# 机器学习 - 周志华 - 线性模型

### 分子布局

如果我们设  $\mathbf{y}=(y_1,y_2,\cdots,y_m)^{\mathrm{T}}$ ,那么对标量 x 求导的符号如下:

$$rac{\partial oldsymbol{y}}{\partial x} = \left(rac{\partial y_1}{\partial x}, \cdots, rac{\partial y_m}{\partial x}
ight)^{\mathrm{T}}$$

这种符号也称为分子布局符号. 也就是对每个位置进行标量求导. 相反的即称为分母布局符号.

反过来就是相当于求梯度:对 $x = (x_1, \dots, x_n)^T$ ,

$$\frac{\partial y}{\partial oldsymbol{x}} = \left( \frac{\partial y}{\partial x_1}, \cdots, \frac{\partial y}{\partial x_n} \right)$$

注意这个求导得到的是列向量,一些经典的标量求导:

② 例: 范数求导

对  $\|\boldsymbol{x}\|^2$  求导.

使用欧氏范数有

$$\|oldsymbol{x}\|^2 = \sum_{k=1}^n x_k^2$$

求导有

$$rac{\partial \|oldsymbol{x}\|^2}{\partial oldsymbol{x}} = (2x_k) = 2oldsymbol{x}^{ ext{T}}$$

②例:内积求导

对  $\langle u, v \rangle$  求导, u, v 均为 x 的函数.

首先 
$$\langle oldsymbol{u}, oldsymbol{v} 
angle = oldsymbol{u}^{ ext{T}} oldsymbol{v} = \sum_{k=1}^n u_k v_k$$
 . 那么我们求导有

机器学习 - 周志华 - 线性模型

$$egin{aligned} rac{\partial oldsymbol{u}^{\mathrm{T}}oldsymbol{v}}{\partial oldsymbol{x}} &= \left(\sum_{k=1}^{n} \left[v_{k}rac{\partial u_{k}}{\partial x_{i}} + u_{k}rac{\partial v_{k}}{\partial x_{i}}
ight]
ight)_{i} \ &= \left(\sum_{k=1}^{n} v_{k}rac{\partial u_{k}}{\partial x_{i}}
ight)_{i} + \left(\sum_{k=1}^{n} u_{k}rac{\partial v_{k}}{\partial x_{i}}
ight)_{i} \ &= oldsymbol{v}^{\mathrm{T}}rac{\partial oldsymbol{u}}{\partial oldsymbol{x}} + oldsymbol{u}^{\mathrm{T}}rac{\partial oldsymbol{v}}{\partial oldsymbol{x}} \end{aligned}$$

#### 向量对向量求导

如果我们记 $\mathbf{x}=(x_1,x_2,\cdots,x_n)^{\mathrm{T}}$ ,  $\mathbf{y}=(y_1,y_2,\cdots,y_m)^{\mathrm{T}}$ , 那么此时有向量对向量求导:

$$egin{aligned} rac{\partial oldsymbol{y}_1}{\partial oldsymbol{x}} = egin{pmatrix} rac{\partial y_1}{\partial x_1} & rac{\partial y_1}{\partial x_2} & \cdots & rac{\partial y_1}{\partial x_n} \ rac{\partial y_2}{\partial x_1} & rac{\partial y_2}{\partial x_2} & \cdots & rac{\partial y_2}{\partial x_n} \ dots & dots & dots & dots \ rac{\partial y_m}{\partial x_1} & rac{\partial y_m}{\partial x_2} & \cdots & rac{\partial y_m}{\partial x_n} \end{pmatrix} \end{aligned}$$

相当于就是说:向量对向量求导将会拉伸一个维度到矩阵的维度,我们看接下来的例子就可以大概明白其原理.

下面的 a 和 A 不是 x 的函数, 均为常量向量、矩阵. 于是有

• 
$$\frac{\partial}{\partial \boldsymbol{x}}(a\boldsymbol{u}) = a\frac{\partial \boldsymbol{u}}{\partial \boldsymbol{x}}$$
.

$$ullet rac{\partial oldsymbol{x}}{\partial oldsymbol{x}} = oldsymbol{I} \ .$$

$$ullet rac{\partial}{\partial m{x}}(m{A}m{x}) = m{A}$$
 .

$$ullet rac{\partial}{\partial oldsymbol{x}}(oldsymbol{x}^{\mathrm{T}}oldsymbol{A}) = oldsymbol{A}^{\mathrm{T}}$$
 .

我们可以拓展上述的内容,矩阵对向量求导将会得到一个三维的**张量**,以此类推有更高维的内容.

# 链式法则和自动求导

向量、矩阵求导也符合链式法则,我们也和正常求导一样拆开即可.

