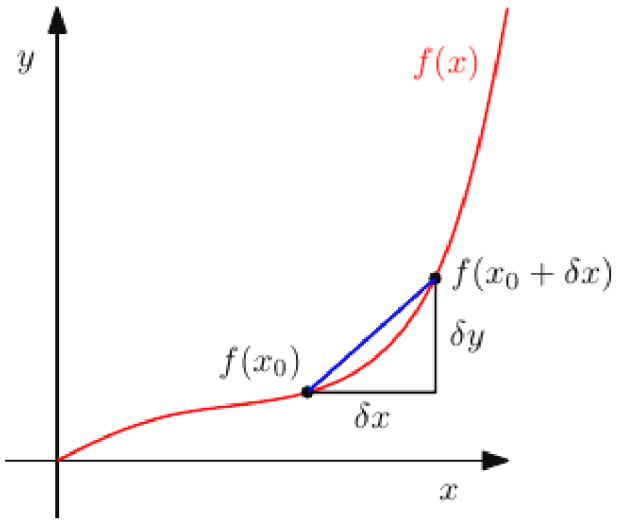
Part 5 - Vector Calculus

5.1 Differentiation of Univariate Functions



$$\frac{\delta y}{\delta x} = \frac{f(x + \delta x) - f(x)}{\delta x}$$

Menyatakan kemiringan garis secan dari dua titik, yakni titik-titik antara x_0 dan $x_0 + \delta x$.

Turunan (Derivative)

Untuk $h > 0$ turunan dari f di x terdefinisi sebagai limit

$$\frac{df}{dx} = f'(x) := \lim_{h \to 0} \frac{f(x + h) - f(x)}{h}$$

di mana $h = \delta x$,dan secan pada gambar menjadi tangen.

- Untuk suatu polinomial $f(x) = ax^n$ maka turunannya adalah $f'(x) = anx^{n-1}$

5.1.1 Deret Taylor

Polinomial Taylor

Polinomial derajat n dari $f : \mathbb{R} \to \mathbb{R}$ di x_0 didefinisikan sebagai

$$T_n(x) := \sum_{k=0}^{n} \frac{f^(k)(x_0)}{k!}(x - x_0)^k$$

di mana $f^(k)(x_0)$ adalah turunan ke-k dari f di x_0 dan $\frac{f^(k)(x_0)}{k!}$ adalah koefisien polinomial.

Deret Taylor

Untuk suatu fungsi $f \in \mathcal{C}^\infty,\ f : \mathbb{R} \to \mathbb{R}$, baris taylor dari f di x_0 didefinisikan sebagai:

$$T_\infty(x) = \sum_{k=0}^{\infty} \frac{f^(k)(x_0)}{k!}(x - x_0)^k$$

Untuk $x_0 = 0$ kita mendapatkan *deret mclaurin*. Jika $f(x) = T_\infty(x)$ maka f disebut *analitik*.

5.1.2 Aturan Diferensiasi

- Product rule: $(f(x)g(x))' = f(x)'g(x) + g'(x)f(x)$
- Quotient rule: $\left(\frac{f(x)}{g(x)}\right)' = \frac{f'(x)g(x) - g'(x)f(x)}{g(x)^2}$
- Sum rule: $(f(x) + g(x))' = f'(x) + g'(x)$
- Chain rule: $(g(f(x)))' = (g \circ f)'(x) = g'(f(x))f'(x)$

5.2 Partial Differentiation and Gradients

Partial Differentiation

Untuk suatu fungsi $f : \mathbb{R}^n \to \mathbb{R}, x \to f(x), x \in \mathbb{R}^n$ dari n variabel $x_1, \ldots, x_n$, diferensiasi parsial didefinisikan sebagai:

$$\begin{aligned}
\frac{\delta f}{\delta x_1} &= \lim_{h \to 0} \frac{f(x_1 + h, x_2, \ldots, x_n) - f(x)}{h} \\
&\vdots \\
\frac{\delta f}{\delta x_n} &= \lim_{h \to 0} \frac{f(x_1, \ldots, x_{n-1}, x_n + h) - f(x)}{h}
\end{aligned}$$

dan koleksikan dalam vektor baris

$$J = \nabla_x f = grad\ f = \frac{df}{dx} = \begin{bmatrix} \frac{\delta f(x)}{\delta x_1} & \frac{\delta f(x)}{\delta x_2} & \dots & \frac{\delta f(x)}{\delta x_n} \end{bmatrix} \in \mathbb{R}^{1 \times n}$$

Vektor baris ini disebut *gradien* dari f atau ***Jacobian***.

