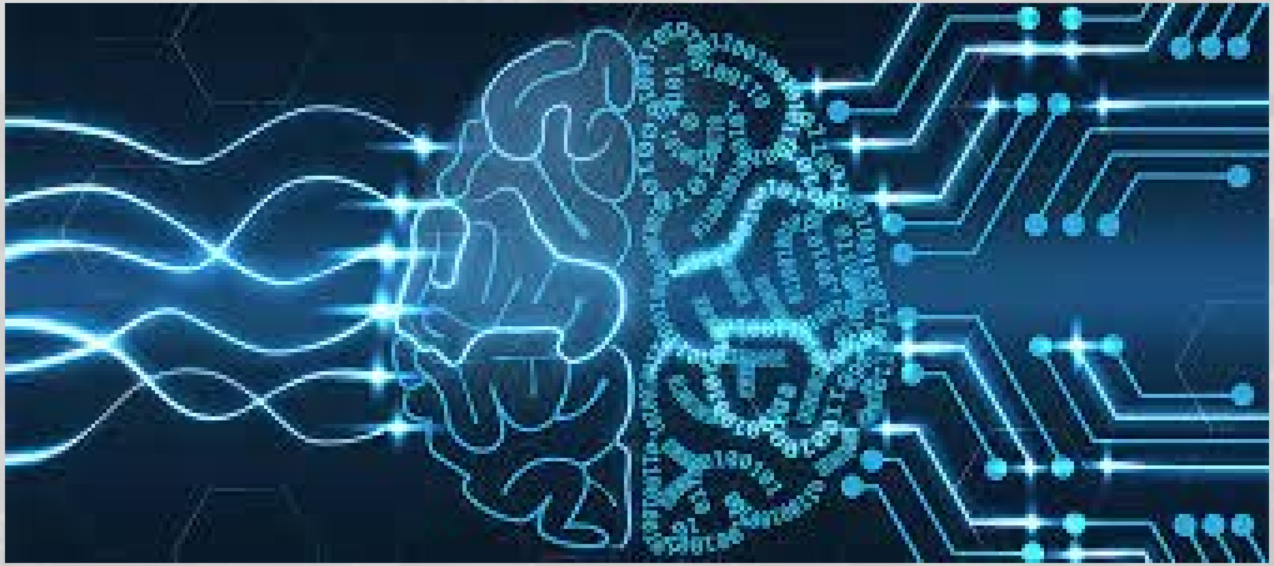
14624533
DEEP LEARNING



Mathematics for Machine Learning -

Numerical Computation

PENGAMPU

Dr. Fajar Astuti Hermawati, S.Kom.,M.Kom.

Bagus Hardiansyah, S.Kom.,M.Si

Andrey Kartika Widhy H., S.Kom., M.Kom.

Capaian Pembelajaran

- **Sub-CPMK-1**: Mampu mengidentifikasi konsep dasar pemelajaran mendalam, konsep matematika dan mesin pemelajar yang mendasari prinsip-prinsip algoritma cerdas serta menentukan karakteristik permasalahan yang dapat diselesaikan dengan algoritma deep learning [C2, A3]

Numerical Computation

Gradient-Based Optimization

Jacobian Matrices

Hessian Matrices

Why **Numerical Computation**?

- Algoritma pembelajaran mesin biasanya membutuhkan perhitungan numerik yang tinggi.
- Ini biasanya merujuk pada algoritma yang memecahkan masalah matematika dengan metode yang memperbarui perkiraan solusi melalui proses iteratif, daripada menurunkan rumus secara analitik untuk memberikan ekspresi simbolis untuk solusi yang tepat.
- Operasi umum termasuk **optimisasi** (menemukan nilai argumen yang meminimalkan atau memaksimalkan suatu fungsi) dan menyelesaikan sistem persamaan linier.

