Statistic

for machine learning

Tran Trong Khiem

AI lab tranning

2024/05/29

1. Solving systems of linear equations
2. Matrix calculus

Solving systems of linear equations

- We can represent this in matrix-vector form as follows:

$$Ax = b$$

- if we have m equations and n unknowns,
  - A will be a $m \times n$ matrix.
  - b will be a $m \times 1$ vector.
  - If $m = n$ (and A is full rank), there is a **single unique solution**.
  - If $m < n$, the system is **underdetermined**, so there is **not a unique solution**.
  - If $m > n$, the system is **overdetermined**.
    - there are **more constraints than unknowns**.

Solving square systems

In the case where $m = n$,

- can solve for x by computing an LU decomposition, $A = LU$,
- $x = U^{-1}L^{-1}b$
- L and U are both **triangular matrices**.
    - avoid taking matrix inverses, and use a method known as **backsubstitution** instead.

backsubstitution:

- First we write :

$$\begin{pmatrix} L_{11} & 0 & \cdots & 0 \\ L_{21} & L_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L_{n1} & L_{n2} & \cdots & L_{nn} \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \tag{7.219}$$

- solving $L_{11}y_1 = b_1$ to find y_1, then substitute this in to solve $L_{21}y_1 + L_{22}y_2 = b_2$

Solving underconstrained systems (least norm estimation)

Consider the **underconstrained setting**, where $m < n$.

- Assume the rows are **linearly independent**, so A is **full rank**.
- When $m < n$, there are **multiple possible solutions**.

$$\{\mathbf{x} : A\mathbf{x} = \mathbf{b}\} = \{\mathbf{x}_p + \mathbf{z} : \mathbf{z} \in \text{nullspace}(A)\}$$

  - Where x_p is any **particular solution**.
  - It is **standard to pick the particular solution** with minimal ℓ_2 norm.

$$\hat{x} = \arg\min \|x\|_2^2 \quad \text{subject to} \quad A\mathbf{x} = \mathbf{b}$$

  - can compute the minimal norm solution using the **right pseudo inverse:** $\mathbf{x}_{\text{pinv}} = A^T(AA^T)^{-1}\mathbf{b}$
  - suppose x is some other solutio, $Ax = b$, so we have $A(\mathbf{x} - \mathbf{x}_{\text{pinv}}) = 0$

$$(\mathbf{x} - \mathbf{x}_{\text{pinv}})^T \mathbf{x}_{\text{pinv}} = (\mathbf{x} - \mathbf{x}_{\text{pinv}})^T A^T (AA^T)^{-1}\mathbf{b} \quad = (A(\mathbf{x} - \mathbf{x}_{\text{pinv}}))^T (AA^T)^{-1}\mathbf{b}$$
$$= 0$$

Solving underconstrained systems (least norm estimation)

- We have $(\mathbf{x} - \mathbf{x}_{\text{pinv}}) \perp \mathbf{x}_{\text{pinv}}$. By **Pythagoras's theorem**, the norm of x is :

$$\|x\|_2 = \|x_{\text{pinv}} + x - x_{\text{pinv}}\|_2 = \|x_{\text{pinv}}\|_2 + \|x - x_{\text{pinv}}\|_2 \geq \|x_{\text{pinv}}\|_2$$

- We can also **solve the constrained optimization problem** :

$$L(x, \lambda) = x^T x + \lambda^T (Ax - b)$$

- Optimality conditions are :

$$\nabla_x L = 2x + A^T \lambda = 0, \quad \nabla_\lambda L = Ax - b = 0$$

- From the first condition we have $x = -\frac{A^T \lambda}{2}$. Thus $Ax = \frac{-AA^T \lambda}{2} = b$
- $\lambda = -2(AA^T)^{-1} b$. Hence $x = A^T (AA^T)^{-1} b$,

Solving overconstrained systems (least squares estimation)

If $m > n$, we have an **overdetermined solution**,

- does not have an **exact solution**.
- find the solution that **gets as close as possible** to **satisfying all of the constraints** specified by $Ax = b$.
- **Minimizing** the following **cost function** : $f(x) = \frac{1}{2}\|Ax - b\|_2^2$
- The **gradient** is given by :

$$g(x) = \frac{\partial f(x)}{\partial x} = A^T Ax - A^T b$$

- The optimum can be found by solving $g(x) = 0$. This gives :

$$A^T Ax = A^T b$$

- The corresponding solution $\hat{x}$ is the **ordinary least squares** (OLS) solution : $\hat{x} = (A^T A)^{-1} A^T b$

Solving overconstrained systems (least squares estimation)

- The quantity $A^{\dagger} = (A^T A)^{-1} A^T$ is the **left pseudo-inverse** of the (non-square) matrix A.
- The **Hessian** is given by: $H(x) = \frac{\partial^2 f(x)}{\partial x^2} = A^T A$
- If A is full rank, then H is **positive definite**, Since for any $v > 0$, we have

$$v^T (A^T A) v = (Av)^T (Av) = \|Av\|_2^2 > 0$$

- Hence in the **full rank case**, the least squares objective has a **unique global minimum**.

Derivatives

Consider a **scalar-argument function** $f : \mathbb{R} \to \mathbb{R}$

- Define its **derivative** at a point x : $f'(x) \hat{=} \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$
  - assuming the limit **exists.**
  - measures **how quickly the output changes** when we **move a small distance in input** space away from x.
  - We can interpret $f'(x)$ as : $f(x+h) \approx f(x) + f'(x)h$,for small h.
- We can compute a **finite difference approximation** to the **derivative** .
  - forward difference : $f'(x) \equiv \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$
  - central difference : $f'(x) \equiv \lim_{h \to 0} \frac{f(x+h/2) - f(x-h/2)}{h}$
  - backward difference : $f'(x) \equiv \lim_{h \to 0} \frac{f(x) - f(x-h)}{h}$
  - The **smaller the step size** h, the **better the estimate**
  - If h is **too small**, there can be errors due to numerical cancellation.

