☑ 写文章

X

矩阵求导术(上)



关注他

1,680 人赞了该文章

矩阵求导的技术,在统计学、控制论、机器学习等领域有广泛的应用。鉴于我看过的一些资料或言 之不详、或繁乱无绪,本文来做个科普,分作两篇,上篇讲标量对矩阵的求导术,下篇讲矩阵对矩 阵的求导术。本文使用小写字母x表示标量,粗体小写字母 **2** 表示(列)向量,大写字母X表示矩 阵。

首先来琢磨一下定义,标量例矩阵X的导数,定义为 $\frac{\partial f}{\partial X} = \left[\frac{\partial f}{\partial X_{ii}} \right]$,即例X逐元素求导排成

与X尺寸相同的矩阵。然而,这个定义在计算中并不好用,实用上的原因是对函数较复杂的情形难 以逐元素求导;哲理上的原因是逐元素求导破坏了整体性。试想,为何要将f看做矩阵X而不是各元 素 X_{ij} 的函数呢?答案是用矩阵运算更整洁。所以在求导时不宜拆开矩阵,而是要找一个从整体 出发的算法。

为此,我们来回顾,一元微积分中的导数(标量对标量的导数)与微分有联系: df=f'(x)dx;多元微积分中的梯度(标量对向量的导数)也与微分有联系:

$$df = \sum_{i=1}^n rac{\partial f}{\partial x_i} dx_i = rac{\partial f}{\partial oldsymbol{x}}^T doldsymbol{x}$$
,这里第一个等号是全微分公式,第二个等号表达了梯度与微

分的联系:全微分 df 是梯度向量 $\frac{\partial f}{\partial x}$ (n×1)与微分向量 dx (n×1)的内积;受此启发,我们将矩

阵导数与微分建立联系:
$$df = \sum_{i=1}^m \sum_{j=1}^n rac{\partial f}{\partial X_{ij}} dX_{ij} = \mathrm{tr}\left(rac{\partial f}{\partial X}^T dX
ight)$$
。 其中tr代表迹

 $({
m trace})$ 是方阵对角线元素之和,满足性质:对尺寸相同的矩阵 ${\sf A,B}$, ${f tr}({m A}^T{m B}) = \sum_{i,j} {m A_{ij}} {m B_{ij}}$

,即 ${
m tr}({m A}^T{m B})$ 是矩阵A,B的**内积**。与梯度相似,这里第一个等号是全微分公式,第二个等号表达 了矩阵导数与微分的联系:全微分 df 是导数 $\frac{\partial f}{\partial \mathbf{v}}$ (m×n)与微分矩阵 dX (m×n)的内积。

然后来建立运算法则。回想遇到较复杂的一元函数如 $f = \log(2 + \sin x)e^{\sqrt{x}}$,我们是如何求 导的呢?通常不是从定义开始求极限,而是先建立了初等函数求导和四则运算、复合等法则,再来 运用这些法则。故而,我们来创立常用的矩阵微分的运算法则:

- 1. 加减法: $d(X\pm Y)=dX\pm dY$;矩阵乘法: d(XY)=(dX)Y+XdY ;转置: $d(X^T) = (dX)^T$; 迹: $d\operatorname{tr}(X) = \operatorname{tr}(dX)$.
- 2. 逆: $dX^{-1} = -X^{-1}dXX^{-1}$ 。 此式可在 $XX^{-1} = I$ 两侧求微分来证明。
- 3. 行列式: $d|X|=\mathrm{tr}(X^{\#}dX)$,其中 $X^{\#}$ 表示X的伴随矩阵,在X可逆时又可以写作 $d|X| = |X| \mathrm{tr}(X^{-1} dX)$ 。此式可用Laplace展开来证明,详见张贤达《矩阵分析与应用》
- 4. 逐元素乘法: $d(X\odot Y)=dX\odot Y+X\odot dY$, \odot 表示尺寸相同的矩阵X,Y逐元素相
- 5. 逐元素函数: $d\sigma(X) = \sigma'(X) \odot dX$, $\sigma(X) = [\sigma(X_{ij})]$ 是逐元素标量函数运算 , $\sigma'(X) = [\sigma'(X_{ij})]$ 是逐元素求导数。举个例子,



