METODE GRADIEN (BAGIAN 2)

Kuliah 7 |Optimisasi Statistika

rahmaanisa@apps.ipb.ac.id

OUTLINE | METODE GRADIEN

(Bagian 1)

Pendahuluan tentang Metode Gradien

Metode Steepest Descent

Metode Gradien Konjugat

(Bagian 2)

Metode Newton (dan berbagai modifikasinya)

Metode Quasi-Newton

Metode *Gradient Descent*

METODE NEWTON

METODE NEWTON

- Pendekatan Newton dapat digunakan untuk meminimumkan suatu fungsi peubah ganda.
- Perhatikan pendekatan menggunakan deret Taylor berikut:

$$f(\mathbf{X}) \approx f\left(\mathbf{X}_{(i)}\right) + \nabla f_{(i)}^{T}\left(\mathbf{X} - \mathbf{X}_{(i)}\right) + \frac{1}{2}\left(\mathbf{X} - \mathbf{X}_{(i)}\right)^{T}\left[\mathbf{J}_{(i)}\right]\left(\mathbf{X} - \mathbf{X}_{(i)}\right)$$

dengan $\mathbf{J}_{(i)}$ adalah matriks Hessian ($\nabla^2 f\left(\boldsymbol{X}_{(i)}\right)$). Jika $\mathbf{J}_{(i)}$ adalah definit positif, maka nilai yang meminimumkan pendekatan kuadratik di atas adalah: $X_{(i)} - J_{(i)}^{-1} \nabla f_{(i)}$, dengan

$$\boldsymbol{X}_{(\boldsymbol{i}+\mathbf{1})} = \boldsymbol{X}_{(\boldsymbol{i})} - J_{(i)}^{-1} \nabla f_{(i)} \qquad (7.1)$$

ILUSTRASI

Contoh 6.11 di Rao (2020)

Minimumkan $f(x_1, x_2) = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2$, dengan nilai awal $\boldsymbol{X}_{(1)} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$.

$$[J_{(1)}] = \begin{bmatrix} \dfrac{\partial^2 f}{\partial x_1^2} & \dfrac{\partial^2 f}{\partial x_1 \partial x_2} \\ \dfrac{\partial^2 f}{\partial x_2 \partial x_1} & \dfrac{\partial^2 f}{\partial x_2^2} \end{bmatrix}_{X_{(1)}} = \begin{bmatrix} 4 & 2 \\ 2 & 2 \end{bmatrix}$$

$$[J_{(1)}]^{-1} = \frac{1}{4}\begin{bmatrix} 2 & -2 \\ -2 & 4 \end{bmatrix} = \begin{bmatrix} 1/2 & -1/2 \\ -1/2 & 1 \end{bmatrix}$$

$$g_{(1)} = \begin{Bmatrix} \partial f/\partial x_1 \\ \partial f/\partial x_2 \end{Bmatrix}_{X_{(1)}} = \begin{Bmatrix} 1 + 4x_1 + 2x_2 \\ -1 + 2x_1 + 2x_2 \end{Bmatrix}_{(0,0)} = \begin{Bmatrix} 1 \\ -1 \end{Bmatrix}$$

$$X_{(2)} = X_{(1)} - [J_{(1)}]^{-1} g_{(1)}$$

$$= \begin{Bmatrix} 0 \\ 6 \end{Bmatrix} - \begin{bmatrix} 1/2 & -1/2 \\ -1/2 & 1 \end{bmatrix} \begin{Bmatrix} 1 \\ -1 \end{Bmatrix}$$

$$= \begin{Bmatrix} -1 \\ 3/2 \end{Bmatrix}$$

memeriksa keoptimalan pada $X_{(2)}$

$$g_{(2)} = \begin{Bmatrix} \partial f/\partial x_1 \\ \partial f/\partial x_2 \end{Bmatrix}_{X_{(2)}} = \begin{Bmatrix} 1 + 4x_2 + 2x_2 \\ -1 + 2x_1 + 2x_2 \end{Bmatrix}_{(-1,\, 3/2)}$$

$$= \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \rightarrow \text{sudah optimum}$$

$$X^* = \begin{Bmatrix} -1 \\ 3/2 \end{Bmatrix} \text{ diperoleh dalam 1 iterasi}$$