Numerical concerns for implementations of deep learning algorithms

- Algoritma sering ditentukan dalam bentuk **bilangan real**; bilangan real tidak dapat diimplementasikan dalam komputer yang terbatas
  - Apakah algoritma masih berfungsi saat diimplementasikan dengan **jumlah bit yang terbatas**.
- Apakah perubahan kecil pada input ke suatu fungsi menyebabkan perubahan besar pada output?
  - Kesalahan pembulatan, noise, kesalahan pengukuran dapat menyebabkan perubahan besar
  - Pencarian iteratif untuk input terbaik sulit dilakukan

Overflow and Underflow

- Kesulitan mendasar dalam melakukan matematika kontinu pada komputer digital adalah kita perlu **merepresentasikan bilangan real tak terhingga banyaknya dengan pola bit yang jumlahnya terhingga**.
- Ini berarti bahwa untuk hampir semua bilangan real, kita mengalami beberapa **kesalahan perkiraan** saat merepresentasikan bilangan di komputer. Dalam banyak kasus, ini hanyalah **kesalahan pembulatan**.
- **Galat pembulatan bermasalah**, terutama jika digabungkan di banyak operasi, dan dapat menyebabkan algoritma yang bekerja secara teori gagal dalam praktiknya jika tidak dirancang untuk **meminimalkan akumulasi kesalahan pembulatan**.

Overflow and Underflow

- Salah satu bentuk kesalahan pembulatan yang sangat merusak adalah ***underflow***.
- ***Underflow*** terjadi ketika angka mendekati nol dibulatkan menjadi nol.
- Banyak fungsi berperilaku berbeda secara kualitatif ketika argumen mereka nol daripada angka positif kecil.
- Misalnya, kita biasanya ingin menghindari pembagian dengan nol(divide by zero) atau mengambil logaritma nol
  - ini biasanya diperlakukan sebagai $-\infty$ (infinite), yang kemudian menjadi bukan angka (NaN) jika digunakan untuk banyak operasi aritmatika selanjutnya

Overflow and Underflow

- Bentuk lain dari kesalahan numerik yang sangat merusak adalah Overflow.
- Overflow terjadi ketika **angka dengan magnitudo besar** didekati sebagai ∞ atau $-\infty$.
  - Aritmatika lebih lanjut biasanya akan mengubah nilai tak terhingga ini menjadi nilai bukan angka (NaN)

Overflow and Underflow

- Salah satu contoh fungsi yang harus distabilkan terhadap underflow dan overflow adalah **fungsi softmax**.
- Fungsi softmax sering digunakan untuk memprediksi probabilitas yang terkait dengan distribusi multinoulli.
- Fungsi softmax didefinisikan sbg:

$$\text{softmax}(\boldsymbol{x})_i = \frac{\exp(x_i)}{\sum_{j=1}^{n} \exp(x_j)}.$$

Overflow and Underflow

- Pertimbangkan apa yang terjadi ketika semua xi sama dengan beberapa konstanta *c*. Secara analitis, kita dapat melihat bahwa semua keluaran harus sama dengan 1/*n*
  - Secara numerik, hal ini mungkin tidak terjadi jika *c* bermagnitudo besar.
  - Jika c sangat negatif, maka *exp*(*c*) akan berkurang.
  - Ini berarti penyebut softmax akan menjadi 0, sehingga hasil akhirnya tidak terdefinisi.
  - Ketika *c* sangat besar dan positif, *exp*(*c*) akan overflow, sekali lagi menghasilkan ekspresi secara keseluruhan menjadi tidak terdefinisi.
- Kedua kesulitan ini dapat diatasi dengan mengevaluasi **softmax**(*z*) di mana $z = x - \max_i x_i$.
  - Aljabar sederhana menunjukkan bahwa nilai fungsi softmax tidak diubah secara analitik dengan menambahkan atau mengurangkan skalar dari vektor masukan.
  - Mengurangi $\max_i x_i$ menghasilkan argumen terbesar untuk *exp* menjadi **0**, yang mengesampingkan kemungkinan overflow.
  - Demikian juga, setidaknya satu suku dalam penyebut memiliki nilai 1, yang mengesampingkan kemungkinan underflow pada penyebut yang mengarah ke pembagian dengan nol.