我们试图利用矩阵导数与微分的联系 $df=\mathrm{tr}\left(rac{\partial f}{\partial X}^TdX
ight)$,在求出左侧的微分 df 后 ,该如何写成右侧的形式并得到导数呢?这需要一些迹技巧(trace trick):

1. 标量套上迹:a = tr(a)

2. 转置: $\operatorname{tr}(A^T) = \operatorname{tr}(A)$ 。

3. 线性: $\operatorname{tr}(A \pm B) = \operatorname{tr}(A) \pm \operatorname{tr}(B)$ 。

4. 矩阵乘法交换: $\mathrm{tr}(AB)=\mathrm{tr}(BA)$,其中 A 与 B^T 尺寸相同。 两侧都等于 $\sum_{i,j}A_{ij}B_{ji}$ 。

5. 矩阵乘法/逐元素乘法交换: $\operatorname{tr}(A^T(B\odot C))=\operatorname{tr}((A\odot B)^TC)$,其中 A,B,C 尺寸相同。 两侧都等于 $\sum_{i,j}A_{ij}B_{ij}C_{ij}$ 。

观察一下可以断言,若标量函数f是矩阵X经加减乘法、逆、行列式、逐元素函数等运算构成,则使用相应的运算法则对f求微分,再使用迹技巧给df套上迹并将其它项交换至dX左侧,即能得到导数。

在建立法则的最后,来谈一谈复合:假设已求得 $\frac{\partial f}{\partial Y}$,而Y是X的函数,如何求 $\frac{\partial f}{\partial X}$ 呢?在微积分中有标量求导的链式法则 $\frac{\partial f}{\partial x} = \frac{\partial f}{\partial y} \frac{\partial y}{\partial x}$,但这里我们不能沿用链式法则,因为矩阵对矩阵的导数 $\frac{\partial Y}{\partial X}$ 截至目前仍是未定义的。于是我们继续追本溯源,链式法则是从何而来?源头仍然是微分。我们直接从微分入手建立复合法则:先写出 $df = \operatorname{tr}\left(\frac{\partial f}{\partial Y}^T dY\right)$,再将dY用dX表示出来代入,并使用迹技巧将其他项交换至dX左侧,即可得到 $\frac{\partial f}{\partial X}$ 。

接下来演示一些算例。特别提醒要依据已经建立的运算法则来计算,不能随意套用微积分中标量导数的结论,比如认为AX对X的导数为A,这是没有根据、意义不明的。

例 $1: f = \boldsymbol{a}^T X \boldsymbol{b}$,求 $\frac{\partial f}{\partial X}$ 。其中 \boldsymbol{a} 是 $m \times 1$ 列向量,X是 $m \times n$ 矩阵, \boldsymbol{b} 是 $n \times 1$ 列向量,f是标量。

解:先使用矩阵乘法法则求微分,这里的 $m{a}, m{b}$ 是常量, $m{da} = m{0}, m{db} = m{0}$,得到: $m{df} = m{a}^T m{dX} m{b}$,再套上迹并做矩阵乘法交换: $m{df} = \mathrm{tr}(m{a}^T m{dX} m{b}) = \mathrm{tr}(m{ba}^T m{dX})$,注意这里我们根据 $\mathbf{tr}(m{AB}) = \mathbf{tr}(m{BA})$ 交换了 $m{a}^T m{dX} = m{b}$ 。 对照导数与微分的联系 $m{df}^T$

$$df = \mathrm{tr}\left(rac{\partial f}{\partial X}^T dX
ight)$$
,得到 $rac{\partial f}{\partial X} = (oldsymbol{ba}^T)^T = oldsymbol{ab}^T$.