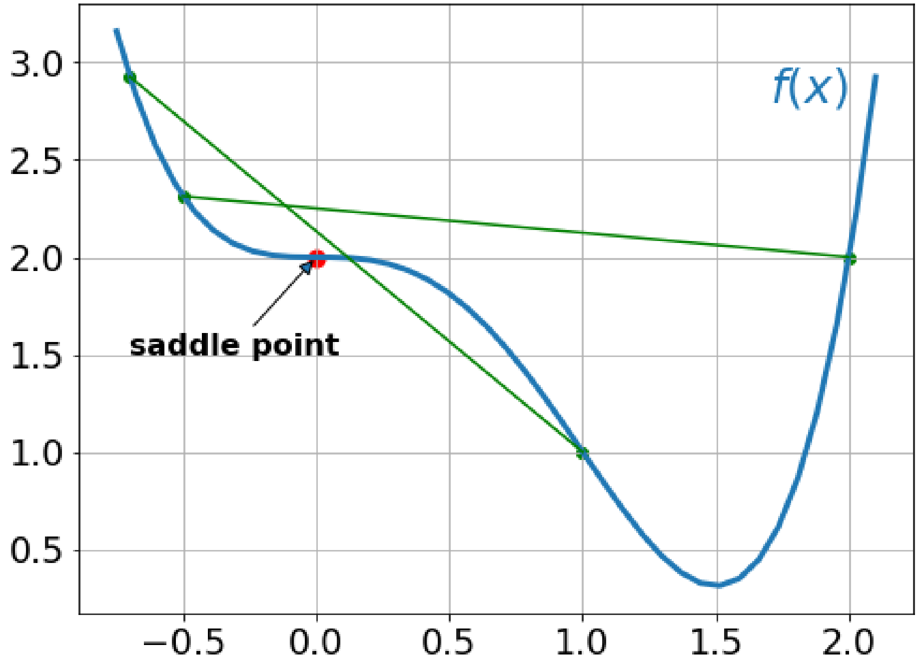
MODIFIKASI METODE NEWTON

Catatan:

jika $f(\boldsymbol{X})$ adalah suatu fungsi nonkuadratik, metode ini mungkin saja mengalami divergensi, dan mungkin pula konvergen pada suatu titik pelana dan maxima relatif. Permasalahan ini dapat diatasi dengan memodifikasi persamaan (7.1) menjadi:

$$\boldsymbol{X}_{(i+1)} = \boldsymbol{X}_{(i)} + \lambda_{(i)}^{*}\boldsymbol{S}_{(i)} = \boldsymbol{X}_{(i)} - \lambda_{(i)}^{*}\left[\boldsymbol{J}_{(i)}\right]^{-1}\nabla f_{(i)}$$

dimana $\lambda_{(i)}^{*}$ adalah panjang langkah optimum pada arah $\boldsymbol{S}_{(i)}$.



Sumber gambar: Kwiatkowski (2021)

MODIFIKASI METODE NEWTON

- Kelebihan
  - Akan mempercepat konvergensi
  - Akan selalu dapat menemukan titik minimum
  - Umumnya akan menghindari konvergensi pada titik pelana
- Kekurangan
  - Harus menyimpan matriks J yang berukuran $n \times n$
  - Terkadang sulit (bahkan mustahil) untuk menghitung setiap elemen matriks J
  - Harus menghitung matriks kebalikan dari J pada setiap iterasi
  - Harus mengevaluasi perhitungan $\left[\boldsymbol{J}_{(i)}\right]^{-1} \nabla f_{(i)}$ pada setiap iterasi

METODE MARQUADT

- Latar belakang:
  - Kelebihan metode steepest descent → mampu mengurangi nilai fungsi ketika nilai awal $\boldsymbol{X}_{(i)}$ jauh dari titik optimum $\boldsymbol{X}^*$
  - Kelebihan metode Newton → cenderung lebih cepat konvergen ketika nilai awal$\boldsymbol{X}_{(i)}$cukup dekat dari titik optimum $\boldsymbol{X}^*$
- Metode Marquadt → menggabungkan kelebihan dari kedua metode tersebut dengan cara memodifikasi elemen diagonal matriks Hessian sebagai berikut:

$$\left[\tilde{J}_{(i)}\right] = \left[J_{(i)}\right] + \alpha_{(i)}[I]$$

- Untuk nilai α yang bernilai cukup besar, nilai $\alpha_i[I]$ akan mendominasi sehingga matriks kebalikan $\left[\tilde{J}_{(i)}\right]^{-1}$ dapat dituliskan sebagai:

$$\left[\tilde{J}_{(i)}\right]^{-1} \approx \left[\alpha_{(i)}[I]\right]^{-1} = \frac{1}{\alpha_{(i)}}[I]$$

