矩阵求导总结（二）

2020-01-13

本文承接上一篇。

链式法则

当目标函数有层级结构，用链式法则可能会比较方便。如$l=f(Y),Y=g(X)$，我们可以分别求$\frac{\partial l}{\partial Y},\frac{\partial Y}{\partial X}$，再用乘积之类的方式连接起来。但个人并不推荐使用链式法则，原因如下

- 注意到，我们要算$\frac{\partial Y}{\partial X}$，这可能是矩阵对矩阵求导，或者向量对向量求导，这经常会将问题变得更加复杂
- 链式法则公式受求导布局影响，容易记错
- 即使有许多层级结构，也可以不用链式法则完成，我会在例题中给出方法

链式法则介绍

本节我们来介绍各种情况下的链式法则

1、向量对向量求导。比如三个向量存在这样的依赖关系$\mathbf{x}\to\mathbf{y}\to\mathbf{z}$，三个向量长度分别为$a,b,c$有链式法则如下

- 分子布局：$\frac{\partial \mathbf{z}}{\partial \mathbf{x}}=\frac{\partial \mathbf{z}}{\partial \mathbf{y}}\frac{\partial \mathbf{y}}{\partial \mathbf{x}}$，注意到维度关系：$(c\times a):(c\times b)\times(b\times a)$
- 分母布局：$\frac{\partial \mathbf{z}}{\partial \mathbf{x}}=\frac{\partial \mathbf{y}}{\partial \mathbf{x}}\frac{\partial \mathbf{z}}{\partial \mathbf{y}}$，注意到维度关系：$(a\times c):(a\times b)\times(b\times c)$

这两个公式只适用于三个都是向量的情况。可以发现，两种布局方式的公式是不同的，分子布局形式更符合我们对链式法则公式的认知，但兼容性不好，就比如将$\mathbf{z}$退化成标量，此时标量对向量求导一般用的是分母布局，而向量对向量求导则用分子布局，布局方式混用导致混乱不说，链式法则公式也会改变，详情可见下一部分。

2、标量对向量求导

- 分子布局：$\frac{\partial z}{\partial \mathbf{x}}=\left(\frac{\partial \mathbf{y}}{\partial \mathbf{x}}\right)^T\frac{\partial z}{\partial \mathbf{y}}$，注意到维度关系：$(a\times 1):(a\times b)\times(b\times 1)$

- 分母布局：$\frac{\partial z}{\partial \mathbf{x}}=\frac{\partial \mathbf{y}}{\partial \mathbf{x}}\frac{\partial z}{\partial \mathbf{y}}$，注意到维度关系：$(a\times 1):(a\times b)\times(b\times 1)$

可以看到，使用分母布局时，公式比较统一，但顺序不符合我们对链式法则公式的认知，不太好记，大概就是从右往左写，顺序完全反过来。

如果有更多变量，如$\mathbf{y}_1\to\mathbf{y}_2\to\cdots\to\mathbf{y}_n\to z$，则分母布局的链式法则公式如下

$$\frac{\partial z}{\partial \mathbf{y}_1}=\frac{\partial \mathbf{y}_2}{\partial \mathbf{y}_1}\frac{\partial \mathbf{y}_3}{\partial \mathbf{y}_2}\cdots\frac{\partial \mathbf{y}_n}{\partial \mathbf{y}_{n-1}}\frac{\partial z}{\partial \mathbf{y}_n}\tag{1}$$

3、标量对矩阵求导。不太方便写链式法则，因为其中进行了向量化改变了矩阵的结构。

假设依赖关系为$X\to Y\to z$，两个矩阵维度分别为$m\times n, p\times q$，那么导数的维度如下（这里只考虑分母布局）

$$\frac{\partial z}{\partial X}:m\times n\qquad \frac{\partial z}{\partial Y}:p\times q\qquad \frac{\partial Y}{\partial X}:mn\times pq$$

从矩阵维度来看，三者关系不会再是$\frac{\partial z}{\partial X}=\frac{\partial Y}{\partial X}\frac{\partial z}{\partial Y}$，但可能是$\mathrm{vec}(\frac{\partial z}{\partial X})=\frac{\partial Y}{\partial X}\mathrm{vec}(\frac{\partial z}{\partial Y})$，这个式子我没有查到资料证实，不过我试过几个例子都是对的，从下面的例题中可以看出。不过就算它是对的，计算过程也过于繁琐了。

