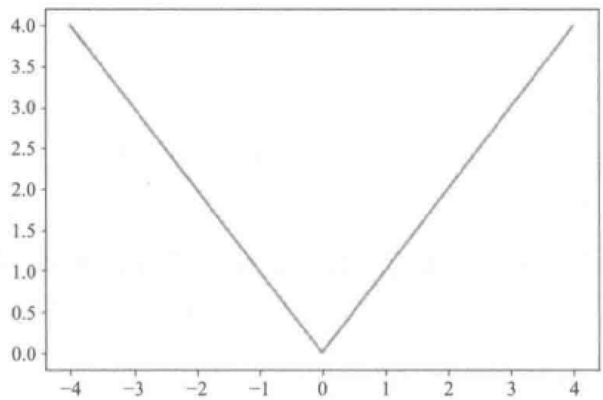

图 1.14 绝对值函数的曲线

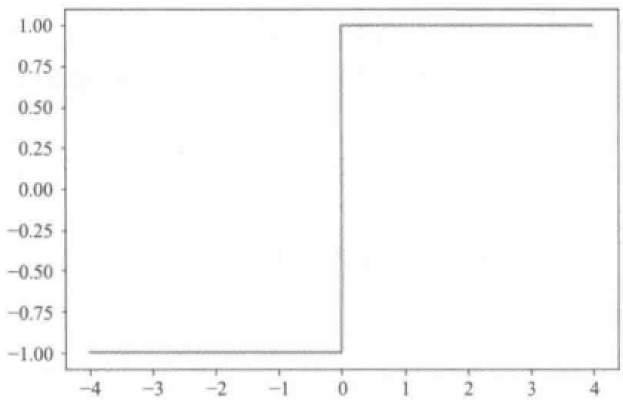

图 1.15 符号函数的曲线

1.2.3高阶导数

在 Python 语言中，符号计算（即计算问题的公式解，也称为解析解）库 sympy 提供了计算各阶导数的功能，由函数 diff 实现。函数的输入值为被求导函数的表达式，要求导的变量，以及导数的阶数（如果不指定，则默认计算一阶导数）；函数的输出值为导数的表达式。下面是示例代码，计算 $\cos(x)$ 的一阶导数。

```
from sympy import *
x = symbols('x')
r = diff(cos(x),x)
print(r)
```

程序运行结果为

```
-sin(x)
```

2. **创建符号变量**:

```python
x = symbols('x')
```

这里，`symbols('x')` 创建了一个名为 `x` 的符号变量。这意味着 `x` 在这个上下文中是一个数学符号，而不是一个具体的数值。

3. **计算导数**:

```python
r = diff(cos(x), x)
```

`diff(cos(x), x)` 计算的是 `cos(x)` 关于 `x` 的导数。在微积分中，余弦函数的导数是负的正弦函数，因此导数是 `-sin(x)`。

1.2.4微分

函数 $y=f(x)$ 在某一区间上有定义，对于区间内的点 x_0，当 x 变为 $x_0+\Delta x$ 时，如果函数的增量 $\Delta y=f(x_0+\Delta x)-f(x_0)$ 可以表示成

$$\Delta y=A\Delta x+o(\Delta x)$$

1.2.5导数与函数的单调性

1.2.6极值判别法则

1.费马定理：极值点的导数值为0（导数值为0的点不一定是极值点）

2.利用二阶导数判断x_0的极大极小值情况

情况一：$f''(x_0)>0$，则 x_0 为函数 $f(x)$ 的严格极小值点。

情况二：$f''(x_0)<0$，则 x_0 为函数 $f(x)$ 的严格极大值点。

情况三：$f''(x_0)=0$，则不定，x_0 可能是极值点也可能不是极值点，需作进一步讨论。

下面对第三种情况进一步细分，假设 $f'(x_0)=\cdots=f^{(n-1)}(x_0)=0$，且 $f^{(n)}(x_0)\neq 0$。可分为两种情况。

情况一：如果 n 是偶数，则 x_0 是极值点。当时 $f^{(n)}(x_0)>0$ 是 $f(x)$ 的严格极小值点，当 $f^{(n)}(x_0)<0$ 时是 $f(x)$ 的严格极大值点。

情况二：如果 n 是奇数，则 x_0 不是 $f(x)$ 的极值点。

该充分条件可以用泰勒公式证明，在 1.4 节给出。

下面举例说明。考虑函数

$$f(x)=x^2$$

其一阶导数为

$$f'(x)=2x$$

令 $f'(x)=0$ 可以解得其驻点为 $x=0$。由于 $f''(0)=2>0$，该点是函数的极小值点。

对于函数

$$f(x)=-x^2$$

其一阶导数为

$$f'(x)=-2x$$

令 $f'(x)=0$ 可以解得其驻点为 $x=0$。由于 $f''(0)=-2<0$，该点是函数的极大值点。

对于函数

$$f(x)=x^3$$

其一阶导数为

$$f'(x)=3x^2$$

令 $f'(x)=0$ 可以解得其驻点为 $x=0$。其二阶导数为 $f''(x)=6x$，三阶导数为 $f^{(3)}(x)=6$。由于 $f''(0)=0$，$f^{(3)}(0)=6$，因此该点不是极值点。这种情况称为鞍点（Saddle Point），会导致数值优化算法如梯度下降法无法找到真正的极值点，在 3.3.3 节和 4.5.1 节会做更详细的介绍。此

1.2.7导数与函数的凹凸性

·凸函数（下凸函数）（二阶导数大于零）

有两点 x、y，如果对于任意的实数 $0\leqslant\theta\leqslant 1$ 都满足如下不等式

$$f(\theta x+(1-\theta)y)\leqslant \theta f(x)+(1-\theta)f(y)$$