Poor Conditioning

- Pengkondisian mengacu pada **seberapa cepat suatu fungsi berubah** sehubungan dengan perubahan kecil pada inputnya.
- Fungsi yang **berubah dengan cepat ketika inputnya sedikit terganggu** dapat **menimbulkan masalah bagi perhitungan** ilmiah karena **kesalahan pembulatan dalam pindaian input menghasilkan perubahan besar pada output**.

Poor Conditioning

- Misalkan fungsi $f(x) = \boldsymbol{A}^{-1}x$. Ketika $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ memiliki dekomposisi nilai eigen, **nomor kondisinya (*condition number*)** adalah

$$\max_{i,j} \left| \frac{\lambda_i}{\lambda_j} \right|.$$

- Ini adalah **rasio besarnya nilai eigen terbesar dan terkecil**.
- Ketika **angka ini besar**, **inversi matriks sangat sensitif terhadap kesalahan input**

Poor Conditioning

- Sensitivitas ini adalah **properti intrinsik** dari matriks itu sendiri, **bukan hasil kesalahan pembulatan** selama inversi matriks.
- Matriks yang dikondisikan dengan buruk **memperkuat kesalahan** yang sudah ada sebelumnya **saat kita mengalikan dengan invers** matriks yang sebenarnya.
- Dalam prakteknya, kesalahan tersebut akan **diperparah** lebih lanjut dengan **kesalahan numerik dalam proses inversi** itu sendiri