Gradients

Vector-argument functions $f : \mathbb{R}^n \to \mathbb{R}$

- Defining the **partial derivative** of f with **respect** to x_i to be :

$$\frac{\partial f}{\partial x_i}(x) = \lim_{h \to 0} \frac{f(x + he_i) - f(x)}{h}$$

  - where e_i is the i-th unit vector.

- The **gradient of a function** at a point x is the **vector of its partial derivatives**:

$$g = \frac{\partial f}{\partial x} = \nabla f = \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \frac{\partial f}{\partial x_2} \\ \vdots \\ \frac{\partial f}{\partial x_n} \end{pmatrix}$$

- We can write $g(x^*) = \frac{\partial f}{\partial x}\Big|_{x=x^*}$

Directional derivative

The **directional derivative** measures how much the function $f : \mathbb{R}^n \to \mathbb{R}$ **changes along a direction v** in space.

- $D_v f(x) = \lim_{h \to 0} \frac{f(x+h\mathbf{v}) - f(x)}{h}$
- Note that the **directional derivative** along v:

$$D_v f(x) = \nabla f(x) \cdot \mathbf{v}$$

Total derivative

- **suppose the function** has the form $f(t, x(t), y(t))$.
- Define the total derivative of f wrt t as follows:

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial y}\frac{dy}{dt}$$

- multiply both sides by the **differential** dt, we get the **total differential**: $df = \frac{\partial f}{\partial t}dt + \frac{\partial f}{\partial x}dx + \frac{\partial f}{\partial y}dy$

Jacobian

Consider a **function that maps a vector to another vector**, $f : \mathbb{R}^n \to \mathbb{R}^m$.

- The **Jacobian matrix** of this function is an $m \times n$ matrix of **partial derivatives**:

$$J_f(x) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix} = \begin{pmatrix} \nabla f_1(x)^T \\ \vdots \\ \vdots \\ \nabla f_m(x)^T \end{pmatrix}$$

Multiplying Jacobians and vectors

- The **Jacobian vector product** or **JVP** is defined as :

$$J_f(x)v = \begin{pmatrix} \nabla f_1(x)^T \\ \vdots \\ \nabla f_m(x)^T \end{pmatrix} v = \begin{pmatrix} \nabla f_1(x)^T v \\ \vdots \\ \nabla f_m(x)^T v \end{pmatrix}$$

Jacobian of a composition

Multiplying Jacobians and vectors

- The vector **Jacobian product** or **VJP** is defined as :

$$u^T J_f(x) = u^T \begin{pmatrix} \frac{\partial f}{\partial x_1} & \cdots & \frac{\partial f}{\partial x_n} \end{pmatrix} = \left(u \cdot \frac{\partial f}{\partial x_1}, \cdots, u \cdot \frac{\partial f}{\partial x_n} \right)$$

  - Jacobian matrix $J \in \mathbb{R}^{m \times n}$, $u \in R^m$
  - JVP is more efficient if $m \geq n$
  - VJP is more efficient if $m \leq n$.

Jacobian of a composition

- Let $h(x) = g(f(x))$. By the chain rule of calculus, we have :

$$J_h(x) = J_g(f(x)) \cdot J_f(x)$$

Hessian

Hessian:

- For a function $f : \mathbb{R}^n \to \mathbb{R}$ that is **twice differentiable,**
- **Hessian matrix** as the **(symmetric)** $n \times n$ **matrix**

$$H_f = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{pmatrix}$$

Functions that map vectors to scalars

- Consider a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$.
  - $\frac{\partial (a^T x)}{\partial x} = a$
  - $\frac{\partial (b^T A x)}{\partial x} = A^T b$
  - $\frac{\partial (x^T A x)}{\partial x} = (A + A^T)x$

Functions that map matrices to scalars

Consider a function $f : \mathbb{R}^{m \times n} \to \mathbb{R}$ which **maps a matrix to a scalar.**

- $$\frac{\partial f}{\partial X} = \begin{pmatrix} \frac{\partial f}{\partial x_{11}} & \frac{\partial f}{\partial x_{12}} & \cdots & \frac{\partial f}{\partial x_{1n}} \\ \frac{\partial f}{\partial x_{21}} & \frac{\partial f}{\partial x_{22}} & \cdots & \frac{\partial f}{\partial x_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial x_{m1}} & \frac{\partial f}{\partial x_{m2}} & \cdots & \frac{\partial f}{\partial x_{mn}} \end{pmatrix}$$
- Identities **involving quadratic forms**:
  - $\frac{\partial (a^T X b)}{\partial X} = ab^T$
  - $\frac{\partial (a^T X^T b)}{\partial X} = ba^T$
- **Identities involving matrix trace**:
  - $\frac{\partial \operatorname{tr}(AXB)}{\partial X} = A^T B^T$

Functions that map matrices to scalars

Identities involving matrix trace:

- $\frac{\partial \text{tr}(AXB)}{\partial X} = A^T B^T$
- $\frac{\partial \text{tr}(X^T A)}{\partial X} = A$
- $\frac{\partial \text{tr}(X^{-1} A)}{\partial X} = -X^{-T} A^T X^{-T}$
- $\frac{\partial \text{tr}(X^T A X)}{\partial X} = (A + A^T)X$

Identities involving matrix determinant

- $\frac{\partial \det(AXB)}{\partial X} = \det(AXB) \cdot X^{-T}$
- $\frac{\partial \log(\det(X))}{\partial X} = X^{-T}$