- Sehingga arah pencarian dapat dihitung sebagai:

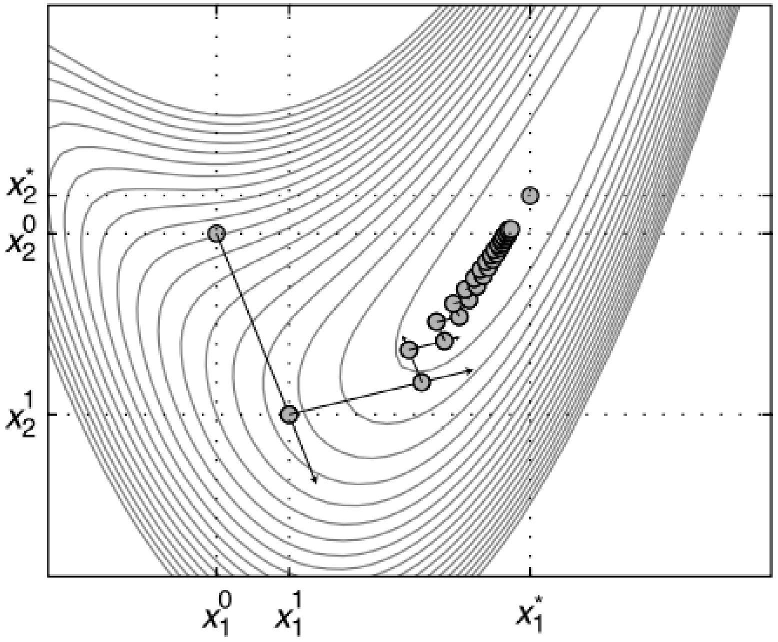
$$S_{(i)} = -\left[\tilde{J}_{(i)}\right]^{-1}\nabla f_{(i)}$$

- Jika $\alpha_{(i)}$ sangat besar maka $S_{(i)}$ akan menjadi arah steepest descent. Metode Marquadt menggunakan nilai $\alpha_{(i)}$yang besar kemudian mengecil secara bertahap sampa menjadi nol.

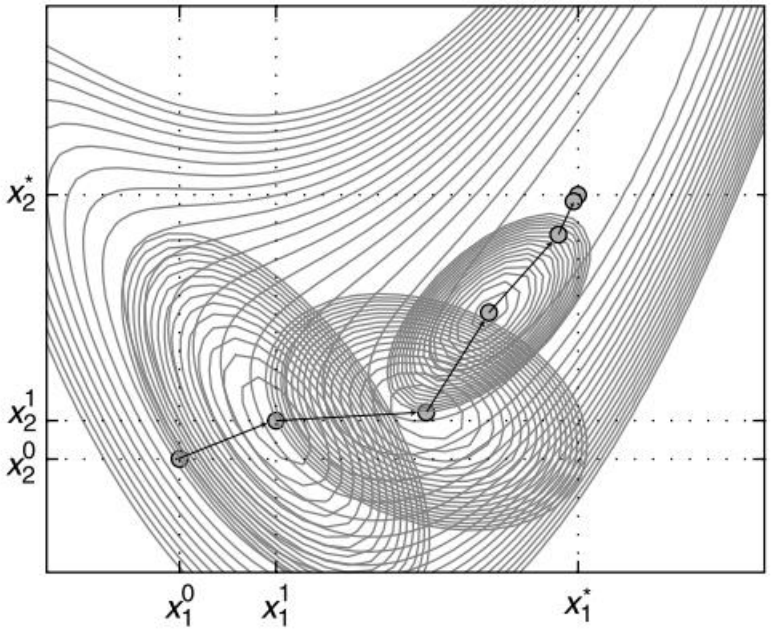
METODE QUASI-NEWTON

LATAR BELAKANG

- Metode steepest descent akan melalui jalur berpola zig-zag untuk mencapai titik optimum
- Metode Newton cenderung dapat mencapai titik optimum dengan jalur yang lebih langsung



Steepest Descent



Newton

Perbandingan kedua metode dalam mengoptimumkan fungsi yang sama. (Gili *et al.*, 2019)

LATAR BELAKANG

- Jalur pada metode Newton cenderung lebih *direct*
- Namun, metode Newton memiliki kelemahan utama, yaitu beban komputasi untuk perhitungan matriks Hessian

- Alternatifnya, kita dapat menggunakan pendekatan untuk menghindari perhitungan matriks Hessian pada setiap iterasi.