#### ② 例:矩阵求导链式法则

设  $\mathbf{X} \in \mathbb{R}^{m \times n}$  ,  $\mathbf{w} \in \mathbb{R}^n$  ,  $\mathbf{y} \in \mathbb{R}^m$  , 对如下函数求导:

$$z = \|oldsymbol{X}oldsymbol{w} - oldsymbol{y}\|^2$$

我们依次分解:

$$a = Xw, b = a - y, z = ||b||^2$$

求导有:

$$egin{aligned} rac{\partial z}{\partial oldsymbol{w}} &= rac{\partial z}{\partial oldsymbol{b}} rac{\partial oldsymbol{b}}{\partial oldsymbol{a}} rac{\partial oldsymbol{a}}{\partial oldsymbol{w}} rac{\partial oldsymbol{a}}{\partial oldsymbol{b}} rac{\partial oldsymbol{a}}{\partial oldsymbol{a}} rac{\partial oldsymbol{X}oldsymbol{w}}{\partial oldsymbol{w}} \ &= 2oldsymbol{b}^{\mathrm{T}}oldsymbol{I}oldsymbol{X} = 2oldsymbol{b}^{\mathrm{T}}oldsymbol{X} \ &= 2(oldsymbol{X}oldsymbol{w} - oldsymbol{y})^{\mathrm{T}}oldsymbol{X} \end{aligned}$$

#### 回归任务

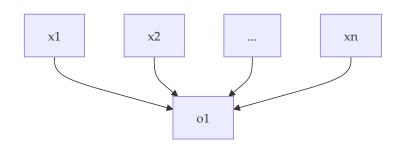
回归任务是通过输入去预测连续数值的任务,线性回归就是通过如下的 n 维输入:

$$oldsymbol{x} = (x_1, x_2, \cdots, x_n)^{ ext{T}}$$

计算出 n 维权重  $\boldsymbol{w}=(w_1,w_2,\cdots,w_n)^{\mathrm{T}}$  和标量偏差 b ,从而可得输出为:

$$y = \langle oldsymbol{w}, oldsymbol{x} 
angle + b$$

线性模型可以看作是一个单层的神经网络:



上面的第一层为输入层,第二层为输出层.

# 衡量回归质量

考虑 MSE:

机器学习 - 周志华 - 线性模型

$$\ell(y,\hat{y})=rac{1}{2}(y-\hat{y})^2$$

线性回归任务可以计算其解析解,我们先将训练损失计算为:

$$\ell(oldsymbol{X},oldsymbol{y},oldsymbol{w},b) = rac{1}{2n}\sum_{i=1}^n(y_i - \langle oldsymbol{x}_i,oldsymbol{w}-b
angle)^2 = rac{1}{2n}\|oldsymbol{y} - oldsymbol{X}oldsymbol{w}-b\|^2$$

那么我们实际上就是要求出如下参数:

$$oldsymbol{w}^*, oldsymbol{b}^* = rg\min_{oldsymbol{w}, b} \ell(oldsymbol{X}, oldsymbol{y}, oldsymbol{w}, b)$$

这实际上是一个优化类问题,其解法也比较简单,首先我们将偏差加入到权重当中,我们将偏差对应的项视为新增一个变量 1 ,权重为 b ,也就是说

$$m{X} 
ightarrow (m{X},1); m{w} 
ightarrow (m{w},b)$$

因此有

$$\ell(oldsymbol{X},oldsymbol{y},w) = rac{1}{2n}\|oldsymbol{y} - oldsymbol{X}oldsymbol{w}\|^2$$

求导后有

$$rac{\partial}{\partial oldsymbol{w}} \ell(oldsymbol{X}, oldsymbol{y}, oldsymbol{w}) = rac{1}{n} (oldsymbol{y} - oldsymbol{X} oldsymbol{w})^{ ext{T}} oldsymbol{X}$$

损失函数是凸函数,上述导数取0即有最值,可得

$$\boldsymbol{w}^* = (\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}\boldsymbol{y}$$

这是唯一解. 注意这里我们默认了  $X^TX$  是可逆的,

### 线性模型的变化

#### 

若函数 g(x) 为单调可微函数,则

$$y = g^{-1}(oldsymbol{w}^{\mathrm{T}}oldsymbol{x} + b)$$

则称该线性模型为广义线性模型.

## 线性模型的分类任务: 对数几率回归

我们讨论的是线性回归模型,但是如果任务是分类的怎么办?考虑二分类任务,其输出标记  $y\in\{0,1\}$  ,而线性回归模型产生的预测值  $z=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}+b$  是实值,于是,我们需将实值转换为 0/1 值.最理想的是"单位阶跃函数"(unit-step function):

$$y = egin{cases} 0, & x < 0 \ 0.5, & x = 0 \ 1, & x > 0 \end{cases}$$

但是这种函数不可微,因此我们需要选取一种类似的可微函数.