4、总结：我个人并不推荐使用链式法则，如果要用，则只推荐公式(1)这个用法，使用分母布局，只涉及向量；但只用公式(1)则适用范围太小。下面我们来看两个例题，我会在例题中给出我推荐使用的方法。

例题

1、标量对向量求导。已知$l=\mathbf{z}^T\mathbf{z},\quad \mathbf{z}=A\mathbf{x}$，求$\frac{\partial l}{\partial \mathbf{x}}$。

- 使用链式法则。由于

$$\frac{\partial l}{\partial \mathbf{x}}=\frac{\partial \mathbf{z}}{\partial \mathbf{x}}\frac{\partial l}{\partial \mathbf{z}}$$

所以接下来我们需要分别求出$\frac{\partial \mathbf{z}}{\partial \mathbf{x}},\frac{\partial l}{\partial \mathbf{z}}$。

$$\begin{gathered}\mathrm{d}l=\mathrm{tr}[\mathrm{d}(\mathbf{z}^T\mathbf{z})]=\mathrm{tr}[\mathrm{d}\mathbf{z}^T\mathbf{z}+\mathbf{z}^T\mathrm{d}\mathbf{z}]=\mathrm{tr}[2\mathbf{z}^T\mathrm{d}\mathbf{z}]\\ \mathrm{d}\mathbf{z}=\mathrm{d}(A\mathbf{x})=A\mathrm{d}\mathbf{x}\end{gathered}$$

所以

$$\frac{\partial l}{\partial \mathbf{z}}=2\mathbf{z},\quad \frac{\partial \mathbf{z}}{\partial \mathbf{x}}=A^T$$

所以

$$\frac{\partial l}{\partial \mathbf{x}} = \frac{\partial \mathbf{z}}{\partial \mathbf{x}} \frac{\partial l}{\partial \mathbf{z}} = 2A^T\mathbf{z} = 2A^TA\mathbf{x}$$

- 只算微分法（推荐）。首先对l进行微分可得

$$\mathrm{d}l = \mathrm{tr}[\mathrm{d}(\mathbf{z}^T\mathbf{z})] = \mathrm{tr}[\mathrm{d}\mathbf{z}^T\mathbf{z} + \mathbf{z}^T\mathrm{d}\mathbf{z}] = \mathrm{tr}[2\mathbf{z}^T\mathrm{d}\mathbf{z}] \tag{2}$$

这里发现式子中带有$\mathrm{d}\mathbf{z}$，于是我们把它求出来

$$\mathrm{d}\mathbf{z} = \mathrm{d}(A\mathbf{x}) = A\mathrm{d}\mathbf{x}$$

将$\mathrm{d}\mathbf{z}$替换入式(2)可得

$$\mathrm{d}l = \mathrm{tr}[2\mathbf{z}^T\mathrm{d}\mathbf{z}] = \mathrm{tr}[2\mathbf{z}^TA\mathrm{d}\mathbf{x}]$$

于是可以直接写出

$$\frac{\partial l}{\partial \mathbf{x}} = 2A^T\mathbf{z} = 2A^TA\mathbf{x}$$

- **总结**：对比两种方法，要算的东西都差不多，都要对给出的两个式子取微分，差别就在于，第二种方法取完微分是直接带入使用，而不是求出中间步骤的导数。这种方法不需要额外记什么东西，也不会增加计算量。在“综合例题-神经网络”一节中，我们可以看到这种方法在复杂案例中的应用。

2、标量对矩阵求导。已知$l = \mathbf{z}^T\mathbf{z}, \quad \mathbf{z} = X\boldsymbol{\beta}$，求$\frac{\partial l}{\partial X}$。

- 使用链式法则。由于

$$\mathrm{d}l = \mathrm{tr}[\mathrm{d}(\mathbf{z}^T\mathbf{z})] = \mathrm{tr}[\mathrm{d}\mathbf{z}^T\mathbf{z} + \mathbf{z}^T\mathrm{d}\mathbf{z}] = \mathrm{tr}[2\mathbf{z}^T\mathrm{d}\mathbf{z}]$$
$$\mathrm{d}\mathbf{z} = \mathrm{d}(X\boldsymbol{\beta}) = \mathrm{d}X\boldsymbol{\beta}$$