注意:这里不能用 $\frac{\partial f}{\partial X} = \boldsymbol{a}^T \frac{\partial X}{\partial X} \boldsymbol{b} = ?$,导数与乘常数矩阵的交换是不合法则的运算(而微分是合法的)。有些资料在计算矩阵导数时,会略过求微分这一步,这是逻辑上解释不通的。

例2: $f=m{a}^T\exp(m{X}m{b})$,求 $\frac{\partial f}{\partial m{X}}$ 。其中 $m{a}$ 是 $m{m} imes 1$ 列向量, $m{X}$ 是 $m{m} imes n$ 矩阵, $m{b}$ 是 $m{n} imes 1$ 列向量, \exp 表示逐元素求指数, $m{f}$ 是标量。

解:先使用矩阵乘法、逐元素函数法则求微分: $df=oldsymbol{a}^Tig(\exp(Xoldsymbol{b}ig)\odot(dXoldsymbol{b}ig)$,再套上迹并做

 $df=\operatorname{tr}(a^T(\exp(Xb)\odot(dXb)))=\operatorname{tr}((a\odot\exp(Xb))^TdXb)=\operatorname{tr}(b(a\odot\exp(Xb))^TdX)$,注意这里我们先根据 $\operatorname{tr}(A^T(B\odot C))=\operatorname{tr}((A\odot B)^TC)$ 交换了 $a\cdot\exp(Xb)$ 与 dXb,再根据 $\operatorname{tr}(AB)=\operatorname{tr}(BA)$ 交换了 $(a\odot\exp(Xb))^TdX$ 与。对照导数与微分的

联系
$$df=\mathrm{tr}\left(rac{\partial f}{\partial X}^TdX
ight)$$
,得到

$$\frac{\partial f}{\partial X} = (\boldsymbol{b}(\boldsymbol{a}\odot \exp(X\boldsymbol{b}))^T)^T = (\boldsymbol{a}\odot \exp(X\boldsymbol{b}))\boldsymbol{b}^T$$

例3: $f=\mathrm{tr}(Y^TMY), Y=\sigma(WX)$,求 $\dfrac{\partial f}{\partial X}$ 。其中W是l imes m列向量,X是m imes n矩阵,Y是l imes n矩阵,M是l imes l对称矩阵, σ 是逐元素函数,f是标量。

解:先求 $\frac{\partial f}{\partial Y}$, 求微分,使用矩阵乘法、转置法则:

$$df= ext{tr}((dY)^TMY)+ ext{tr}(Y^TMdY)=2 ext{tr}(Y^TMdY)$$
 ,对照导数与微分的联系,得到 $rac{\partial f}{\partial Y}=2MY$ 。 为求 $rac{\partial f}{\partial X}$,写出 $df= ext{tr}\left(rac{\partial f}{\partial Y}^TdY
ight)$,再将dY用dX表示出来代入,并

使用矩阵乘法/逐元素乘法交换:

$$df=\mathrm{tr}\left(rac{\partial f}{\partial Y}^T(\sigma'(WX)\odot(WdX))
ight)=\mathrm{tr}\left(\left(rac{\partial f}{\partial Y}\odot\sigma'(WX)
ight)^TWdX
ight)$$
 , সৃ

照导数与微分的联系,得到

$$rac{\partial f}{\partial X} = W^T \left(rac{\partial f}{\partial Y} \odot \sigma'(WX)
ight) = W^T((2M\sigma(WX)) \odot \sigma'(WX))\,.$$

例4【线性回归】: $m{l} = \| X m{w} - m{y} \|^2$,求 $m{w}$ 的最小二乘估计,即求 $\dfrac{\partial l}{\partial m{w}}$ 的零点。其中 $m{y}$ 是 $m{m} imes 1$ 列向量, $m{X}$ 是 $m{m} imes n$ 矩阵, $m{w}$ 是 $m{n} imes 1$ 列向量, $m{l}$ 是标量。

解:严格来说这是标量对向量的导数,不过可以把向量看做矩阵的特例。先将向量模平方改写成向量与自身的内积: $m{l}=(m{X}m{w}-m{y})^T(m{X}m{w}-m{y})$,求微分,使用矩阵乘法、转置等法则:

$$dl = (Xdoldsymbol{w})^T(Xoldsymbol{w}-oldsymbol{y}) + (Xoldsymbol{w}-oldsymbol{y})^T(Xdoldsymbol{w}) = 2(Xoldsymbol{w}-oldsymbol{y})^TXdoldsymbol{w}$$
。对照导数与微分的联系 $dl = rac{\partial l}{\partial oldsymbol{w}}^T doldsymbol{w}$,得到 $rac{\partial l}{\partial oldsymbol{w}} = (2(Xoldsymbol{w}-oldsymbol{y})^TX)^T = 2X^T(Xoldsymbol{w}-oldsymbol{y})$ 。

$$\dfrac{\partial l}{\partial oldsymbol{w}}$$
 的零点即 $oldsymbol{w}$ 的最小二乘估计为 $oldsymbol{w} = (X^TX)^{-1}X^Toldsymbol{y}$ 。