METODE QUASI-NEWTON

- Proses iterasi pada metode Newton adalah menggunakan fungsi berikut:

$$\boldsymbol{X}_{(i+1)} = \boldsymbol{X}_{(i)} - \left[\boldsymbol{J}_{(i)}\right]^{-1} \nabla f\left(\boldsymbol{X}_{(i)}\right)$$

- Metode quasi-Newton menggunakan matriks $\boldsymbol{A}_{(i)}$ sebagai aproksimasi bagi matriks $\boldsymbol{J}_{(i)}$, atau $\boldsymbol{B}_{(i)}$sebagai aproksimasi bagi matriks $\left[\boldsymbol{J}_{(i)}\right]^{-1}$, dengan hanya menggunakan turunan pertama saja.
- Sehingga persamaan yang digunakan menjadi:

$$\boldsymbol{X}_{(i+1)} = \boldsymbol{X}_{(i)} - \lambda^*_{(i)} \boldsymbol{B}_{(i)} \nabla f\left(\boldsymbol{X}_{(i)}\right) \qquad (7.2)$$

dimana $\lambda^*_{(i)}$ adalah panjang langkah optimal sepanjang arah $\boldsymbol{S}_{(i)}$ berikut:

$$\boldsymbol{S}_{(i)} = -\boldsymbol{B}_{(i)} \nabla f\left(\boldsymbol{X}_{(i)}\right)$$

- Perhatikan bahwa jika kita menggunakan $\boldsymbol{B}_{(i)} = I$, maka persamaan yang digunakan akan sama seperti metode *steepest descent*.

APPROKSIMASI MATRIKS HESSIAN

Penentuan matriks $\boldsymbol{B}_{(i)} = A_{(i)}^{-1}$ dapat dilakukan dengan cara berikut:

- Perhatikan persamaan berikut:

$$\nabla f(\boldsymbol{X}) \approx \nabla f\left(\boldsymbol{X}_{(0)}\right) + \boldsymbol{J}_{(0)}\left(\boldsymbol{X} - \boldsymbol{X}_{(0)}\right)$$

- Selanjutnya jika dipilih dua titik $\boldsymbol{X}_{(i)}$ dan $\boldsymbol{X}_{(i+1)}$ dan kita gunakan $\boldsymbol{A}_{(i)}$ sebagai hampiran bagi $\boldsymbol{J}_{(0)}$ maka persamaan di atas dapat ditulis sebagai berikut:

$$\nabla f_{(i+1)} = \nabla f\left(\boldsymbol{X}_{(0)}\right) + \boldsymbol{A}_{(i)}\left(\boldsymbol{X}_{(i+1)} - \boldsymbol{X}_{(0)}\right)$$
$$\nabla f_{(i)} = \nabla f\left(\boldsymbol{X}_{(0)}\right) + \boldsymbol{A}_{(i)}\left(\boldsymbol{X}_{(i)} - \boldsymbol{X}_{(0)}\right)$$

- Selisih dari dua persamaan di atas akan menghasilkan:

$$\boldsymbol{A}_{(i)}\boldsymbol{d}_{(i)} = \boldsymbol{g}_{(i)} \qquad (7.3)$$

  dimana $\boldsymbol{d}_{(i)} = \boldsymbol{X}_{(i+1)} - \boldsymbol{X}_{(0)}$ dan $\boldsymbol{g}_{(i)} = \nabla f_{(i+1)} - \nabla f_{(i)}$

- Persamaan (7.3) dapat pula ditulis sebagai berikut:

$$\boldsymbol{d}_{(i)} = \boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)} \qquad (7.4)$$

RANK 1 UPDATE

- Kebanyakan pendekatan dalam metode quasi-Newton adalah dengan menentukan nilai awal matriks $\boldsymbol{B}_{(i)}$, kemudian memperbaruinya dengan penjumlahan berikut:

$$\boldsymbol{B}_{(i+1)} = \boldsymbol{B}_{(i)} + \nabla\boldsymbol{B}_{(i)}$$

dimana $\nabla\boldsymbol{B}_{(i)}$ disebut sebagai matriks *update* (atau koreksi).

