Lecture 5 Part 1: Derivative of Matrix Determinant and Inverse

MIT 18.S096 Matrix Calculus For Machine Learning And Beyond

March 6 to March 10, 2024

Table of contents

1 Norms and derivatives

Terminology: continuous vector space + norm = *Banach space*

Given a vector space V, a norm $\|v\|$ for $v\in V$ is a map $V\to\mathbb{R}$ satisfying:

1. Non-negativity: $\|v\|\geq 0$, $\|v\|=0$ iff $v=0$
2. Scaling: $\|\alpha v\|=|\alpha|\|v\|$ for any $\alpha\in\mathbb{R}$
3. Triangle inequality: $\|u+v\|\leq\|u\|+\|v\|$

Example of a norm: Any inner product $u\cdot v$ gives a norm $\|u\|=\sqrt{u\cdot u}$.

Derivatives require a norm of both input and output. Why? Consider the following expression:

$$f(x+\delta x)-f(x)=f'(x)[\delta x]+o(\delta x),$$

where $o(\delta x)$ is small as $\delta x\to 0$. But how do we define "small"? $o(\delta x)$ is any function such that

$$\lim_{\delta x\to 0}\frac{\|o(\delta x)\|}{\|\delta x\|}=0.$$

2 Derivative of matrix determinant – Jacobi's formula

Theorem (Jacobi's formula). $d\det(A)=\operatorname{tr}(\operatorname{adj}(A)\,dA)\underset{\text{(if invertible)}}{=}\det(A)\operatorname{tr}(A^{-1}dA)$.

I find this theorem remarkably simple when written as the gradient, i.e.,

$$\nabla \det(A) = \text{cofactor}(A).$$

3 Direct proof

3.1 Preliminaries

Determinant. The determinant of a matrix is the volume scaling factor when that matrix is applied to a hypercube of volume 1. It is also the unique function that satisfies four important properties. These two interpretations are actually equivalent[1]. *Leibniz's formula* can be derived from these two interpretations, so it's generally regarded as the definition of determinant.

Laplace expansion. For an n-by-n matrix A, its determinant can be expressed (one can prove that this is equivalent to the Leibniz's formula) as a *Laplace/cofactor expansion* along its i-th row:

$$\det(A) = \sum_{j=1}^{n} A_{i,j} \underbrace{(-1)^{i+j} m_{i,j}}_{c_{i,j}},$$

where $m_{i,j}$ (called the (i,j)-*minor*) is the determinant of the submatrix obtained by removing the i-th row and j-th column of A. Meanwhile, $c_{i,j} = (-1)^{i+j} m_{i,j}$ is called the (i,j)-*cofactor*.

Cofactor matrix. The cofactor matrix of a square matrix A, denoted as $C(A)$, is another square matrix (of the same shape as A) in which the (i,j)-th entry holds the (i,j)-cofactor of A.

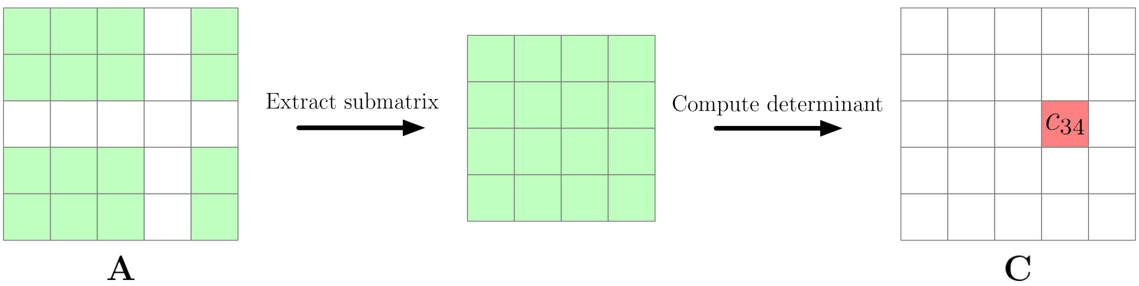

Figure 1. Given how cofactors are defined, the (i,j)-th entry of the cofactor matrix C is not dependent on any element on the i-th row or the j-th column of the original matrix.