例5【方差的最大似然估计】:样本 $m{x}_1,\dots,m{x}_n\sim N(m{\mu},\Sigma)$,求方差 Σ 的最大似然估计。写成数学式是: $m{l}=\log |\Sigma|+rac{1}{n}\sum_{i=1}^n (m{x}_i-ar{m{x}})^T\Sigma^{-1}(m{x}_i-ar{m{x}})$,求 $rac{\partial l}{\partial \Sigma}$ 的零点。其中 $m{x}_i$ 是 $m{m}\times 1$ 列向量, $m{x}=rac{1}{n}\sum_{i=1}^n m{x}_i$ 是样本均值, Σ 是 $m{m}\times m{m}$ 对称正定矩阵, $m{l}$ 是标量。

解:首先求微分,使用矩阵乘法、行列式、逆等运算法则,第一项是

$$d\log |\Sigma| = |\Sigma|^{-1} d|\Sigma| = \operatorname{tr}(\Sigma^{-1} d\Sigma)$$
,第二项是

$$rac{1}{n}\sum_{i=1}^n(oldsymbol{x}_i-ar{oldsymbol{x}})^Td\Sigma^{-1}(oldsymbol{x}_i-ar{oldsymbol{x}})=-rac{1}{n}\sum_{i=1}^n(oldsymbol{x}_i-ar{oldsymbol{x}})$$
 赞同 1.7K $oldsymbol{v}$ 9 176 条评论 $oldsymbol{arphi}$ 分享 $ldot$ 收藏

第二项套上迹做交换:
$$\operatorname{tr}\left(\frac{1}{n}\sum_{i=1}^n(\boldsymbol{x}_i-\bar{\boldsymbol{x}})^T\Sigma^{-1}d\Sigma\Sigma^{-1}(\boldsymbol{x}_i-\bar{\boldsymbol{x}})\right)$$
 $=\frac{1}{n}\sum_{i=1}^n\operatorname{tr}((\boldsymbol{x}_i-\bar{\boldsymbol{x}})^T\Sigma^{-1}d\Sigma\Sigma^{-1}(\boldsymbol{x}_i-\bar{\boldsymbol{x}}))$ $=\frac{1}{n}\sum_{i=1}^n\operatorname{tr}\left(\Sigma^{-1}(\boldsymbol{x}_i-\bar{\boldsymbol{x}})(\boldsymbol{x}_i-\bar{\boldsymbol{x}})^T\Sigma^{-1}d\Sigma\right)=\operatorname{tr}(\Sigma^{-1}S\Sigma^{-1}d\Sigma)$, 其中先交换迹与求和,然后将 $\Sigma^{-1}(\boldsymbol{x}_i-\bar{\boldsymbol{x}})$ 交换到左边,最后再交换迹与求和,并定义 $S=\frac{1}{n}\sum_{i=1}^n(\boldsymbol{x}_i-\bar{\boldsymbol{x}})(\boldsymbol{x}_i-\bar{\boldsymbol{x}})^T$ 为样本方差矩阵。得到 $dl=\operatorname{tr}\left((\Sigma^{-1}-\Sigma^{-1}S\Sigma^{-1})d\Sigma\right)$ 。对照导数与微分的联系,有 $\frac{\partial l}{\partial \Sigma}=(\Sigma^{-1}-\Sigma^{-1}S\Sigma^{-1})^T$,其零点即 Σ 的最大似然估计为 $\Sigma=S$ 。

例6【多元logistic回归】: $m{l} = -m{y}^T \log \operatorname{softmax}(W m{x})$,求 $\frac{\partial l}{\partial W}$ 。其中 $m{y}$ 是除一个元素为1外其它元素为0的 $m{m} imes 1$ 列向量, $m{W}$ 是 $m{m} imes n$ 矩阵, $m{x}$ 是 $m{n} imes 1$ 列向量, $m{l}$ 是标量; $m{softmax}(m{a}) = \frac{\exp(m{a})}{\mathbf{1}^T \exp(m{a})}$,其中 $\exp(m{a})$ 表示逐元素求指数, $m{1}$ 代表全1向量。

