CoGrammar

Welcome to this session:

Skills Bootcamp - Simulations and Stochastic Processes (Theory and Case Studies)

The session will start shortly...

Questions? Drop them in the chat.
We'll have dedicated moderators
answering questions.

Skills Bootcamp Data Science Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly. **(Fundamental British Values: Mutual Respect and Tolerance)**
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. We will be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Academic Sessions. You can submit these questions here: **Questions**

CoGrammar

Skills Bootcamp Data Science Housekeeping

- For all **non-academic questions**, please submit a query: ***www.hyperiondev.com/support***

- **Report a safeguarding incident: *www.hyperiondev.com/safeguardreporting***

- We would love your feedback on lectures: ***Feedback on Lectures.***

- Find all the lecture **content** in your **Lecture Backpack** on GitHub.

- If you are hearing impaired, kindly use your computer's function through Google chrome to enable captions.

CoGrammar

Safeguarding & Welfare

We are committed to all our students and staff feeling safe and happy; we want to make sure there is always someone you can turn to if you are worried about anything.

If you are feeling upset or unsafe, are worried about a friend, student or family member, or you feel like something isn't right, speak to our safeguarding team:

Ian Wyles
Designated Safeguarding Lead

Simone Botes

Nurhaan Snyman

Rafiq Manan

Ronald Munodawafa

Tevin Pitts

Scan to report a safeguarding concern

or email the Designated Safeguarding Lead:
Ian Wyles
safeguarding@hyperiondev.com

CoGrammar HyperionDev

Skills Bootcamp Progression Overview

✅ Criterion 1 - Initial Requirements

Specific achievements within the first two weeks of the program.

To meet this criterion, students need to, by no later than **01 December 2024 (C11)** or **22 December 2024 (C12):**

- **Guided Learning Hours** (GLH): Attend a minimum of 7-8 GLH per week (lectures, workshops, or mentor calls) for a total minimum of **15 GLH**.
- **Task Completion:** Successfully complete the **first 4 of the assigned tasks**.

✅ Criterion 2 - Mid-Course Progress

Progress through the successful completion of tasks within the first half of the program.

To meet this criterion, students should, by no later than **12 January 2025 (C11)** or **02 February 2025 (C12):**

- **Guided Learning Hours** (GLH): Complete at least **60 GLH**.
- **Task Completion :** Successfully complete the **first 13 of the assigned tasks**.

Skills Bootcamp Progression Overview

✅ Criterion 3 – End-Course Progress

Showcasing students' progress nearing the completion of the course.

To meet this criterion, students should:

- **Guided Learning Hours** (GLH)**:** Complete the **total minimum required GLH,** by the **support end date.**
- **Task Completion : Complete all mandatory tasks**, including any necessary resubmissions, by the end of the bootcamp, **09 March 2025 (C11)** or **30 March 2025 (C12)**.

✅ Criterion 4 - Employability

Demonstrating progress to find employment.

To meet this criterion, students should:

- **Record an Interview Invite:** Students are required to record proof of invitation to an interview by **30 March 2025 (C11)** or **04 May 2025 (C12)**.
  - **South Holland Students** are required to proof and interview by **17 March 2025.**
- **Record a Final Job Outcome :** Within 12 weeks post-graduation, students are required to record a job outcome.

CoGrammar

Learning Outcomes

- **Define stochastic processes** and **explain their role** in data science and simulations.
- **Identify different types** of stochastic processes (Markov chains, Poisson processes, Brownian motion).
- **Understand Monte Carlo simulations** and their applications in real-world problems.
- **Analyse case studies** where stochastic processes are applied in fields such as finance, epidemiology, and physics.
- **Implement basic simulations** in Python using NumPy and SciPy.

Lecture Overview

- Fundamentals of Stochastic Processes
- Monte Carlo Simulations
- Case Studies and Real World Applications

Which of the following is an example of a stochastic process?

A. Sorting a list of numbers
B. Flipping a coin repeatedly
C. Printing “Hello World” in Python
D. Running a deterministic algorithm

Which of the following is an example of a stochastic process?

A. Sorting a list of numbers

B. Flipping a coin repeatedly

C. Printing “Hello World” in Python

D. Running a deterministic algorithm

What is Monte Carlo simulation commonly used for?