我们可以写出

$$\frac{\partial l}{\partial \mathbf{z}} = 2\mathbf{z} \qquad \frac{\partial \mathbf{z}}{\partial X} = \boldsymbol{\beta} \otimes I_n$$

列出各个矩阵维度如下

$$X : n \times p, \quad \boldsymbol{\beta} : p \times 1, \quad \mathbf{z} : n \times 1$$
$$\frac{\partial l}{\partial \mathbf{z}} : n \times 1, \quad \frac{\partial \mathbf{z}}{\partial X} : np \times n$$

则

$$\frac{\partial l}{\partial X} = \frac{\partial \mathbf{z}}{\partial X}\frac{\partial l}{\partial \mathbf{z}} = 2[\boldsymbol{\beta} \otimes I_n]\mathbf{z} \qquad (\frac{\partial l}{\partial X} : np \times 1)$$

这个结果如果做一个向量化的逆，可以得到

$$\frac{\partial l}{\partial X} = 2\mathbf{z}\boldsymbol{\beta}^T = 2X\boldsymbol{\beta}\boldsymbol{\beta}^T \qquad (\frac{\partial l}{\partial X} : n \times p)$$

注：可以看到这种方法比较麻烦，要对矩阵的结构进行各种调整。这里$\mathbf{z}$是个向量还好一点，如果是个矩阵，两个导数都不能直接相乘，如$z = f(Y), Y = AX + B$。这里多说一句，这个式子中Y和X的特定关系下，有$\frac{\partial z}{\partial X} = A^T \frac{\partial z}{\partial Y}$，这个结果可以用上面的链式法则推导出（但很繁琐），也可以用下面的只算微分方法非常容易地得到；所以掌握下面这种方法，是不需要记这个特定关系的。

- 只算微分法（推荐）。首先对l进行微分可得

$$\mathrm{d}l = \mathrm{tr}[\mathrm{d}(\mathbf{z}^T\mathbf{z})] = \mathrm{tr}[\mathrm{d}\mathbf{z}^T\mathbf{z} + \mathbf{z}^T\mathrm{d}\mathbf{z}] = \mathrm{tr}[2\mathbf{z}^T\mathrm{d}\mathbf{z}] \tag{3}$$

然后计算$\mathrm{d}\mathbf{z}$如下

$$\mathrm{d}\mathbf{z} = \mathrm{d}(X\boldsymbol{\beta}) = \mathrm{d}X\boldsymbol{\beta}$$

将微分结果带入(3)式可得

$$\mathrm{d}l = \mathrm{tr}[2\mathbf{z}^T\mathrm{d}\mathbf{z}] = \mathrm{tr}[2\mathbf{z}^T\mathrm{d}X\boldsymbol{\beta}] = \mathrm{tr}[2\boldsymbol{\beta}\mathbf{z}^T\mathrm{d}X]$$

所以

$$\frac{\partial l}{\partial X} = 2\mathbf{z}\boldsymbol{\beta}^T = 2X\boldsymbol{\beta}\boldsymbol{\beta}^T$$

综合例题

logistic二分类

对数似然函数如下

$$\begin{aligned} l &= \sum_{i=1}^n y_i \log p_i + (1-y_i)\log(1-p_i) \\ &= \sum_{i=1}^n y_i \log \frac{e^{\mathbf{x}_i^T\boldsymbol{\beta}}}{1+e^{\mathbf{x}_i^T\boldsymbol{\beta}}} + (1-y_i)\log\frac{1}{1+e^{\mathbf{x}_i^T\boldsymbol{\beta}}} \\ &= \sum_{i=1}^n y_i\mathbf{x}_i^T\boldsymbol{\beta} - \log(1+\exp(\mathbf{x}_i^T\boldsymbol{\beta})) \\ &= \mathbf{y}^T X\boldsymbol{\beta} - \mathbf{1}^T\log(1+\exp(X\boldsymbol{\beta})) \end{aligned}$$

