PH504M Lab 5: Interpolation

Vikram Khaire

07 February 2025

Quadratic Interpolation of Projectile Height

A projectile is launched, and its height is recorded at different horizontal distances. The given data is:

- **Horizontal distance array**:

  `distance` = $[0.0, 113.14, 226.27, 339.41, 452.55, 565.69, 678.82, 791.96, 905.1, 1018.23, 1131.37]$

- **Height array**:

  `height` = $[0.0, 100.59, 176.1, 226.52, 251.84, 252.09, 227.24, 177.3, 102.28, 2.17, 0.0]$

- **Interpolation points**: Use quadratic interpolation to estimate the height for the following distances:

  `interp_dist` = `numpy.arange(50, 1200, 50)`

  Store the interpolated heights in an array named `interp_height`.

Plotting Instructions

1. Create a scatter plot of the given **distance vs. height** data.
   - Use the blue color for original data points.
2. On the same plot, overplot a scatter plot of **interpolated distances vs. interpolated heights**.
   - Use the red color for interpolated data points.
3. Properly label axis and data points.

Bonus Task (Extra Points)

Using the solution from **Lab 2, Question 2**, generate the **same projectile trajectory** assuming:

- **Initial velocity**: 100 m/s
- **Launch angle** (θ): 45°

Plot this trajectory on the same graph using a **dashed black line**.

Hint

To find the three nearest points for interpolation, you can use the following code snippet:

```
import numpy as np

# Example data (x-values) use the distance array given above

# Choose a point where you want to interpolate
distance_you_want_to_interpolate = 150

# Find indices of the 3 nearest points
idx = np.argsort(np.abs(distance_array - distance_you_want_to_interpolate))[:3]

# For testing you can print the nearest distances
print("Nearest distances:", distance_array[idx])
```

Explanation:

- `np.abs(distance_array - distance_you_want_to_interpolate)` finds how far each value in the array is from the given distance.
- `np.argsort(...)` sorts these distances in increasing order.
- `[:3]` selects the indices of the three smallest values (i.e., the nearest three points).

When you run this code, it will return the three closest distances to 150 from the given list.