Numerical Computation

Gradient-Based Optimization

meminimalkan $f(x)$ **meminimalkan $-f(x)$**

meminimalkan atau memaksimalkan suatu fungsi dengan superskrip $*$

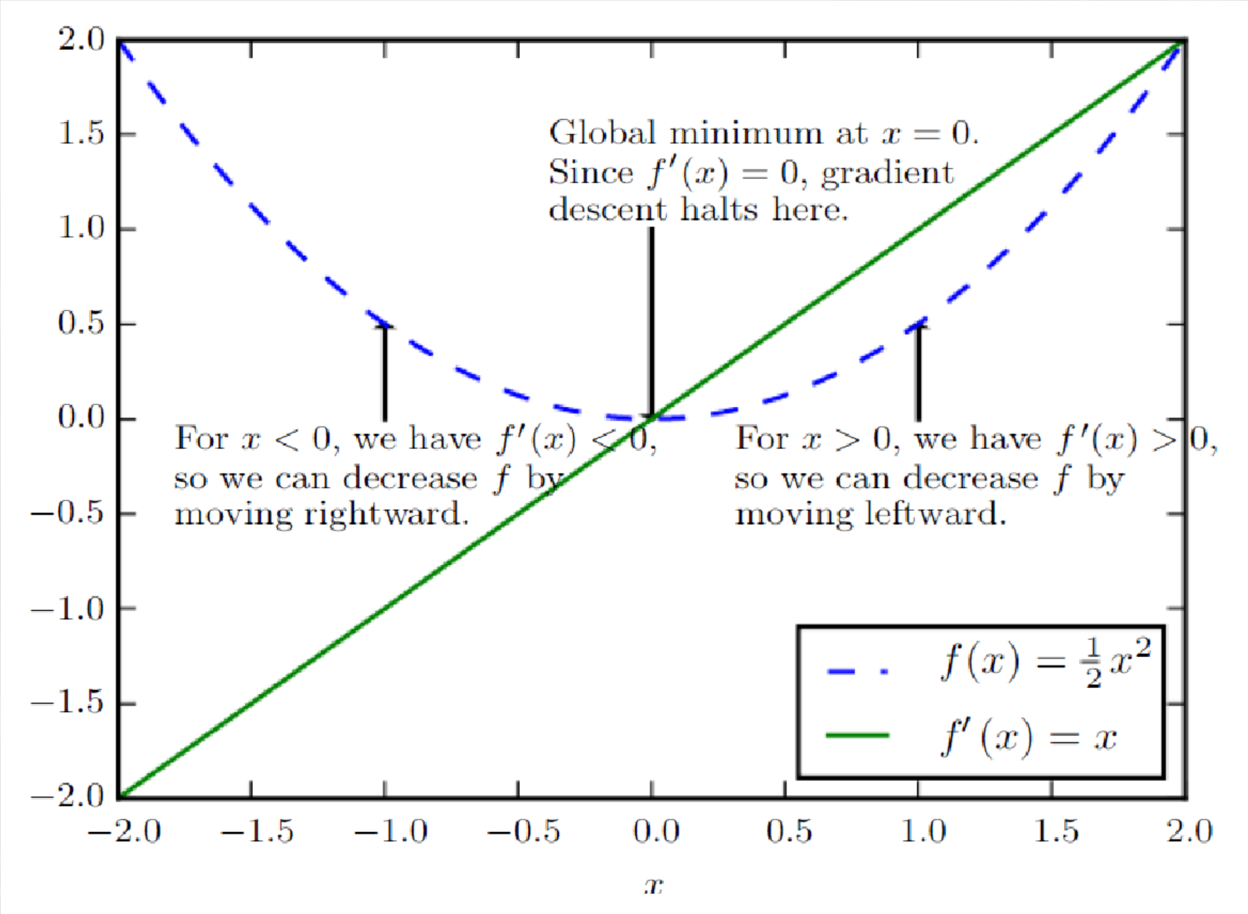
$$x^{*} = \arg\min f(x)$$

Fungsi $y = f(x)$, dimana x dan y adalah bilagnan real

$f'(x)$ dititik x

$$f(x + \in) \approx f(x) + \in f'(x)$$

Gradient-Based Optimization



metode gradient descent bekerja:

$$f(x)\frac{1}{2}x^2$$

Langkah-langkah dalam Gradient Descent:

1. Menentukan Turunan (Gradient) dari Fungsi:

Turunan pertama dari $f(x)$ adalah:

$$f'(x) = \frac{\partial}{\partial x}\left(\frac{1}{2}x^2\right) = x$$

Gradient dari fungsi $f(x)$ pada titik x adalah $f'(x) = x$.

2. Memilih Learning Rate (α)

Misalkan kita memilih learning rate α misalnya $\alpha = 0,1$

3. Proses iterative gradient descent

Dimulai dari titik awal x_0 (Misalnya $x_0 = 2$), kita melakukan pembaruan iterative pada x menggunakan formula :