最后一步整理成了矩阵形式，去掉了前面的求和符号，其实也可以带着求和符号算导数，最后再将导数整理成矩阵形式；整理成矩阵的技巧，是关注目标的维度、各个矩阵向量的维度。微分如下

$$
\begin{aligned}
\mathrm{d}l &= \mathbf{tr}\left[\mathbf{y}^T X \mathrm{d}\boldsymbol{\beta} - \mathbf{1}^T\left(\frac{1}{1+\exp(X\boldsymbol{\beta})} \odot \mathrm{d}\exp(X\boldsymbol{\beta})\right)\right] \\
&= \mathbf{tr}\left[\mathbf{y}^T X \mathrm{d}\boldsymbol{\beta} - \left(\mathbf{1}^T \odot \frac{1}{1+\exp(X\boldsymbol{\beta})}\right)^T \mathrm{d}\exp(X\boldsymbol{\beta})\right] \\
&= \mathbf{tr}\left[\mathbf{y}^T X \mathrm{d}\boldsymbol{\beta} - \left(\frac{1}{1+\exp(X\boldsymbol{\beta})}\right)^T (\exp(X\boldsymbol{\beta}) \odot X\mathrm{d}\boldsymbol{\beta})\right] \\
&= \mathbf{tr}\left[\mathbf{y}^T X \mathrm{d}\boldsymbol{\beta} - \left[\left(\frac{1}{1+\exp(X\boldsymbol{\beta})}\right) \odot \exp(X\boldsymbol{\beta})\right]^T X\mathrm{d}\boldsymbol{\beta}\right] \\
&= \mathbf{tr}\left[\mathbf{y}^T X \mathrm{d}\boldsymbol{\beta} - \sigma(X\boldsymbol{\beta})^T X \mathrm{d}\boldsymbol{\beta}\right] \\
&= \mathbf{tr}\left[(\mathbf{y}^T - \sigma(X\boldsymbol{\beta})^T) X \mathrm{d}\boldsymbol{\beta}\right]
\end{aligned}
$$

因此$\nabla_{\beta} l = X^T(\mathbf{y} - \sigma(X\boldsymbol{\beta}))$。其中$\sigma(x) = \frac{e^x}{1+e^x}$。

求$\nabla_{\beta}^2 l$的过程是向量对向量求导，两端同时取微分

$$
\begin{aligned}
\mathrm{d}\nabla_{\beta} l &= -X^T \mathrm{d}\sigma(X\boldsymbol{\beta}) \\
&= -X^T[\sigma'(X\boldsymbol{\beta}) \odot X\mathrm{d}\boldsymbol{\beta}] \\
&= -X^T \mathrm{diag}[\sigma'(X\boldsymbol{\beta})] X \mathrm{d}\boldsymbol{\beta}
\end{aligned}
$$

因此$\nabla_{\beta}^2 l = -X^T \mathrm{diag}[\sigma'(X\boldsymbol{\beta})] X$。如果保留样本求和符号，可以写成

$$
\nabla_{\beta}^2 l = \frac{\partial^2 l(\boldsymbol{\beta})}{\partial \boldsymbol{\beta} \partial \boldsymbol{\beta}^T} = -\sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^T \sigma(\mathbf{x}_i^T \boldsymbol{\beta})(1 - \sigma(\mathbf{x}_i^T \boldsymbol{\beta}))
$$

softmax多分类

首先定义变量维度维度为

$$
\begin{aligned}
&Y : n \times c, \quad \mathbf{y}_i : c \times 1 \\
&X : n \times d, \quad \mathbf{x}_i : d \times 1 \\
&W : d \times c \\
&\mathbf{1}_c : c \times 1, \quad \mathbf{1}_n : n \times 1
\end{aligned}
$$

对数似然函数如下

$$
\begin{aligned}
l &= \sum_{i=1}^{n} \mathbf{y}_i^T \log \frac{\exp(W^T \mathbf{x}_i)}{\mathbf{1}_c^T \exp(W^T \mathbf{x}_i)} && (\text{注：} \log \frac{\mathbf{v}}{u} = \log(\mathbf{v}) - \mathbf{1}\log(u)) \\
&= \sum_{i=1}^{n} \mathbf{y}_i^T W^T \mathbf{x}_i - \mathbf{y}_i^T \mathbf{1}_c \log(\mathbf{1}_c^T \exp(W^T \mathbf{x}_i)) && (\text{注：} \mathbf{y}_i^T \mathbf{1}_c = 1) \\
&= \sum_{i=1}^{n} \mathbf{y}_i^T W^T \mathbf{x}_i - \log(\mathbf{1}_c^T \exp(W^T \mathbf{x}_i)) \\
&= \mathrm{tr}(XWY^T) - \mathbf{1}_n^T \log[\exp(XW)\mathbf{1}_c]
\end{aligned}
$$