Contoh 1

Untuk $f(x, y) = x^2y + xy^3 \in \mathbb{R}$, turunan parsialnya adalah:

$$\frac{\delta f}{\delta x} = 2xy + y^3$$
$$\frac{\delta f}{\delta y} = x^2 + 3xy^2$$

dan gradiennya adalah $J = [2xy + y^3 \quad x^2 + 3xy^2]$

5.2.1 Partial Differentiation Rules

- Product Rule: $\frac{\delta}{\delta x}(f(x)g(x)) = \frac{\delta f}{\delta x}g(x) + f(x)\frac{\delta g}{\delta x}$
- Sum Rule: $\frac{\delta}{\delta x}(f(x) + g(x)) = \frac{\delta f}{\delta x} + \frac{\delta g}{\delta x}$
- Chain Rule: $\frac{\delta}{\delta x}(g \circ f)(x) = \frac{\delta}{\delta x}(g(f(x))) = \frac{\delta g}{\delta f}\frac{\delta f}{\delta x}$

Contoh 2

Ambil $f(x_1, x_2) = x_1^2 + 2x_2$, di mana $x_1 = \sin t$ dan $x_2 = \cos t$, maka

$$\begin{aligned} \frac{\delta f}{\delta t} &= \frac{\delta f}{\delta x_1}\frac{\delta x_1}{\delta t} + \frac{\delta f}{\delta x_2}\frac{\delta x_2}{\delta t} \\ &= 2\sin t \frac{\delta \sin t}{\delta t} + 2\delta \cos \frac{t}{\delta t} \\ &= 2\sin t \cos t - 2\sin t \\ &= 2\sin t(\cos t - 1) \end{aligned}$$

adalah turunan t dari f.

5.3 Gradient of Vector-Valued Functions

Misal kita punya fungsi-fungsi $f_1, f_2, \dots, f_m$ di mana masing-masing fungsi menerima variabel $x = [x_1 \quad \dots \quad x_n]^T$. Kita bisa susun fungsi-fungsi tersebut menjadi:

$$f = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_m(x) \end{bmatrix}_{1 \times m} \in \mathbb{R}^m$$

yang mana bentuk f ini disebut **vector-valued functions**. Gradien dari f dijabarkan sebagai:

$$\frac{df}{dx} = \begin{bmatrix} \frac{\delta f_1}{\delta x_1} & \frac{\delta f_1}{\delta x_2} & \cdots & \frac{\delta f_1}{\delta x_n} \\ \frac{\delta f_2}{\delta x_1} & \frac{\delta f_2}{\delta x_2} & \cdots & \frac{\delta f_2}{\delta x_n} \\ \vdots & \vdots & & \vdots \\ \frac{\delta f_m}{\delta x_1} & \frac{\delta f_m}{\delta x_2} & \cdots & \frac{\delta f_m}{\delta x_n} \end{bmatrix}_{m \times n}, J(i,j) = \frac{\delta f_i}{\delta x_j}$$

Contoh 3

Diberikan

$$f(x) = Ax,\ f(x) \in \mathbb{R}^M,\ A \in \mathbb{R}^{M \times N},\ x \in \mathbb{R}^N$$

Untuk menghitung df/dx pertama kita tentukan dimensi turunannya yakni $M \times N$. Turunan parsial f untuk setiap x_j adalah:

$$f_i(x) = \sum_{j=1}^{n} a_{ij} x_j \implies \frac{\delta f_i}{\delta x_j} = A_{ij}$$

$\therefore df/dx$ tidak lain adalah A.

Contoh 4 (Gradient of Least-Square Loss in Linear Model)

Untuk suatu model linear

$$y = \theta \tag{1}$$

di mana $\theta \in \mathbb{R}^D$ adalah suatu parameter vektor, $\in \mathbb{R}^{N \times D}$ sebuah input features, dan $y \in \mathbb{R}^N$ adalah observasi terkait. kita definisikan fungsi:

$$L(e) = \|e\|^2 \tag{2}$$
$$e(\theta) = y - \theta \tag{3}$$

fungsi L disebut **Least-Square Loss**. Kita akan mencari $\frac{dL}{d\theta}$.
Pertama, kita cari dimensi turunannya, karena fungsi $L : R^D \mapsto R$ maka $\frac{dL}{d\theta} \in \mathbb{R}^{1 \times D}$.