$$x_{new} = x_{old} - \alpha . f'(x_{old})$$

4. Contoh Iterasinya

Iterasi 1

$\boldsymbol{x_0 = 2}$

Hitung gradient : $f'(x_0) = 2$

Pembaruan : $x_1 = 2 - 0,1 \times 2 = 2 - 0,2 = 1,8$

Iterasi 2

$x_1 = 1,8$

Hitung gradient : $f'(x_1) = 1,8$

Pembaruan : $x_2 = 1,8 - 0,1 \times 1,8 = 1,8 - 0,18 = 1,62$

Iterasi 3

$x_2 = 1,62$

Hitung gradient : $f'(x_1) = 1,62$

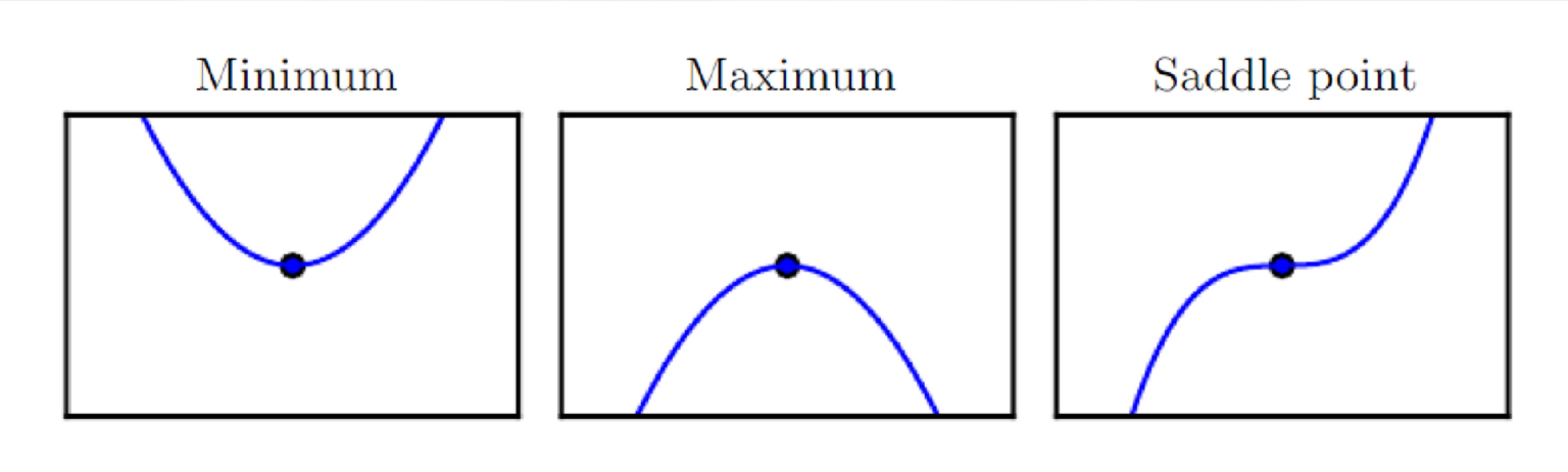
Pembaruan : $x_3 = 1,62 - 0,1 \times 1,62 = 1,62 - 0,162 = 1,458$

Proses ini di ulang sampai x mendekati titik minimum, yaitu $x = 0$

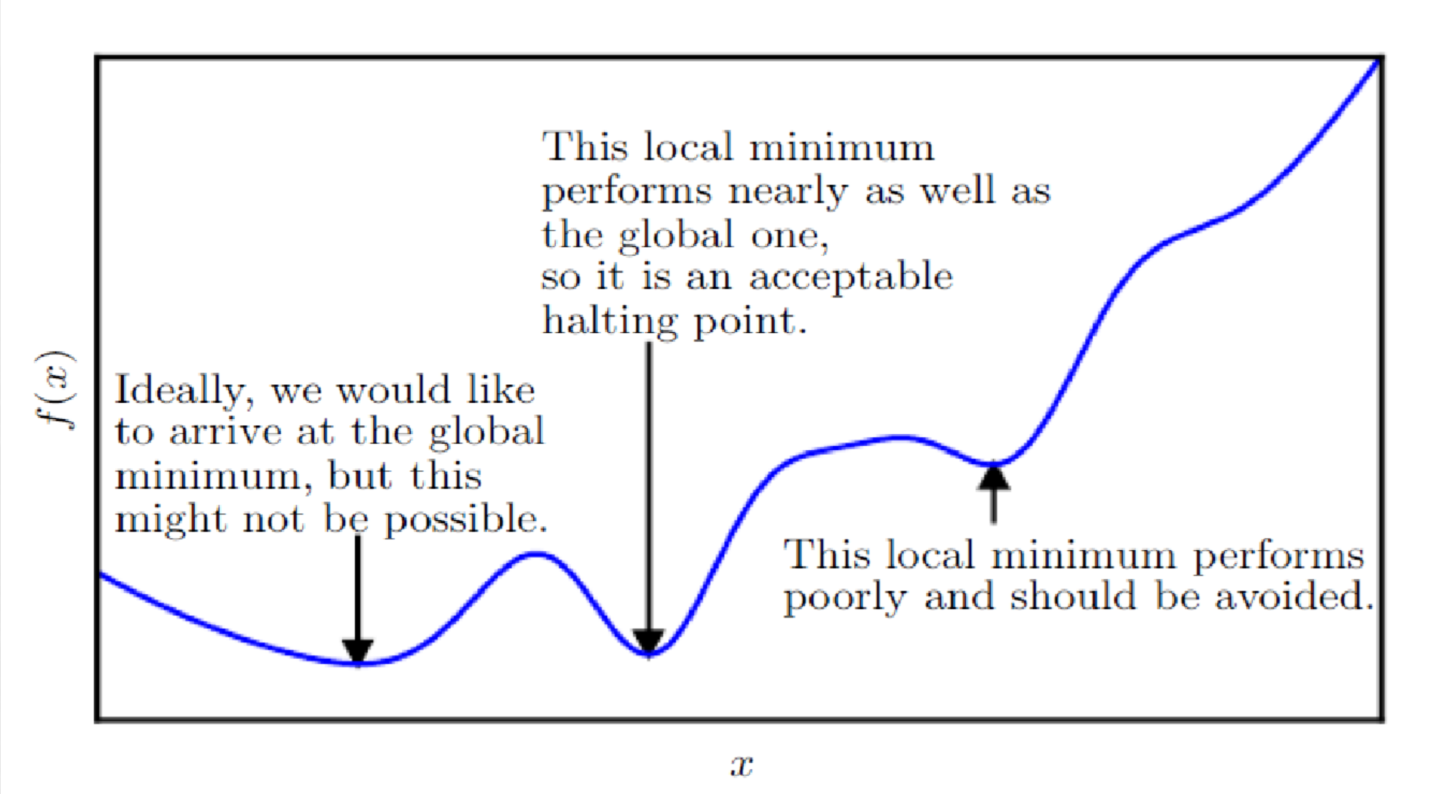
Maka Hasil Akhir :