A. Exact computation of probabilities

B. Encrypting data

C. Random sampling to estimate probabilities

D. None of the above

What is Monte Carlo simulation commonly used for?

A. Exact computation of probabilities

B. Encrypting data

C. Random sampling to estimate probabilities

D. None of the above

What is a key assumption of a Markov Process?

A. The future state depends only on the present state
B. The future state depends on all previous states
C. The process is completely unpredictable
D. The process must be deterministic

What is a key assumption of a Markov Process?

A. **The future state depends only on the present state**
B. The future state depends on all previous states
C. The process is completely unpredictable
D. The process must be deterministic

Uncertainty in Modelling

Many real-world systems involve randomness, uncertainty, or complex interactions, whether it's predicting stock market trends, modelling disease spread, or optimizing logistics.

➢ How do we **mathematically model uncertainty**?

Uncertainty in Modelling

Stochastic processes allow us to simulate and predict uncertain events based on probabilistic rules. From finance to epidemiology, simulations help us make informed decisions by running experiments in silico before implementing them in real life.

Uncertainty in Modelling

<u>If you roll a die 1000 times, can you predict the next roll?</u>

- ➢ What if we model a **biased die**?
- ➢ What if we simulate a **random walk** with that die?
- ➢ How does this relate to **stock markets**, **traffic congestion**, or **AI**?

Stochastic Processes

A stochastic process is a mathematical model of a system that evolves over time with randomness.

KEY TERMS	
Random Variable	A variable whose possible values are outcomes of a random phenomenon.
State Space	The set of all possible states a process can be in.
Transition Probability	The likelihood of moving from one state to another.
Stationarity	When the statistical properties (e.g. mean, variance) of a process do not change over time.

CoGrammar

❖ **Deterministic systems** have **predictable outputs**, while **stochastic systems** incorporate **probability distributions**.

Stochastic Systems	Deterministic Systems
Stock market fluctuations	Sorting a list
Weather predictions	Computing the Fibonacci sequence
Epidemic modeling	Planetary Orbits

Stochastic Processes

- **Markov Property (Memoryless Property):** A stochastic process has the Markov property if the future state only depends on the present state, not past states.
  - For example: **Weather prediction** (Today's weather determines tomorrow's).
  - But not: **Stock market** (Past trends influence future predictions).
- **Markov Chains** model sequential decision-making problems.

Markov Chains

```python
import numpy as np

states = ["Sunny", "Rainy"]
transition_matrix = np.array([[0.8, 0.2], [0.4, 0.6]])

def simulate_markov_chain(start_state=0, steps=10):
    state = start_state
    for _ in range(steps):
        print(states[state])
        state = np.random.choice([0, 1], p=transition_matrix[state])

simulate_markov_chain()
```

The transition matrix dictates the probability of a future state given the present state.

Stochastic Processes

- **Poisson Process:** A stochastic process modelling the occurrence of **random events over time**, assuming:
  - Events happen independently.
  - The probability of an event in a small time window is proportional to the length of the window.
  - No two events happen at exactly the same time.
- **Examples:** Customer arrivals at a bank, Network traffic, Earthquake occurrences.

Poisson Processes

```python
import numpy as np

# Simulate inter-arrival times (exponentially distributed)
lambda_requests = 5  # 5 requests per second
n_requests = 100
inter_arrival_times = np.random.exponential(1/lambda_requests, n_requests)

# Compute arrival times
arrival_times = np.cumsum(inter_arrival_times)
print("Arrival times:", arrival_times)
```

Poisson processes model event arrivals in real-world systems like customer queues, server logs, and traffic flow.

Stochastic Processes

- **Brownian Motion:** A stochastic process that describes the **random movement of particles** suspended in a **fluid** (liquid or gas) due to **collisions with molecules** in the medium.
- Mathematically, it is a **continuous-time stochastic process** with the following properties:
  - **Starts at zero**: $B(0) = 0$.
  - **Independent increments**: Changes do not depend on past.
  - **Normally distributed increments**: The change over any time interval follows a normal distribution with mean 0 and variance proportional to the length of the interval.
  - **Continuous paths**: The motion does not have sudden jumps.

