Day - 33

1. Introduction

Verified whether a heavy object dropped from a significant height will accurately hit a target on the ground. The initial scenario involves dropping a 60 kg weight from a height of 100 meters onto a 10-meter-diameter target. To safely replicate this setup in a home environment, specifically within a two-floor apartment, I scaled down all relevant parameters. Additionally, I developed a Python-based simulation to analyze the influence of wind on the object's trajectory. My analysis demonstrates that while the object will reliably hit the target under ideal (windless) conditions, even a light wind can cause the object to miss.

2. Original Setup vs. Scaled Setup

2.1 Original Setup Parameters

Parameter	Value
Drop Height	100 meters
Object Weight	60kg
Target Diameter	10 meters
Allowed Horizontal Drift	±5 meters

2.2 Scaled Setup for Home Simulation

For the home simulation, we scaled the experiment to a two-floor apartment, which typically has a height of approximately 6 meters.

Scaling Factor: 6/100 = 0.06

Parameter	Value
Drop Height	6 meters
Object Weight	~13-15 grams
Target Diameter	0.6 meters (60 cm)
Allowed Horizontal Drift	±0.3 meters

2.3 Practical Home Simulation Steps

1. Selected a Drop Height:
- Used a balcony in a two-floor apartment (~6 meters).

2. Selected a Scaled Object:
- Used a small ball weighing around 13-15 grams (a rubber ball)

3. Set Up Target:

- Placed a 60 cm diameter target on the ground directly beneath the drop point.

4. Drop Under Controlled Conditions:
- Ensure no initial horizontal push.
- Drop the ball from rest directly above the center of the target.

Under these conditions, the object reliably lands within the target area.

3. Python Simulation Design

3.1 Simulation Parameters

- Drop Heights: 100 meters (original) and 6 meters (scaled)
- Target Radii: 5 meters (original) and 0.3 meters (scaled)
- Wind Speed: Variable, tested up to 3 mph (~1.34 m/s)
- Number of Drops: 50 simulated trials

3.2 Simulation Process

1. Calculate Fall Time: t = sqrt(2h/g)

2. Compute Horizontal Drift: x = v_wind * t

3. Determine Hit or Miss:
- If |x| <= target radius, the drop is successful.
- Otherwise, it is a miss.

4. Introduce Random Variation:
- To simulate real-world variability, we added small random deviations to the horizontal displacement.

3.3 Key Python Code Snippet

```
horizontal_displacements = wind_speed * fall_time + np.random.normal(0, 0.2, num_drops)

for displacement in horizontal_displacements:
    if abs(displacement) <= target_radius:
        plt.plot(displacement, 0, 'go')  # Hit
    else:
        plt.plot(displacement, 0, 'rx')  # Miss
```

3.4 Simulation Results

The simulation showed that:
- Under no wind, the object consistently hit the target in both the original and scaled setups.
- Under wind speeds as low as 3 mph (1.34 m/s), the object frequently missed the target in both scenarios.
- The maximum allowable wind speed to consistently hit the target in the original setup was approximately 1.11 m/s (~2.5 mph).

For the scaled setup, the tolerance reduced further due to the shorter fall time and smaller target radius, with the threshold being approximately 0.27 m/s (~0.6 mph).

4. Conclusion

Through both physical scaling and Python-based simulation, we have validated that:

- Under ideal conditions (no wind), the dropped object will reliably hit the target whether in the original 100-meter setup or the scaled home simulation.
- Even minimal wind speeds (<3 mph) can cause the object to drift outside the target area, resulting in a miss.
- In the scaled apartment simulation, wind tolerance is even lower, making controlled indoor conditions essential for accurate replication.

This experiment demonstrates the critical role of lateral forces in vertical drop scenarios and highlights how simple home-based setups, when properly scaled, can effectively simulate and study real-world physics problems.