Dengan megikuti poses gradient descent, maka nilai x akan semakin mendekati nilai minimum dari fungsi $f(x)$, yaitu $x = 0$, yang merupakan titik minimum global dari fungsi ini.

Gradient-Based Optimization



Gradient-Based Optimization



Gradient-Based Optimization

$$f: \mathbb{R}^2 \to \mathbb{R}$$

Turunan parsial $\frac{\partial}{\partial_{xi}} f(x)$

Gradien $\nabla_x f(x)$

Gradient-Based Optimization

$f(x + \alpha u)$ **terhadap** α **dievaluasi di** $\alpha = 0$

$\frac{\partial}{\partial \alpha} f(x + \alpha u)$ **bernilai** $u^T \nabla f(x) ketika\ \alpha = 0$

$$\min_{u,u^T u=1} u^T \nabla_x f(x) = \min_{u,u^T,u=1} \|u\|_2 \|\nabla_x f(x)\|_2 cos\theta$$

Gradient-Based Optimization

$$x' = x - \in \nabla_x f(x)$$

$$f(x - \in \nabla_x f(x))$$

Pencarian Garis (Line Search)

Gradient-Based Optimization

Steepest descent $\nabla x f(x) = 0$ **untuk** x.

***Hill Climbing*.**

Numerical Computation
Jacobian Matrices

Ludwig Otto Hesse
Carl Gustav Jacob Jacobi

Gradien (Gradient)

$$\nabla f(x, y, z) = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}$$

Jacobian

Formulasi : jika $F = \left(f_1(x_1, x_2, \ldots, x_n), \ldots, f_m(x_1, x_2, \ldots x_n)\right)$,maka Jacobian $J(F)$ adalah matiks $m \times n$ yang elemennya adalah

$$J(\mathbf{F}) = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \frac{\partial f_m}{\partial x_2} & \cdots & \frac{\partial f_m}{\partial x_n} \end{pmatrix}$$