最后一步整理成了矩阵形式，去掉了前面的求和符号，其实也可以带着求和符号算导数，最后再将导数整理成矩阵形式；整理成矩阵的技巧，是关注目标的维度、各个矩阵向量的维度。微分如下

$$
\begin{aligned}
\mathrm{d}l &= \operatorname{tr}(X\mathrm{d}WY^T) - \operatorname{tr}\left(\mathbf{1}_n^T\left[\frac{1}{\exp(XW)\mathbf{1}_c}\odot \mathrm{d}\exp(XW)\mathbf{1}_c\right]\right)\\
&= \operatorname{tr}(Y^TX\mathrm{d}W) - \operatorname{tr}\left(\left[\mathbf{1}_n\odot\frac{1}{\exp(XW)\mathbf{1}_c}\right]^T \mathrm{d}\exp(XW)\mathbf{1}_c\right)\\
&= \operatorname{tr}(Y^TX\mathrm{d}W) - \operatorname{tr}\left(\left[\frac{1}{\exp(XW)\mathbf{1}_c}\right]^T \left[\exp(XW)\odot X\mathrm{d}W\right]\mathbf{1}_c\right)\\
&= \operatorname{tr}(Y^TX\mathrm{d}W) - \operatorname{tr}\left(\left[\frac{1}{\exp(XW)\mathbf{1}_c}\mathbf{1}_c^T\right]^T \left[\exp(XW)\odot X\mathrm{d}W\right]\right)\\
&= \operatorname{tr}(Y^TX\mathrm{d}W) - \operatorname{tr}\left(\left[\frac{1}{\exp(XW)\mathbf{1}_c}\mathbf{1}_c^T\odot\exp(XW)\right]^T X\mathrm{d}W\right)\\
&= \operatorname{tr}(Y^TX\mathrm{d}W) - \operatorname{tr}\left(\operatorname{Softmax}(XW)^TX\mathrm{d}W\right)\\
&= \operatorname{tr}((Y^T - \operatorname{Softmax}(XW)^T)X\mathrm{d}W)
\end{aligned}
$$

因此$\nabla_W l = X^T(Y - \operatorname{Softmax}(XW))$。其中$\operatorname{Softmax}(XW)$是个$n\times c$的矩阵，表示对$XW$的每行都计算

$$
\operatorname{softmax}(\mathbf{x}) = \frac{\exp(\mathbf{x})}{\mathbf{1}^T\exp(\mathbf{x})},\qquad (\mathbf{x}: c\times 1)
$$

如果保留样本求和符号，一阶导可以写成这样

$$
\nabla_W l = \sum_{i=1}^n \mathbf{x}_i(\mathbf{y}_i - \operatorname{softmax}(W^T\mathbf{x}_i))^T
$$

求$\nabla_W^2 l$的过程是向量对向量求导，两端同时取微分

$$
\begin{aligned}
\mathrm{d}\nabla_W l &= -\sum_{i=1}^n \mathbf{x}_i\mathrm{d}\left[\operatorname{softmax}(W^T\mathbf{x}_i))^T\right]\\
&= -\sum_{i=1}^n \mathbf{x}_i\mathrm{d}\left[\frac{\exp(W^T\mathbf{x}_i)}{\mathbf{1}_c^T\exp(W^T\mathbf{x}_i)}\right]^T\\
&= -\sum_{i=1}^n \mathbf{x}_i\mathrm{d}\left[\frac{\exp(W^T\mathbf{x}_i)\odot\mathrm{d}W^T\mathbf{x}_i}{\mathbf{1}_c^T\exp(W^T\mathbf{x}_i)} - \frac{\exp(W^T\mathbf{x}_i)\mathbf{1}_c^T(\exp(W^T\mathbf{x}_i)\odot\mathrm{d}W^T\mathbf{x}_i)}{[\mathbf{1}_c^T\exp(W^T\mathbf{x}_i)]^2}\right]^T\\
&= -\sum_{i=1}^n \mathbf{x}_i\left[\frac{\operatorname{diag}[\exp(W^T\mathbf{x}_i)]\mathrm{d}W^T\mathbf{x}_i}{\mathbf{1}_c^T\exp(W^T\mathbf{x}_i)} - \frac{\exp(W^T\mathbf{x}_i)(\exp(W^T\mathbf{x}_i))^T\mathrm{d}W^T\mathbf{x}_i}{[\mathbf{1}_c^T\exp(W^T\mathbf{x}_i)]^2}\right]^T\\
&= -\sum_{i=1}^n \mathbf{x}_i\mathbf{x}_i^T\mathrm{d}W\left[\operatorname{diag}(\operatorname{softmax}(W^T\mathbf{x}_i)) - \operatorname{softmax}(W^T\mathbf{x}_i)\operatorname{softmax}(W^T\mathbf{x}_i)^T\right]^T\\
&= -\sum_{i=1}^n \mathbf{x}_i\mathbf{x}_i^T\mathrm{d}WD(W^T\mathbf{x}_i)^T
\end{aligned}
$$