- *Rank 1 update* dilakukan dengan persamaan berikut:

$$\boldsymbol{B}_{(i+1)} = \boldsymbol{B}_{(i)} + \frac{\left(\boldsymbol{d}_{(i)} - \boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}\right)\left(\boldsymbol{d}_{(i)} - \boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}\right)^T}{\left(\boldsymbol{d}_{(i)} - \boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}\right)^T \boldsymbol{g}_{(i)}} \qquad (7.5)$$

dengan terlebih dulu menentukan nilai awal $\boldsymbol{B}_{(1)}$, lalu nilai $\boldsymbol{X}_{(2)}$ dihitung dengan persamaan (7.2), selanjutnya $\boldsymbol{B}_{(i)}$ dihitung dengan persamaan (7.5), iterasi ini terus dilakukan sampai konvergen. Prosedur ini menjamin bahwa matriks $\boldsymbol{B}_{(i)}$ bersifat simetrik, namun tidak dijamin definit positif, sehingga dilakukan *rank 2 update.*

RANK 2 UPDATE

- *Rank 2 update* dilakukan dengan persamaan berikut:

$$\boldsymbol{B}_{(i+1)} = \boldsymbol{B}_{(i)} + \frac{\boldsymbol{d}_{(i)}\boldsymbol{d}_{(i)}^T}{\boldsymbol{d}_{(i)}^T\boldsymbol{g}_{(i)}} - \frac{\left(\boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}\right)\left(\boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}\right)^T}{\left(\boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}\right)^T\boldsymbol{g}_{(i)}} \quad (7.7)$$

- Persamaan di atas dikenal pula sebagai formula **Davidon-Fletcher-Powell (DFP)**.

  Karena $\boldsymbol{X}_{(i+1)} = \boldsymbol{X}_{(i)} + \lambda_{(i)}^*\boldsymbol{S}_{(i)}$, dan $\boldsymbol{d}_{(i)} = \boldsymbol{X}_{(i+1)} - \boldsymbol{X}_{(i)}$ dapat ditulis sebagai $\boldsymbol{d}_{(i)} = \lambda_{(i)}^*\boldsymbol{S}_{(i)}$, maka persamaan (7.7) dapat ditulis sebagai berikut:

$$\boldsymbol{B}_{(i+1)} = \boldsymbol{B}_{(i)} + \frac{\lambda_{(i)}^*\boldsymbol{S}_{(i)}\boldsymbol{S}_{(i)}^T}{\boldsymbol{S}_{(i)}^T\boldsymbol{g}_{(i)}} - \frac{\boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}\boldsymbol{g}_{(i)}^T\boldsymbol{B}_{(i)}}{\boldsymbol{g}_{(i)}^T\boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}}$$

RANK 2 UPDATE

- Formula untuk melakukan *rank 2 update* dapat pula dituliskan menggunakan matriks $\boldsymbol{A}_{(i)}$, persamaan ini dikenal pula sebagai formula **Broydon-Fletcher-Goldfarb-Shanno (BFGS)** :

$$\boldsymbol{A}_{(i+1)} = \boldsymbol{A}_{(i)} + \frac{\boldsymbol{g}_{(i)}\boldsymbol{g}_{(i)}^T}{\boldsymbol{g}_{(i)}^T\boldsymbol{d}_{(i)}} - \frac{\left(\boldsymbol{A}_{(i)}\boldsymbol{d}_{(i)}\right)\left(\boldsymbol{A}_{(i)}\boldsymbol{d}_{(i)}\right)^T}{\left(\boldsymbol{A}_{(i)}\boldsymbol{d}_{(i)}\right)^T\boldsymbol{d}}$$

Namun, untuk kemudahan dalam komputasi, yang digunakan dalam perhitungan adalah formula dengan matriks $\boldsymbol{B}_{(i)}$ berikut ini:

$$\boldsymbol{B}_{(i+1)} = \boldsymbol{B}_{(i)} + \frac{\boldsymbol{d}_{(i)}\boldsymbol{d}_{(i)}^T}{\boldsymbol{d}_{(i)}^T\boldsymbol{g}_{(i)}}\left(1 + \frac{\boldsymbol{g}_{(i)}^T\boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}}{\boldsymbol{d}_{(i)}^T\boldsymbol{g}_{(i)}}\right) - \frac{\boldsymbol{B}_{(i)}\boldsymbol{g}_{(i)}\boldsymbol{d}_{(i)}^T}{\boldsymbol{d}_{(i)}^T\boldsymbol{g}_{(i)}} - \frac{\boldsymbol{d}_{(i)}\boldsymbol{g}_{(i)}^T\boldsymbol{B}_{(i)}}{\boldsymbol{d}_{(i)}^T\boldsymbol{g}_{(i)}}$$