Conto h

Matriks Jacobian dari fungsi $f(x,y) = (f_1(x,y), f_2(x,y))$ pada titik $(1,2)$,

Diberikan :

$$f(x,y) = (f_1(x,y), f_2(x,y))$$

Dimana :

$$f_1(x,y) = x^4 + 3y^2x$$
$$f_2(x,y) = 5y^2 - 2xy + 1$$

Matriks Jacobian

Matriks Jacobian $J(F)$ adalah

$$J(F) = \begin{pmatrix} \dfrac{\partial f_1}{\partial x} & \dfrac{\partial f_1}{\partial y} \\ \dfrac{\partial f\_2}{\partial x} & \dfrac{\partial f_2}{\partial y} \end{pmatrix}$$

1.Turunan parsial dari $f_1(x, y)$:

$$\frac{\partial f_1}{\partial x} = \frac{\partial}{\partial x}\left(x^4 + 3y^2x\right) = 4x^3 + 3y^2$$

$$\frac{\partial f_1}{\partial y} = \frac{\partial}{\partial y}\left(x^4 + 3y^2x\right) = 6yx$$

2. Turunan parsial dari $f_2(x, y)$:

$$\frac{\partial f_2}{\partial x} = \frac{\partial}{\partial x}\left(5y^2 - 2xy + 1\right) = -2y$$

$$\frac{\partial f_2}{\partial y} = \frac{\partial}{\partial y}\left(5y^2 - 2xy + 1\right) = 10y - 2x$$

Matriks Jacobian pada titik $(1, 2)$:

Sekarang kita subtitusi $x = 1$ dan $y = 2$ kedalam turunan parsial diatas :

$$\left.\frac{\partial f_1}{\partial x}\right|_{1,2} = 4(1)^3 + 3(2)^2 = 4 + 12 = 16$$

$$\left.\frac{\partial f_1}{\partial x}\right|_{1,2} = 6(2)(1) = 12$$

$$\left.\frac{\partial f_2}{\partial x}\right|_{1,2} = -2(2) = -4$$

$$\left.\frac{\partial f_2}{\partial x}\right|_{1,2} = 10(2) - 2(1) = 20 - 2 = 18$$

Jadi, Matriks Jacobian pada titik $(1, 2)$ adalah :

$$\left.J(F)\right|_{1,2} = \begin{pmatrix} 16 & 12 \\ -4 & 18 \end{pmatrix}$$

Numerical Computation
Hessian Matrices

Hessian

Formulasi : jika $f = \mathbb{R}^2 \to \mathbb{R}$, maka Hessian $\mathrm{H}(f)$ adalah matiks simetris $n \times n$ yang elemennya adalah

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

Conto
h

Matriks Hessian dari fungsi $f(x,y) = y^4 + x^3 + 3x^2 + 4y^2 - 4xy - 5y + 8$ pada titik $(1,0)$,

1. Turunan Parsial Pertama :

Turunan parsial terhadap x :

$$\frac{\partial f}{\partial x} = \frac{\partial}{\partial x}\left(y^4 + x^3 + 3x^2 + 4y^2 - 4xy - 5y + 8\right) = 3x^2 + 6x - 4y$$

Turunan parsial terhadap y :

$$\frac{\partial f}{\partial y} = \frac{\partial}{\partial y}\left(y^4 + x^3 + 3x^2 + 4y^2 - 4xy - 5y + 8\right) = 4y^3 + 8y - 4x - 5$$

2. Turunan Parsial Kedua :

Turunan kedua terhadap x:

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial}{\partial x}\left(3x^2 + 6x - 4y\right) = 6x + 6$$

Turunan kedua terhadap y (crossterm):

$$\frac{\partial^2 f}{\partial y \partial x} = \frac{\partial}{\partial y}\left(3x^2 + 6x - 4y\right) = -4$$

Turunan kedua terhadap y:

$$\frac{\partial^2 f}{\partial y^2} = \frac{\partial}{\partial y}\left(4y^3 + 8y - 4x - 5\right) = 12y^2 + 8$$