Adjugate matrix. $\text{adj}(A) \triangleq C^T$. Properties (can be proven using Laplace expansion):

- $A\,\text{adj}(A) = \text{adj}(A)\,A = \det(A)\,I$
- If A is invertible (A is obviously square, but not necessarily invertible), then

$$\begin{aligned} \text{adj}(A) &= \det(A)\,A^{-1} \\ A^{-1} &= \det(A)^{-1}\text{adj}(A) \end{aligned}$$

1. Great Youtube video explaining this: https://www.youtube.com/watch?v=Sv7VseMsOQc

3.2 Proof (a cleaned-up version of [1])

Let $A \in \mathbb{R}^{n \times n}$. Trivially, $\det(A)$ can be thought of as a function $f: \mathbb{R}^{n \times n} \to \mathbb{R}$ of the elements of A:

$$\det(A) = f(A_{11}, A_{12}, \ldots, A_{nn}).$$

By chain rule, we have

$$\partial \det(A) = \sum_i \sum_j \frac{\partial f}{\partial A_{ij}} \partial A_{ij}. \tag{1}$$

To find $\partial f / \partial A_{ij}$, we express $\det(A)$ as the Laplace expansion along the i-th row of A:

$$\begin{aligned}
\frac{\partial(\det(A))}{\partial A_{ij}} &= \frac{\partial\left(\sum_{k=1}^n A_{ik}\, c_{ik}\right)}{\partial A_{ij}} \quad \text{(Laplace expansion)} \\
&= \sum_{k=1}^n \left[\frac{\partial A_{ik}}{\partial A_{ij}} c_{ik} + A_{ik} \frac{\partial c_{ik}}{\partial A_{ij}}\right] \quad \text{(product rule)} \\
&= \sum_{k=1}^n [\delta_{jk}\, c_{ik} + 0] \quad (c_{ik} \text{ is independent of } A_{ij}; \text{see Figure 1}) \\
&= c_{ij}.
\end{aligned}$$

Substituting this result back to Equation 1:

$$\begin{aligned}
\partial \det(A) &= \sum_i \sum_j c_{ij}\, \partial A_{ij} \\
&= C(A) \cdot \partial A \quad \text{(definition of matrix dot product)} \\
&= \operatorname{tr}(C(A)^T \partial A) \\
&= \operatorname{tr}(\operatorname{adj}(A)\, \partial A).
\end{aligned}$$

4 Fancy proof

Lemma. $\det(I + dA) = 1 + \operatorname{Tr}(dA)$.

Proof. The lecture gave a very brief proof, so here I want to expand on it.

The power expansion of $\det(1 + tA)$ is as follows [2, 3]:

$$\det(1 + tA) = 1 + t\operatorname{Tr}(A) + O(t^2),$$

where t is a scalar.

As we take $t \to 0$, we would have

$$\det(1 + (dt)A) = 1 + (dt)\operatorname{Tr}(A) = 1 + \operatorname{Tr}((dt)A)$$

To complete the proof, view $(dt)A$ in the equation above as dA, obtaining

$$\det(1 + dA) = 1 + \operatorname{Tr}(dA).$$

(Honestly, I don't know if doing so is 100% justified, but it's intuitive.)

Theorem. $d\det(A) = \det(A)\,\mathrm{tr}(A^{-1}\,dA)$.

$$
\begin{aligned}
d\det(A) &= \det(A + dA) - \det(A) \quad \text{(definition of differential)} \\
&= \det(A + AA^{-1}dA) - \det(A) \\
&= \det(A(I + A^{-1}dA)) - \det(A) \\
&= \det(A)\det(I + A^{-1}dA) - \det(A) \quad (\det(AB) = \det(A)\det(B)) \\
&= \det(A)\,(1 + \mathrm{tr}(A^{-1}\,dA)) - \det(A) \quad \text{(applying the lemma } (*)) \\
&= \det(A)\,\mathrm{tr}(A^{-1}\,dA)
\end{aligned}
$$

$(*)$ This step requires treating $A^{-1}dA$ as dA. Again, I found this to be a bit hand-wavy, but this was what the lecturer did and is intuitive: if dA is a small perturbation then $A^{-1}\,dA$ would be, too.