Stochastic Processes

```python
import numpy as np
import matplotlib.pyplot as plt

T = 1.0
N = 1000
dt = T/N
mu, sigma = 0.1, 0.2
S0 = 100
W = np.random.randn(N) * np.sqrt(dt)
S = S0 * np.exp(np.cumsum((mu - 0.5 * sigma**2) * dt + sigma * W))

plt.plot(S)
plt.title("Stock Price Simulation (Brownian Motion)")
plt.show()
```

Let's Breathe!

Let's take a small break before moving on to the next topic.

CoGrammar

Monte Carlo Simulations

Monte Carlo Simulation uses random sampling to estimate numerical results for problems that are difficult to solve analytically.

- The method is used in scenarios where traditional analytical solutions are difficult or infeasible. It is particularly powerful when dealing with problems involving uncertainty, randomness, and high-dimensional spaces.
- The core idea is to **use randomness** to solve deterministic problems by **approximating** their outcomes.

Monte Carlo Simulations

- **Applications:**
  - Finance (risk assessment, option pricing)
  - Physics (particle interactions, thermodynamics)
  - Artificial Intelligence (reinforcement learning, game simulations)
  - Engineering (structural reliability analysis, fluid dynamics)
  - Climate Science (weather prediction, ice sheet modeling)

Approximating π

```python
import numpy as np

def monte_carlo_pi(n_samples=10000):
    x = np.random.rand(n_samples)
    y = np.random.rand(n_samples)
    inside_circle = (x**2 + y**2) <= 1
    return (inside_circle.sum() / n_samples) * 4

print("Estimated π:", monte_carlo_pi())
```

Here, we're randomly sampling values between 0 and 1 and determining whether it would fall within the circle.

Case Studies

- **Epidemiology:** These tools were extremely valuable during COVID. The research paper "Social Stress Drives the Multi-Wave Dynamics of COVID-19 Outbreaks" introduces the **SIR-social stress (SIR_SS) model**, which integrates social behavior dynamics into traditional epidemic modeling. The model accounts for how public awareness, compliance with restrictions, and subsequent fatigue influence the spread of COVID-19. The findings suggest that incorporating social stress factors can explain the multi-wave patterns observed during the pandemic.

Case Studies

- **Finance:** Polynomial jump-diffusion models extend traditional polynomial processes by incorporating jumps. These models are particularly useful in finance for capturing sudden market movements and providing a more accurate representation of asset price dynamics. The authors discuss the mathematical properties of these models and their applications in option pricing and risk management.

Which stochastic process is widely used for financial modelling?

A. Brownian Motion
B. Linear Regression
C. Neural Networks
D. Sorting Algorithms

Which stochastic process is widely used for financial modelling?

A. **Brownian Motion**
B. Linear Regression
C. Neural Networks
D. Sorting Algorithms

CoGrammar

In an epidemic model (e.g., SIR model), what makes it a stochastic simulation?

A. It has fixed, predetermined outcomes

B. It uses random variables to simulate disease spread

C. It follows a deterministic set of equations

D. The infection rates do not change over time

In an epidemic model (e.g., SIR model), what makes it a stochastic simulation?

A. It has fixed, predetermined outcomes

B. It uses random variables to simulate disease spread

C. It follows a deterministic set of equations

D. The infection rates do not change over time

What is the key idea behind Monte Carlo methods?

A. It follows a deterministic set of equations
B. The infection rates do not change over time
C. It has fixed, predetermined outcomes
D. It uses random variables to simulate disease spread

What is the key idea behind Monte Carlo methods?

A. It follows a deterministic set of equations

B. The infection rates do not change over time

C. It has fixed, predetermined outcomes

D. It uses random variables to simulate disease spread

Summary

★ Stochastic processes model uncertainty—from finance to epidemiology.

★ Monte Carlo simulations are used in risk assessment, physics, and AI.

★ Markov Chains, Poisson Processes, and Brownian Motion are key stochastic models.

★ Python tools like NumPy, SciPy, and Matplotlib allow simulation of real-world systems.

CoGrammar

Q & A SECTION

Please use this time to ask any questions relating to the topic, should you have any.

Thank you for attending

CoGrammar

SKILLS FOR LIFE
SKILLS BOOTCAMPS

Department for Education