解:首先将softmax函数代入并写成

 $m{l} = -m{y}^T \left(\log(\exp(Wm{x})) - \mathbf{1} \log(\mathbf{1}^T \exp(Wm{x})) \right) = -m{y}^T Wm{x} + \log(\mathbf{1}^T \exp(Wm{x}))$,这里要注意逐元素 \log 满足等式 $\log(m{u}/c) = \log(m{u}) - \mathbf{1} \log(c)$,以及 $m{y}$ 满足 $m{y}^T \mathbf{1} = \mathbf{1}$ 。求微分,使用矩阵乘法、逐元素函数等法则:

求微分,使用矩阵乘法、逐元素函数等法则:
$$dl = - \boldsymbol{y}^T dW \boldsymbol{x} + \frac{\boldsymbol{1}^T \left(\exp(W \boldsymbol{x}) \odot (dW \boldsymbol{x}) \right)}{\boldsymbol{1}^T \exp(W \boldsymbol{x})} \text{ . 再套上迹并做交换,注意可化简}$$

$$\boldsymbol{1}^T \left(\exp(W \boldsymbol{x}) \odot (dW \boldsymbol{x}) \right) = \exp(W \boldsymbol{x})^T dW \boldsymbol{x} \text{ , 这是根据等式 } \boldsymbol{1}^T (\boldsymbol{u} \odot \boldsymbol{v}) = \boldsymbol{u}^T \boldsymbol{v} \text{ .}$$
 故 $dl = \operatorname{tr} \left(- \boldsymbol{y}^T dW \boldsymbol{x} + \frac{\exp(W \boldsymbol{x})^T dW \boldsymbol{x}}{\boldsymbol{1}^T \exp(W \boldsymbol{x})} \right) = \operatorname{tr} (\boldsymbol{x} (\operatorname{softmax}(W \boldsymbol{x}) - \boldsymbol{y})^T dW)$ 。 对照导数与微分的联系,得到 $\frac{\partial l}{\partial W} = (\operatorname{softmax}(W \boldsymbol{x}) - \boldsymbol{y}) \boldsymbol{x}^T$ 。

另解:定义 $m{a} = W m{x}$,则 $m{l} = -m{y}^T \log \operatorname{softmax}(m{a})$,先如上求出 $m{\frac{\partial l}{\partial m{a}}} = \operatorname{softmax}(m{a}) - m{y}$,再利用复合法则: $m{d} l = \operatorname{tr}\left(m{\frac{\partial l}{\partial m{a}}}^T m{d} m{a}\right) = \operatorname{tr}\left(m{\frac{\partial l}{\partial m{a}}}^T m{d} W m{x}\right) = \operatorname{tr}\left(m{x} m{\frac{\partial l}{\partial m{a}}}^T m{d} W\right)$,得到 $m{\frac{\partial l}{\partial W}} = m{\frac{\partial l}{\partial m{a}}} m{x}^T$ 。

最后一例留给经典的神经网络。神经网络的求导术是学术史上的重要成果,还有个专门的名字叫做BP算法,我相信如今很多人在初次推导BP算法时也会颇费一番脑筋,事实上使用矩阵求导术来推导并不复杂。为简化起见,我们推导二层神经网络的BP算法。

例7【二层神经网络】: $m{l} = -m{y}^T \log \operatorname{softmax}(W_2\sigma(W_1m{x}))$,求 $\dfrac{\partial l}{\partial W_1}$ 和 $\dfrac{\partial l}{\partial W_2}$ 。 其中 $m{y}$ 是除一个元素为1外其它元素为0的的 $m{m} imes 1$ 列向量, $m{W}_2$ 是 $m{m} imes m{p}$ 矩阵, $m{W}_1$ 是 $m{p} imes m{n}$ 矩阵, $m{x}$ 是 $m{n} imes 1$ 列向量, $m{l}$ 是标量; $m{softmax}(m{a}) = \dfrac{\exp(m{a})}{\mathbf{1}^T \exp(m{a})}$ 同例3, $\sigma(\cdot)$ 是逐元素 sigmoid函数 $\sigma(m{a}) = \dfrac{1}{1 + \exp(-m{a})}$ 。

