

Interpolasi Invers dan Peran Monotonik

1. Apa arti monotonik dalam hal ini?

Dalam konteks interpolasi invers, **monotonik** berarti fungsi yang menghubungkan nilai x dan y bersifat *selalu naik* (monoton naik) atau *selalu turun* (monoton turun) pada interval yang ditinjau.

- **Monoton Naik:** Jika $x_1 < x_2$, maka $f(x_1) \leq f(x_2)$ (atau secara ketat: $f(x_1) < f(x_2)$).
- **Monoton Turun:** Jika $x_1 < x_2$, maka $f(x_1) \geq f(x_2)$ (atau secara ketat: $f(x_1) > f(x_2)$).

Artinya, tidak ada “naik-turun” atau fluktuasi — nilai fungsi hanya bergerak satu arah seiring bertambahnya input.

2. Mengapa monotonik menjadi masalah dalam interpolasi invers?

Interpolasi invers adalah proses mencari nilai x yang sesuai dengan nilai y tertentu, berdasarkan pasangan data (x_i, y_i) . Masalah muncul jika fungsi **tidak monotonik**, karena:

a. Tidak Unik (Non-Unique Solution)

Jika fungsi tidak monotonik, maka satu nilai y dapat berkorespondensi dengan lebih dari satu nilai x .

Contoh: Misalkan $y = x^2$ pada interval $[-2, 2]$. Untuk $y = 4$, terdapat dua solusi: $x = -2$ dan $x = 2$.

⇒ Interpolasi invers tidak tahu solusi mana yang harus dipilih.

b. Tidak Invertibel (Not Invertible)

Fungsi yang tidak monotonik **tidak memiliki invers yang merupakan fungsi**, karena melanggar definisi fungsi (satu input → satu output). Interpolasi invers mengasumsikan bahwa hubungan antara x dan y dapat dibalik, yang hanya dijamin jika fungsi monotonik.

c. Kesalahan Interpolasi

Algoritma interpolasi (misalnya Lagrange atau spline) dapat memberikan hasil yang salah atau tidak stabil jika diterapkan pada data non-monotonik, karena tidak jelas arah pencarian nilai x untuk y tertentu.

Kesimpulan

- ✓ Interpolasi invers bekerja dengan baik hanya jika fungsi **monotonik**, karena:
 - Menjamin keunikan solusi.
 - Memastikan fungsi invertibel.
 - Mencegah ambiguitas dan kesalahan numerik.
- ✗ Jika fungsi tidak monotonik, pertimbangkan:
 - Membagi interval menjadi bagian-bagian yang monotonik.
 - Menggunakan metode alternatif seperti pencarian akar (*root finding*) dengan batasan tambahan.