其中

$$D(\mathbf{a}) = \mathrm{diag}(\mathrm{softmax}(\mathbf{a})) - \mathrm{softmax}(\mathbf{a})\mathrm{softmax}(\mathbf{a})^T$$

接下来进行向量化可得

$$\mathrm{vec}(\mathrm{d}\nabla_W l) = -\sum_{i=1}^{n}(D(W^T\mathbf{x}_i) \otimes \mathbf{x}_i\mathbf{x}_i^T)\mathrm{vec}(\mathrm{d}W)$$

因此$\nabla_W^2 l = -\sum_{i=1}^{n} D(W^T\mathbf{x}_i)^T \otimes \mathbf{x}_i\mathbf{x}_i^T$。

神经网络

首先定义变量维度维度为

$$\begin{aligned} Y: n \times c, &\quad \mathbf{y}_i: c \times 1 \\ X: n \times p, &\quad \mathbf{x}_i: p \times 1 \\ W_1: p \times d, &\quad \mathbf{b}_1: d \times 1 \\ W_2: d \times c, &\quad \mathbf{b}_2: c \times 1 \\ \mathbf{1}_c: c \times 1, &\quad \mathbf{1}_n: n \times 1 \end{aligned}$$

对数似然函数如下

$$l = \sum_{i=1}^{n} \mathbf{y}_i^T \log \mathrm{softmax}(W_2^T \sigma(W_1^T \mathbf{x}_i + \mathbf{b}_1) + \mathbf{b}_2)$$

其中softmax函数定义如下

$$\mathrm{softmax}(\mathbf{x}) = \frac{\exp(\mathbf{x})}{\mathbf{1}^T \exp(\mathbf{x})}, \qquad (\mathbf{x}: c \times 1)$$

我们可以将似然函数拆解成多个式子

$$\begin{aligned} l &= \sum_{i=1}^{n} \mathbf{y}_i^T \log \mathrm{softmax}(\mathbf{a}_{2i}) \\ \mathbf{a}_{2i} &= W_2^T \mathbf{h}_{1i} + \mathbf{b}_2 \\ \mathbf{h}_{1i} &= \sigma(\mathbf{a}_{1i}) \\ \mathbf{a}_{1i} &= W_1^T \mathbf{x}_i + \mathbf{b}_1 \end{aligned}$$

下面我们要将样本的求和符号去掉，推导过程和上一节softmax多分类差不多，这里就不重复推导了，直接给出结果

$$
\begin{aligned}
l &= \operatorname{tr}(A_2 Y^T) - \mathbf{1}_n^T \log[\exp(A_2)\mathbf{1}_c] \\
A_2 &= H_1 W_2 + \mathbf{1}_n \mathbf{b}_2^T \\
H_1 &= \sigma(A_1) \\
A_1 &= X W_1 + \mathbf{1}_n \mathbf{b}_1^T
\end{aligned}
$$

同时也可以得到

$$
\mathrm{d}l = \operatorname{tr}\left(\left[\frac{\partial l}{\partial A_2}\right]^T \mathrm{d}A_2\right) \qquad (\text{其中}\frac{\partial l}{\partial A_2} = Y - \operatorname{Softmax}(A_2)) \tag{4}
$$

对A_2求微分如下

$$
\mathrm{d}A_2 = \mathrm{d}H_1 W_2 + H_1 \mathrm{d}W_2 + \mathbf{1}_n \mathrm{d}\mathbf{b}_2^T
$$