Dengan menerapkan chain rule didapat:

$$\frac{dL}{d\theta} = \frac{dL}{de}\frac{de}{d\theta}$$

dengan elemen ke-d adalah:

$$\frac{dL}{d\theta}[1,d] = \sum_{n=1}^{N} \frac{dL}{de}[n]\frac{de}{d\theta}[n,d]$$

lalu

$$\|e\| = e^T e \implies \frac{dL}{de} = 2e^T, \text{dan}$$
$$\frac{de}{d\theta} = -$$

$\therefore \frac{dL}{d\theta} = -2e^T = -2(y^T - {}^T\theta^T \quad)_{N\times D} \in \mathbb{R}^{1\times D}$
$\underset{1\times N}{}$

5.4 Gradients of Matrices

Misalkan kita punya matriks $A \in \mathbb{R}^{m\times n},\ B \in \mathbb{R}^{p\times}$, jika kita hendak menghitung $\delta A/\delta B$ maka setiap entri A_{ij} diturunkan terhadap entri B_{kl} sehingga kita dapatkan **Jacobian tensor 4D**:

$$J_{ijkl} = \frac{\delta A_{ij}}{\delta B_{kl}},\ J \in \mathbb{R}^{m\times n\times p\times}$$

> fungsi yang memetakan objek d-dimensional ke objek e-dimensional, Jacobian-nya punya rank $d + e$.

Untuk mempermudah, perhatikan bahwa $\mathbb{R}^{m\times n}$ **isomorfik** dengan $\mathbb{R}^{mn}$.
Sehingga kita bisa reshape matriks menjadi matriks vektor dengan panjang mn, sebagai contoh:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}_{2\times 2} \to vec(A) = \begin{bmatrix} a_{11} \\ a_{21} \\ a_{12} \\ a_{22} \end{bmatrix}_{1\times 4}$$

maka Jacobian bisa ditulis sebagai:

$$J = \frac{\delta vec(A)}{\delta vec(B)^T} \in \mathbb{R}^{mn \times p}$$

5.5 Useful Identities for Computing Gradients

$$\frac{\partial}{\partial \boldsymbol{X}} \boldsymbol{f}(\boldsymbol{X})^\top = \left(\frac{\partial \boldsymbol{f}(\boldsymbol{X})}{\partial \boldsymbol{X}}\right)^\top \tag{5.99}$$

$$\frac{\partial}{\partial \boldsymbol{X}} \mathrm{tr}(\boldsymbol{f}(\boldsymbol{X})) = \mathrm{tr}\left(\frac{\partial \boldsymbol{f}(\boldsymbol{X})}{\partial \boldsymbol{X}}\right) \tag{5.100}$$

$$\frac{\partial}{\partial \boldsymbol{X}} \det(\boldsymbol{f}(\boldsymbol{X})) = \det(\boldsymbol{f}(\boldsymbol{X}))\mathrm{tr}\left(\boldsymbol{f}(\boldsymbol{X})^{-1}\frac{\partial \boldsymbol{f}(\boldsymbol{X})}{\partial \boldsymbol{X}}\right) \tag{5.101}$$

$$\frac{\partial}{\partial \boldsymbol{X}} \boldsymbol{f}(\boldsymbol{X})^{-1} = -\boldsymbol{f}(\boldsymbol{X})^{-1}\frac{\partial \boldsymbol{f}(\boldsymbol{X})}{\partial \boldsymbol{X}}\boldsymbol{f}(\boldsymbol{X})^{-1} \tag{5.102}$$

$$\frac{\partial \boldsymbol{a}^\top \boldsymbol{X}^{-1}\boldsymbol{b}}{\partial \boldsymbol{X}} = -(\boldsymbol{X}^{-1})^\top \boldsymbol{a}\boldsymbol{b}^\top (\boldsymbol{X}^{-1})^\top \tag{5.103}$$

$$\frac{\partial \boldsymbol{x}^\top \boldsymbol{a}}{\partial \boldsymbol{x}} = \boldsymbol{a}^\top \tag{5.104}$$

$$\frac{\partial \boldsymbol{a}^\top \boldsymbol{x}}{\partial \boldsymbol{x}} = \boldsymbol{a}^\top \tag{5.105}$$

$$\frac{\partial \boldsymbol{a}^\top \boldsymbol{X}\boldsymbol{b}}{\partial \boldsymbol{X}} = \boldsymbol{a}\boldsymbol{b}^\top \tag{5.106}$$

$$\frac{\partial \boldsymbol{x}^\top \boldsymbol{B}\boldsymbol{x}}{\partial \boldsymbol{x}} = \boldsymbol{x}^\top(\boldsymbol{B} + \boldsymbol{B}^\top) \tag{5.107}$$

$$\frac{\partial}{\partial \boldsymbol{s}}(\boldsymbol{x} - \boldsymbol{A}\boldsymbol{s})^\top \boldsymbol{W}(\boldsymbol{x} - \boldsymbol{A}\boldsymbol{s}) = -2(\boldsymbol{x} - \boldsymbol{A}\boldsymbol{s})^\top \boldsymbol{W}\boldsymbol{A} \quad \text{for symmetric } \boldsymbol{W} \tag{5.108}$$