Turunan kedua terhadap x (crossterm):

$$\frac{\partial^2 f}{\partial x \partial y} = \frac{\partial}{\partial x}\left(4y^3 + 8y - 4x - 5\right) = -4$$

Matriks Hessian

Matriks Hessian H(f) **adalah**

$$H(f) = \begin{pmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{pmatrix}$$

Sekarang kita subtitusi $x = 1$ dan $y = 0$ kedalam turunan parsial kedua yang telah kita hitung:

$$\left.\frac{\partial^2 f}{\partial x^2}\right|_{(1,0)} = 6(1) + 6 = 12$$

$$\left.\frac{\partial^2 f}{\partial x \partial y}\right|_{(1,0)} = -4$$

$$\left.\frac{\partial^2 f}{\partial y^2}\right|_{(1,0)} = 12(0)^2 + 8 = 8$$

4. Matriks Hessian pada titik $(1, 0)$ adalah:

$$H(f)\Big|_{1,0} = \begin{pmatrix} 12 & -4 \\ -4 & 8 \end{pmatrix}$$

Menghitung Gradient

Fungsi $f(x, y) = 5x^2 + 3xy + 3y^3$

Perlu menemukan turunan parsial terhadap variabel x **dan** y

$$\frac{\partial f}{\partial x} = 10x + 3y$$

$$\frac{\partial f}{\partial y} = 3x + 9y^2$$

Diberikan Gradien :

$$\nabla f = \begin{bmatrix} 10x & + & 3y \\ 3y & + & 9y^2 \end{bmatrix}$$

Determinan dari Jacobian matrix.

Transformasi $f\colon \mathbb{R}^2 \to \mathbb{R}$, kita menggunakan framework berikut.

$$det\left(\begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \\ c_1 & c_2 & c_3 \end{bmatrix}\right) = a_1 det\left(\begin{bmatrix} b_2 & b_3 \\ c_2 & c_3 \end{bmatrix}\right) + a_3 det\left(\begin{bmatrix} b_1 & b_2 \\ c_1 & c_2 \end{bmatrix}\right)$$

Dalam contoh ini menggunakan Jacobian matrix berikut:

$$J(f) = \begin{bmatrix} 2xy & x^2 & 0 \\ 0 & -1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

Kita dapat menghitung determinan sebagai berikut:

$$\boldsymbol{det}\left(\begin{bmatrix} 2xy & x^2 & 0 \\ 0 & -1 & 1 \\ 1 & 0 & 1 \end{bmatrix}\right)$$

$$= 2xy\, det\left(\begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix}\right) - x^2\, det\left(\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}\right) + 0 det\left(\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}\right)$$

Maka,

$$= 2xy\big(-1(1) - (1)(0)\big) - x^2\,\big(0(1) - 1(1)\big) + 0\Big(0(0) - \big(-1(1)\big)\Big)$$

$$= 2xy(-1) - x^2(-1) + 0$$

$$= -2xy + x^2$$

Apa arti dari determinan? Fungsi $-2xy + x^2$ memberikan kita amplitudo di mana ruang mengerut atau mengembang selama transformasi di sekitar suatu titik (x, y, z)

Sebagai contoh, jika kita mengevaluasinya pada titik $(1, 0, 2)$.

$$\begin{aligned} &-2(1)(0) + 1^2 \\ &= 0 + 1 \\ &= 1 \end{aligned}$$

Determinan dari Hessian matrix.

Matriks Hessian $\boldsymbol{Hf(x,y) = \nabla^2 f(x,y) = \begin{bmatrix} 10 & 3 \\ 3 & 18y \end{bmatrix}}$

$$\boldsymbol{det\left(\begin{bmatrix} a & b \\ c & d \end{bmatrix}\right) = ad - bc}$$

$$\boldsymbol{det\left(\begin{bmatrix} 10 & 3 \\ 3 & 18y \end{bmatrix}\right) = 180y - 9}$$

Ringkasan

Gradien

Jacobian

Hessian

Terima Kasih