带入(4)式可得

$$
\begin{aligned}
\mathrm{d}l &= \operatorname{tr}\left(\left[\frac{\partial l}{\partial A_2}\right]^T \left[\mathrm{d}H_1 W_2 + H_1 \mathrm{d}W_2 + \mathbf{1}_n \mathrm{d}\mathbf{b}_2^T\right]\right) \\
&= \operatorname{tr}\left(W_2 \left[\frac{\partial l}{\partial A_2}\right]^T \mathrm{d}H_1 + \left[\frac{\partial l}{\partial A_2}\right]^T H_1 \mathrm{d}W_2 + \mathbf{1}_n^T \left[\frac{\partial l}{\partial A_2}\right] \mathrm{d}\mathbf{b}_2\right) \\
&= \operatorname{tr}\left(\left[\frac{\partial l}{\partial H_1}\right]^T \mathrm{d}H_1 + \left[\frac{\partial l}{\partial W_2}\right]^T \mathrm{d}W_2 + \left[\frac{\partial l}{\partial \mathbf{b}_2}\right]^T \mathrm{d}\mathbf{b}_2\right)
\end{aligned}
$$

其中

$$
\frac{\partial l}{\partial H_1} = \frac{\partial l}{\partial A_2} W_2^T, \quad \frac{\partial l}{\partial W_2} = H_1^T \frac{\partial l}{\partial A_2}, \quad \frac{\partial l}{\partial \mathbf{b}_2} = \left[\frac{\partial l}{\partial A_2}\right]^T \mathbf{1}_n
$$

接下来对H_1求微分

$$
\mathrm{d}H_1 = \sigma(A_1) \odot \mathrm{d}A_1
$$

则l微分的第一部分可以表示成

$$
\begin{aligned}
\mathrm{d}l_1 &= \operatorname{tr}\left(\left[\frac{\partial l}{\partial H_1}\right]^T [\sigma'(A_1) \odot \mathrm{d}A_1]\right) \\
&= \operatorname{tr}\left(\left[\frac{\partial l}{\partial H_1} \odot \sigma'(A_1)\right]^T \mathrm{d}A_1\right) \\
&= \operatorname{tr}\left(\left[\frac{\partial l}{\partial A_1}\right]^T \mathrm{d}A_1\right)
\end{aligned} \tag{5}
$$

其中$\frac{\partial l}{\partial A_1} = \frac{\partial l}{\partial H_1} \odot \sigma'(A_1)$。下面计算$A_1$的微分

$$
\mathrm{d}A_1 = X \mathrm{d}W_1 + \mathbf{1}_n \mathrm{d}\mathbf{b}_1^T
$$

带入(5)式可得

$$\begin{aligned}
\mathrm{d}l_1 &= \operatorname{tr}\left(\left[\frac{\partial l}{\partial A_1}\right]^T \left[X\mathrm{d}W_1 + \mathbf{1}_n \mathrm{d}\mathbf{b}_1^T\right]\right) \\
&= \operatorname{tr}\left(\left[\frac{\partial l}{\partial A_1}\right]^T X\mathrm{d}W_1 + \mathbf{1}_n^T\left[\frac{\partial l}{\partial A_1}\right]\mathrm{d}\mathbf{b}_1\right) \\
&= \operatorname{tr}\left(\left[\frac{\partial l}{\partial W_1}\right]^T \mathrm{d}W_1 + \left[\frac{\partial l}{\partial \mathbf{b}_1}\right]^T \mathrm{d}\mathbf{b}_1\right)
\end{aligned}$$

其中

$$\frac{\partial l}{\partial W_1} = X^T\frac{\partial l}{\partial A_1}, \qquad \frac{\partial l}{\partial \mathbf{b}_1} = \left[\frac{\partial l}{\partial A_1}\right]^T \mathbf{1}_n$$

推导已完成，再一层一层带回去，即可得到l对$W_1, W_2, \mathbf{b}_1, \mathbf{b}_2$的导数。

参考资料

- 知乎-矩阵求导术：上篇和下篇。本文基本上是这两篇文章内容的重新整理。
- 刘建平Pinard系列博客，这个博客主要用于查缺补漏
- 教材：《矩阵分析与应用》，作者张贤达
- 查询手册：The Matrix Cookbook

数学 # 线性代数

< 矩阵求导总结（一）

傅里叶级数与傅里叶变换（一） >