ALGORITMA

Secara umum, prosedur yang dilakukan pada metode quasi-Newton adalah melalui tahap-tahap berikut:

1. Menentukan nilai awal $\boldsymbol{X}_{(1)}$ dan matriks $\boldsymbol{B}_{(1)}$, vektor gradien ∇f, dan menetapkan urutan iterasi $i = 1$
2. Menghitung vektor gradien $\nabla f_{(i)}$ pada titik $\boldsymbol{X}_{(i)}$ dan $\boldsymbol{S}_i = -\boldsymbol{B}_{(i)} \nabla f_{(i)}$
3. Menentukan panjang langkah optimal $\lambda^*_{(i)}$ dan menghitung $\boldsymbol{X}_{(i+1)} = \boldsymbol{X}_{(i)} + \lambda^*_{(i)} \boldsymbol{S}_i$
4. Memeriksa keoptimuman titik $\boldsymbol{X}_{(i+1)}$ (pada metode BFGS, jika $\left\|\nabla f_{(i+1)}\right\| \leq \varepsilon$, dimana ε bernilai sangat kecil, maka dipilih $\boldsymbol{X}^* \approx \boldsymbol{X}_{(i+1)}$). Jika sudah optimal, proses berhenti. Jika tidak, maka dilanjutkan ke langkah 5.
5. Memperbarui matriks Hessian sesuai dengan formula *rank 2 update*.
6. Menetapkan $i = i + 1$ dan kembali ke langkah 2.

ILUSTRASI

Ilustrasi perhitungan manual dapat dipelajari pada contoh 6.15 dan contoh 6.16 pada Rao (2020).

METODE *GRADIENT DESCENT*

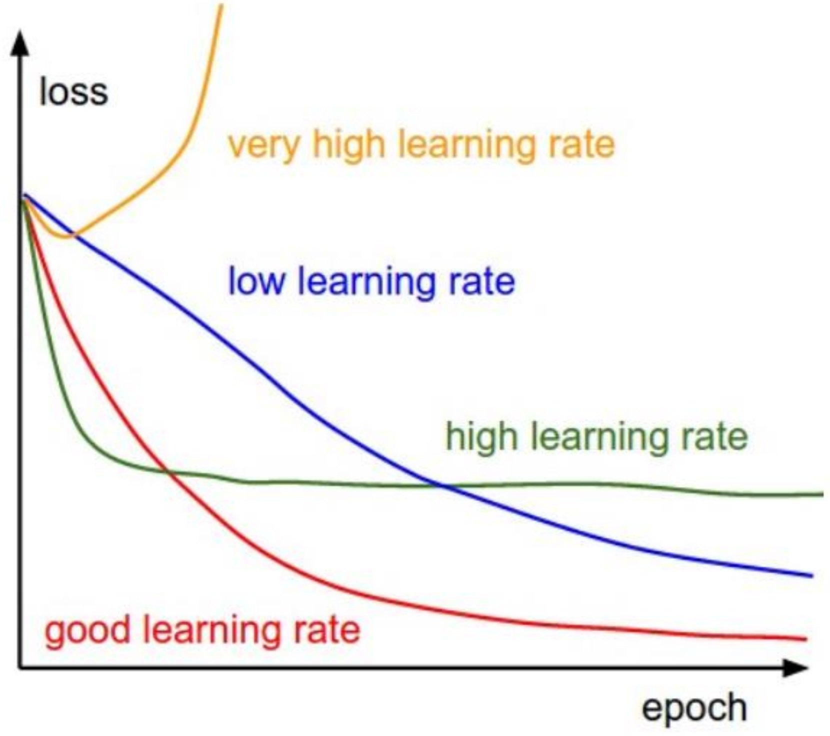
METODE *GRADIENT DESCENT*

- Optimisasi dengan metode ini dilakukan menggunakan formula berikut:

$$\boldsymbol{X}_{(i+1)} = \boldsymbol{X}_{(i)} - \alpha \nabla f_{(i)}$$

  dimana α merupakan *learning rate,* yang menentukan panjang langkah pada setiap iterasi.