5.6 Backpropagation and Automatic Differentiation

5.6.1 Gradients in Deep Network

Dalam jaringan syaraf dengan banyak lapisan K sehingga $i = 1, 2, \ldots, K$, kita punya fungsi

$$f(x_{i-1}) = \sigma(A_{i-1}x_{i-1} + b_{i-1})$$

pada layer ke-i, dengan x_{i-1} adalah output dari layer sebelumnya. Sebagai contoh jika kita punya input x diuraikan berikut:

$$\begin{aligned} f_0 &= x \\ f_i &= \sigma_i(A_{i-1}f_{i-1} + b_{i-1}),\ i = 1, 2, \ldots, K \end{aligned}$$

di mana σ adalah fungsi aktivasi, matriks A adalah beban(*weight*), dan b adalah bias. Dalam konteks optimasi, kita ingin mencari $A_j, B_j,\ j = 0, 1, \ldots, K$ sehingga *squared-loss*:

$$L(\theta) = \|y - f_K(\theta, x)\|^2$$

adalah minimal di mana $\theta = \{A_0, B_0, \ldots, A_{K-1}, B_{k-1}\}$.
Untuk mendapatkan gradien-gradien terhadap himpunan parameter θ, kita membutuhkan turunan parsial dari L terhadap parameter $\theta_j = \{A_j, b_j\}$ untuk setiap layer $j = 0, 1, \ldots, K-1$ menggunakan chain rule.

5.6.2 Automatic Differentiation

Automatic differentiation adalah sebuah metode diferensiasi dengan:

- Memecah fungsi jadi *graph of operations*.
- Gunakan **chain rule** secara lokal di setiap node.
- Hasilkan gradien yang tepat (presisi floating point), dengan biaya seefisien evaluasi fungsi.
  Jika kita punya alur data dari input x ke output y dengan melewati variabel a dan b maka untuk mendapatkan turunan $\frac{dy}{dx}$ kita memakai *chain rule* sebagai berikut:

  $$\frac{dy}{dx} = \frac{dy}{db}\frac{db}{da}\frac{da}{dx}$$

  Karena perkalian matriks bersifat asosiatif, maka pesamaan bisa diselesaikan dengan memilih dua cara:

  $$\frac{dy}{dx} = \left(\frac{dy}{db}\frac{db}{da}\right)\frac{da}{dx} \tag{1}$$
  $$\frac{dy}{dx} = \frac{dy}{db}\left(\frac{db}{da}\frac{da}{dx}\right) \tag{2}$$

  persamaan (1) disebut *reverse mode* sebab gradien bergerak terbalik, sedangkan persamaan (2) disebut *forward mode* sebab gradien bergerak bersama data dari kiri ke kanan dalam *graph of operation*.
  Dalam konteks deep network, di mana dimensi input sering kali jauh lebih besar dari pada dimensi label, *reverse mode* secara komputasi jauh lebih efisien sehingga akan dipakai, *reverse mode* inilah yang disebut **back propagation***.

5.7 Higher-Order Derivatives

Hesian Matrix

Hesian yang dinotasikan sebagai $\nabla^2_{x,y} f(x,y)$ adalah matriks yang menghimpun turunan kedua dari sebuah fungsi. Hesian menunjukan kelengkungan lokal dari sebuah fungsi di (x,y).