解:定义 $m{a}_1 = W_1m{x}$, $m{h}_1 = \sigma(m{a}_1)$, $m{a}_2 = W_2m{h}_1$,则 $m{l} = -m{y}^T\log \operatorname{softmax}(m{a}_2)$ 。

在前例中已求出 $\frac{\partial l}{\partial a_2} = \operatorname{softmax}(a_2) - \boldsymbol{y}$ 。使用复合法则,注意此处 \boldsymbol{h}_1, W_2 都是变量:

$$dl=\mathrm{tr}\left(rac{\partial l}{\partial oldsymbol{a_2}}^T doldsymbol{a_2}
ight)=\mathrm{tr}\left(rac{\partial l}{\partial oldsymbol{a_2}}^T dW_2oldsymbol{h_1}
ight)+\mathrm{tr}\left(rac{\partial l}{\partial oldsymbol{a_2}}^T W_2 doldsymbol{h_1}
ight)$$
,使用矩阵乘

法交换的迹技巧从第一项得到 $\dfrac{\partial l}{\partial W_2}=\dfrac{\partial l}{\partial m{a_2}}m{h_1}^T$,从第二项得到 $\dfrac{\partial l}{\partial m{h_1}}=W_2^T\dfrac{\partial l}{\partial m{a_2}}$ 。接下来

求 $\frac{\partial l}{\partial a}$,继续使用复合法则,并利用矩阵乘法和逐元素乘法交换的迹技巧:

$$\operatorname{tr}\left(rac{\partial l}{\partial m{h}_1}^T dm{h}_1
ight) = \operatorname{tr}\left(rac{\partial l}{\partial m{h}_1}^T (\sigma'(m{a}_1)\odot dm{a}_1)
ight) = \operatorname{tr}\left(\left(rac{\partial l}{\partial m{h}_1}\odot \sigma'(m{a}_1)
ight)^T dm{a}_1
ight)$$

,得到
$$\dfrac{\partial l}{\partial m{a}_1} = \dfrac{\partial l}{\partial m{h}_1} \odot \sigma'(m{a}_1)$$
。为求 $\dfrac{\partial l}{\partial W_1}$,再用一次复合法则:

$$\operatorname{tr}\left(rac{\partial l}{\partial oldsymbol{a}_1}^T doldsymbol{a}_1
ight) = \operatorname{tr}\left(rac{\partial l}{\partial oldsymbol{a}_1}^T dW_1oldsymbol{x}
ight) = \operatorname{tr}\left(oldsymbol{x}rac{\partial l}{\partial oldsymbol{a}_1}^T dW_1
ight)$$
,得到 $rac{\partial l}{\partial W_1} = rac{\partial l}{\partial oldsymbol{a}_1}oldsymbol{x}^T$ 。

下篇见zhuanlan.zhihu.com/p/24...。

编辑于 2018-11-09

机器学习 矩阵分析 优化

推荐阅读

矩阵求导浅析(一)

本文主要关注标量函数对矩阵的求 导,并提供一种简明直观易操作的 矩阵求导方法。推荐矩阵求导相关 的专栏文章:矩阵求导术(上)矩 阵求导术(下)机器学习中的矩阵/ 向量求导1.内积 向量\[{\bf...

倚楼

机器学习中的矩阵/向量求导

"矩阵求导"似乎是一个三不管的 区域。虽然原理确实是数学分析中 所讲的多元函数求导,但是总结一 些公式以及复合函数求导的法则还 是必要的,毕竟逐分量地求导太累 而且易出错,例如一旦涉及矩...

Towser



机器学:

章华燕

⇒ 切换为时间排序 176 条评论

写下你的评论...

(:)

精选评论(2)

resurrectcore

1年前

网上有个pdf叫做 The Matrix Cookbook.

▲ 19 💂 查看回复



🌃 Liwei Cai 回复 resurrectcore

1年前

我看过,相信写这篇文章的人也看过。感觉这里的方法比硬背公式简单,而且对于搞机器学习 的这里的够用了

▲ 赞同 1.7K ▼

● 176 条评论

7 分享 ★ 收藏