- Jika α terlalu kecil maka prosedur ini akan memerlukan waktu yang lama untuk mencapai konvergensi, namun jika α terlalu besar maka proses bisa saja tidak mencapai konvergensi dan melampui titik optimum yang diinginkan.

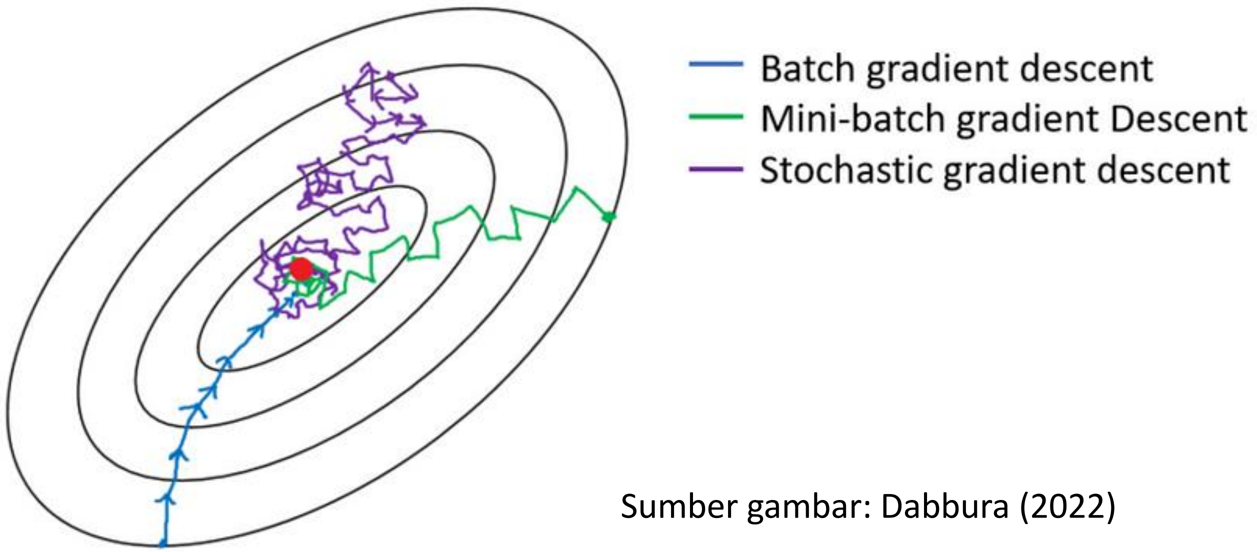


Sumber gambar: Dabbura (2022)

ALGORITMA

Terdapat tiga jenis algoritma dalam metode *gradient descent* yang umum digunakan:

1. *Batch gradient descent* : menggunakan seluruh pengamatan pada data untuk setiap iterasi
2. *Stochastic gradient descent* : menggunakan satu pengamatan
3. *Mini-batch gradient descent :* menggunakan subset dari data (sekumpulan pengamatan)



Sumber gambar: Dabbura (2022)

REFERENSI

Butenko,S., Pardalos, P.M. (2014). Numerical Methods and Optimization: An Introduction. CRC Press.

Dabbura, I. (2022, December 22). Gradient descent algorithm and its variants. Medium. https://towardsdatascience.com/gradient-descent-algorithm-and-its-variants-10f652806a3

Everitt, B.S. (1987). Introduction to Optimization Methods and Their Application in Statistics. New York: Chapman and Hall.

Gentle, J.E. (2004). Optimization Methods for Applications in Statistics.

Gilli, M., Maringer, D., & Schumann, E. (2019). Basic methods. *Numerical Methods and Optimization in Finance*, 229–271. doi:10.1016/b978-0-12-815065-8.00023-6

Skycak , J. (2021, February 4). Single-variable gradient descent. https://www.justinmath.com/single-variable-gradient-descent/

Kwiatkowski, R. (2021, May 22). Gradient descent Algorithm — a deep dive. Medium. https://towardsdatascience.com/gradient-descent-algorithm-a-deep-dive-cf04e8115f21

Lam, A. (2020, November 26). *BFGS in a nutshell: An introduction to Quasi-Newton methods*. Medium. https://towardsdatascience.com/bfgs-in-a-nutshell-an-introduction-to-quasi-newton-methods-21b0e13ee504

Rao, S.S. (2020). Engineering Optimization. New Jersey: John Wiley & Sons.