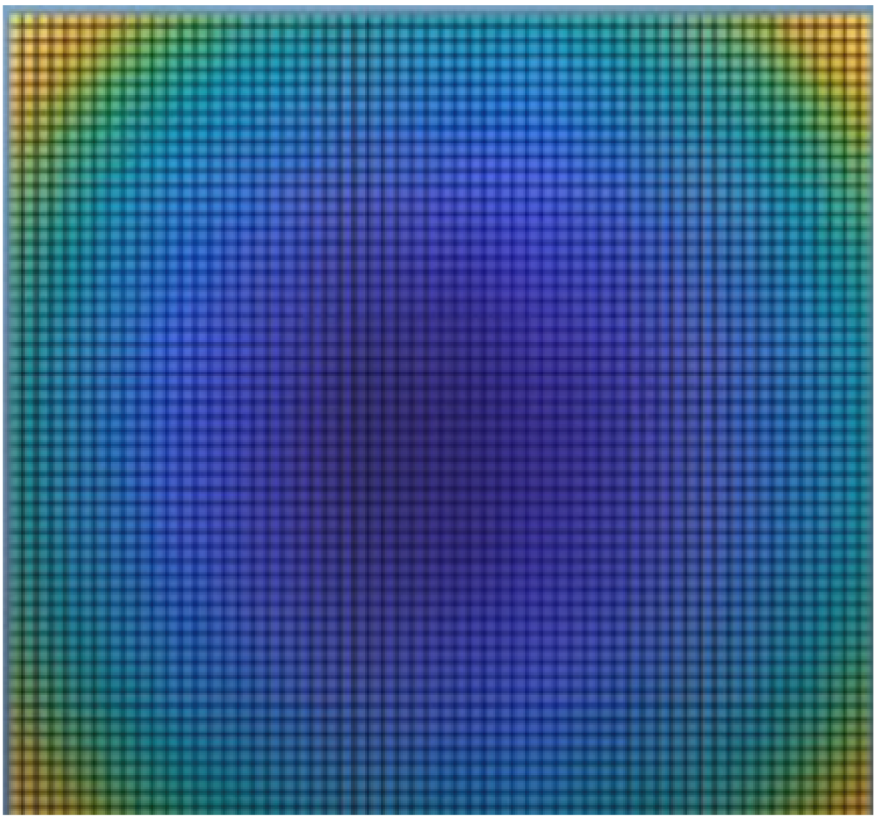
dalam konteks stabilitas training, **Hesian** memberikan informasi apakah titik yang ditemukan gradient descent itu minimum, maksimum, atau saddle.

Contoh 5

Untuk $f(x) = x^2 + y^2$ maka:

$$\begin{aligned} J &= [2x \quad 2y] \\ &= \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \end{aligned}$$

- Jika determinan dari Hesian adalah positif maka titik tersebut adalah maksimum atau minimum
- Jika h_{11} adalah positif maka dia adalah titik minimum, sedangkan jika dia negatif, dia adalah titik maksimum
  dari $f(x)$ didapati visualisasi sebagai berikut:



5.8 Linearization and Multivariate Taylor Series

5.8.1 Linearization dalam Satu Variabel

Untuk fungsi $f : \mathbb{R} \to \mathbb{R}$, aproksimasi linear di sekitar titik $x_0 \in \mathbb{R}$ adalah:

$$f(x) \;\; f(x_0) + f'(x_0)(x - x_0)$$

dengan $x \in \mathbb{R}$ variabel dan $x_0 \in \mathbb{R}$ titik pusat aproksimasi, yaitu titik di mana kita "mendekatkan" fungsi.

5.8.2 Linearization dalam Banyak Variabel

Sekarang untuk fungsi $f : \mathbb{R}^n \to \mathbb{R}$, aproksimasi linear di sekitar $\mathbf{x}_0 \in \mathbb{R}^n$ adalah:

$$f(\mathbf{x}) \;\; f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^\top (\mathbf{x} - \mathbf{x}_0)$$

5.8.3 Multivariate Taylor Expansion (sampai orde 2)

Untuk fungsi skalar $f : \mathbb{R}^n \to \mathbb{R}$, ekspansi Taylor sampai orde dua di sekitar $\mathbf{x}_0$ adalah:

$$f(\mathbf{x}) \;\; f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^\top (\mathbf{x} - \mathbf{x}_0) + \tfrac{1}{2} (\mathbf{x} - \mathbf{x}_0)^\top (\mathbf{x}_0) (\mathbf{x} - \mathbf{x}_0)$$

Contoh 6

Ambil $f(x, y) = x^2 + xy + y^2$

- Input $\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}$, lalu ambil suatu $\mathbf{x}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$.
- $f(\mathbf{x}_0) = 0$.
- Gradien:

$$\nabla f(x, y) = \begin{bmatrix} 2x + y \\ x + 2y \end{bmatrix}, \quad \nabla f(\mathbf{x}_0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

- Hessian:

$$= \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

Maka Taylor expansion di sekitar $(0, 0)$:

$$f(x,y) \;\; 0 + 0 + \tfrac{1}{2}\begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = x^2 + xy + y^2$$