

MENENTUKAN PREDIKSI GLBB

0.1 Studi kasus dataset Gaya GLBB

Dalam kode program Python yang diberikan, terdapat sebuah fungsi bernama ‘PosisiGLB(t)’ yang menghitung posisi (x) dan kecepatan (v) suatu objek pada Gerak Lurus Beraturan (GLB) pada waktu tertentu (t). GLB adalah gerakan dengan percepatan konstan, di mana posisi dan kecepatan dapat dihitung berdasarkan rumus-rumus fisika tertentu.

Pada fungsi ‘PosisiGLB(t)’, terdapat rumus untuk menghitung posisi (x) pada GLB yang dinyatakan sebagai:

$$x(t) = x_0 + v_0 \cdot t + \frac{1}{2} \cdot a \cdot t^2$$

di mana $x(t)$ adalah posisi pada waktu t , x_0 adalah posisi awal, v_0 adalah kecepatan awal, a adalah percepatan yang dalam kasus ini dijadikan konstan, dan t adalah waktu.

Selain itu, terdapat rumus untuk menghitung kecepatan (v) pada GLB yang dinyatakan sebagai:

$$v(t) = v_0 + a \cdot t$$

di mana $v(t)$ adalah kecepatan pada waktu t , v_0 adalah kecepatan awal, a adalah percepatan, dan t adalah waktu.

Program ini melakukan iterasi dari $t = 0$ hingga $t = 2.4$ dengan selang waktu sebesar 0.1 dan memanggil fungsi ‘PosisiGLB(t)’ untuk setiap nilai waktu yang dihasilkan. Hasil perhitungan posisi (x) dan kecepatan (v) pada GLB untuk setiap t kemudian dicetak dalam format yang sesuai dan disimpan ke dalam file ”data_{GL}.txt”.

Dalam kode tersebut, terdapat bagian yang dijadikan komentar menggunakan tanda ”” yang jika diaktifkan akan menghasilkan visualisasi grafik posisi dan kecepatan terhadap waktu. Grafik tersebut menunjukkan perubahan posisi dan kecepatan suatu objek dalam GLB seiring berjalannya waktu.

0.2 Studi kasus prediksi persamaan dengan menggunakan Gaya GLBB

Studi kasus yang menjelaskan kode program Python yang diberikan:

Sebuah studi dilakukan terkait pergerakan suatu objek yang direkam dalam suatu database. Data ini mencakup waktu (t), posisi (*Posisi*), dan kecepatan (*Kecepatan*). Tujuan dari studi ini

adalah untuk memprediksi posisi dan kecepatan objek pada waktu-waktu tertentu menggunakan metode Support Vector Regression (SVR).

Pertama-tama, data dibaca dari file yang berisi informasi tersebut ke dalam program Python menggunakan pustaka seperti NumPy untuk komputasi numerik, Pandas untuk manajemen data, dan SKlearn untuk model SVR. Data kemudian dibersihkan, terutama dari spasi yang tidak diinginkan pada nama kolom.

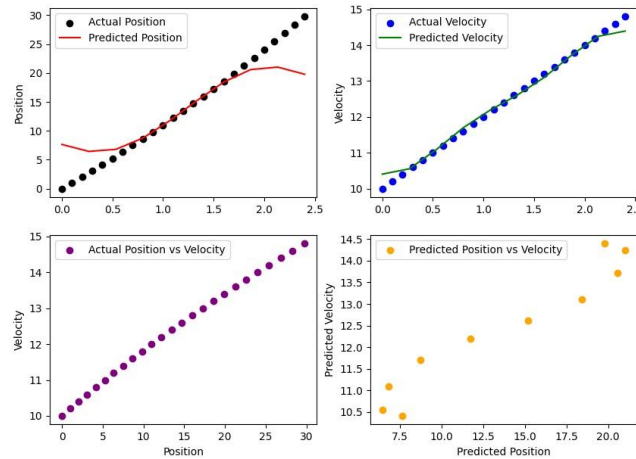
Setelah memeriksa keberadaan kolom 't', 'Posisi', dan 'Kecepatan', data dipisahkan menjadi atribut independen (X) yaitu waktu (t) dan atribut dependen (y) yaitu posisi ($Posisi$) dan kecepatan ($Kecepatan$). Dua model SVR dibuat terpisah untuk memprediksi posisi dan kecepatan.

Prediksi untuk posisi dan kecepatan dilakukan menggunakan model yang telah dibuat dengan menggunakan nilai waktu tertentu. Prediksi ini kemudian divisualisasikan dalam empat grafik yang berbeda: posisi terhadap waktu, kecepatan terhadap waktu, posisi terhadap kecepatan, dan prediksi posisi terhadap prediksi kecepatan.

Grafik-grafik tersebut memberikan pemahaman visual terhadap bagaimana model SVR memprediksi posisi dan kecepatan pada berbagai nilai waktu. Prediksi tersebut juga ditampilkan dalam bentuk tabel yang memuat nilai prediksi posisi dan kecepatan pada waktu-waktu tertentu.

Proses ini memberikan pemahaman yang lebih baik tentang hubungan antara waktu, posisi, dan kecepatan objek yang direkam dalam dataset tersebut. Dengan menggunakan metode SVR, kita dapat membuat prediksi yang cukup akurat tentang posisi dan kecepatan objek pada waktu-waktu tertentu berdasarkan data yang ada.

Namun, penting untuk dicatat bahwa hasil prediksi ini dapat dipengaruhi oleh berbagai faktor seperti jumlah dan kualitas data yang tersedia, serta pemilihan model yang digunakan dalam proses prediksi. Adjustments atau fine-tuning pada parameter model SVR juga dapat meningkatkan akurasi prediksi.



Gambar 0.1: Hasil Pengujian Program GLB

0.3 Diagram Alir

Adapun diagram alir penelitian ini adalah sebagai berikut.