5 Applications

5.1 Derivative of the characteristic polynomial

Old derivation:

$$
\begin{aligned}
\frac{d}{dx}\prod_i (x-\lambda_i) &= \sum_i \prod_{j\neq i} (x-\lambda_j) \quad \text{(product rule for 2 or more functions)} \\
&= \left(\prod_i (x-\lambda_i)\right)\left(\sum_i (x-\lambda_i)^{-1}\right) \quad \text{(extracting common factor)}
\end{aligned}
$$

Derivation with new technology:

$$
\begin{aligned}
d(\det(xI - A)) &= \det(xI - A)\mathrm{Tr}((xI - A)^{-1}\,d(xI - A)) \\
&= \det(xI - A)\mathrm{Tr}((xI - A)^{-1}\,dx\,I) \quad (A \text{ is a constant}) \\
&= \det(xI - A)\mathrm{Tr}((xI - A)^{-1}\,dx\,) \\
&= \det(xI - A)\mathrm{Tr}((xI - A)^{-1}\,)\,dx \quad (dx \text{ is a scalar})
\end{aligned}
$$

A nice application of $d(\det(A))$ is solving for eigenvalues λ by applying Newton's method to $d(xI - A)$. Here is a piece of runnable JAX code:

```
from jax import config
config.update("jax_enable_x64", True)

import numpy as np
import scipy
import jax
import jax.numpy as jnp

# generate a random symmetric matrix (to have real eigenvalues)
np.random.seed(42)
pre_A = np.random.normal(size=(3, 3))
A = pre_A.T @ pre_A
A = jnp.array(A)

# find the groundtruth eigenvalues
print(scipy.linalg.eigvals(A))

# this is f'(x)/f(x)
def newton_update(x, A):
```

```
        return 1 / jnp.trace(
            jnp.linalg.inv( x * jnp.eye(A.shape[0]) - A )
        )

    # newton
    x = jnp.array([1.])  # initial guess
    for i in range(10):
        x = x - newton_update(x, A)
    print(x)
```

But this routine can't yield all eigenvalues at once, and which eigenvalue you get depends on the initial guess. Also the matrix inversion is expensive.

5.2 Derivative of log determinant

$$\begin{aligned} d(\log(\det(A))) &= \det(A^{-1})\, d(\det(A)) \quad \text{(scalar chain rule)} \\ &= \det(A)^{-1} \det(A)\, \mathrm{tr}(A^{-1}\, dA) \\ &= \mathrm{tr}(A^{-1})\, dA \end{aligned}$$

6 Derivative of matrix inverse

Some tricks:

$$A^{-1}A = I \rightarrow d(A^{-1}A) = 0 = d(A^{-1})A + A^{-1}dA$$

Therefore:

$$d(A^{-1}) = -A^{-1}dA\,A^{-1}$$

Using the key Kronecker identity $(A \otimes B)\,\mathrm{vec}(C) = \mathrm{vec}(BCA^T)$, obtain

$$\mathrm{vec}(d(A^{-1})) = \mathrm{vec}(-A^{-1}dA\,A^{-1}) = -(A^{-T} \otimes A^{-1})\,\mathrm{vec}(dA)$$

7 References

[1] Wikipedia page on Jacobi's formula

[2] https://terrytao.wordpress.com/2013/01/13/matrix-identities-as-derivatives-of-determinant-identities/

[3] https://math.stackexchange.com/questions/457242/detia-1-tra-deta-for-n-2-and-for-n2