



# First meeting with the Artificial Intelligence Sustainability Lab at HNEE (KIN'L)

---

Jorge Ruiz



**Eberswalde University  
for Sustainable  
Development**

# What's the KIN'L?

---

Artificial Intelligence Sustainability Lab at Eberswalde University for Sustainable Development (KI-Nachhaltigkeitslabor, KIN'L@HNEE). The Federal Ministry of Education and Research (BMBF) is funding the establishment of an AI sustainability laboratory at HNEE as part of the KI-Nachwuchs@FH funding guideline. The endeavour is centered on the acquisition of an AI computing system for use in teaching, research and transfer.

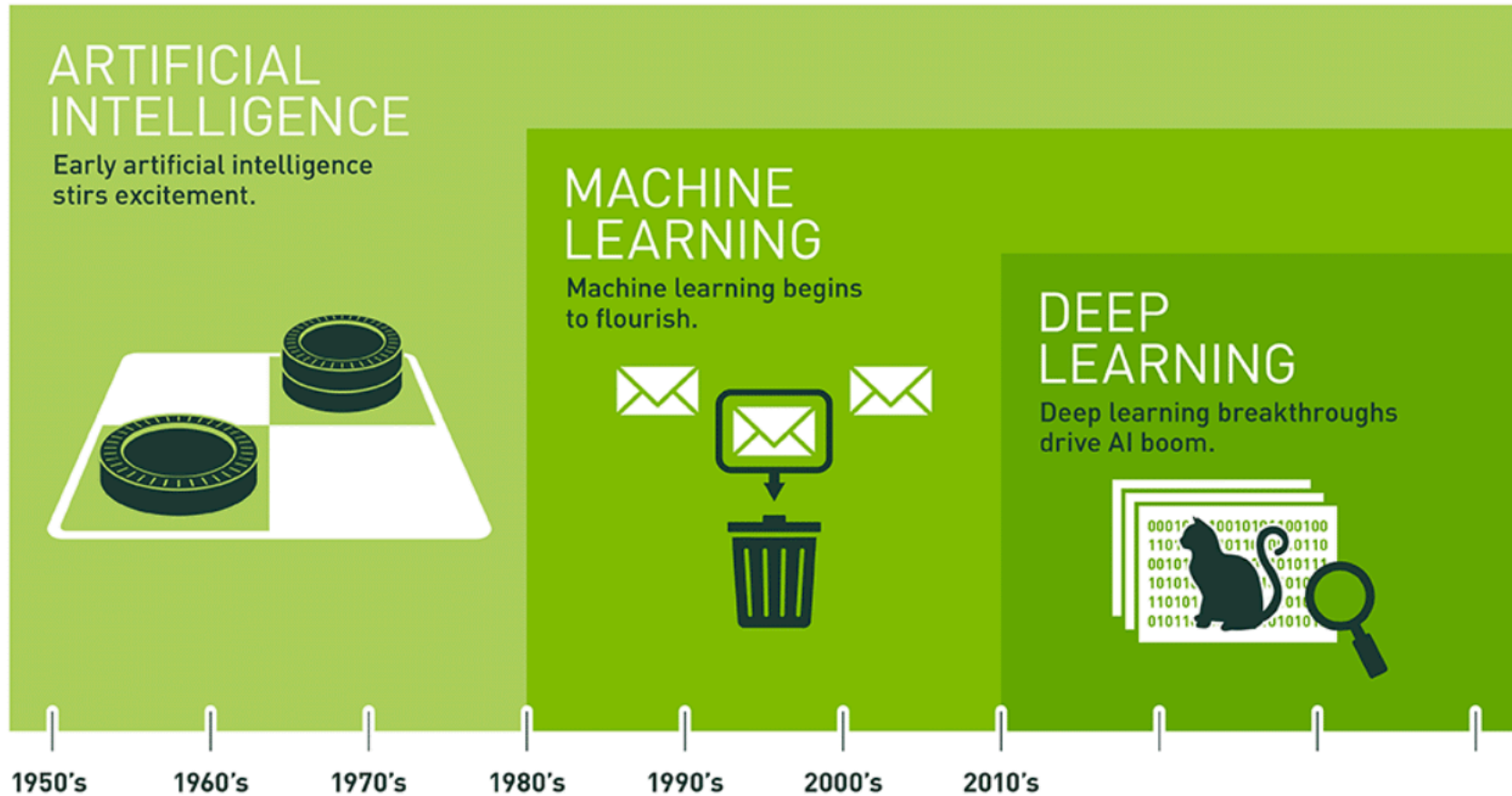
- April 1st 2023 to April 30th 2025
- Funding body: BMBF
- Contact person: Peter Neumeister

GEFÖRDERT VOM



 Bundesministerium  
für Bildung  
und Forschung

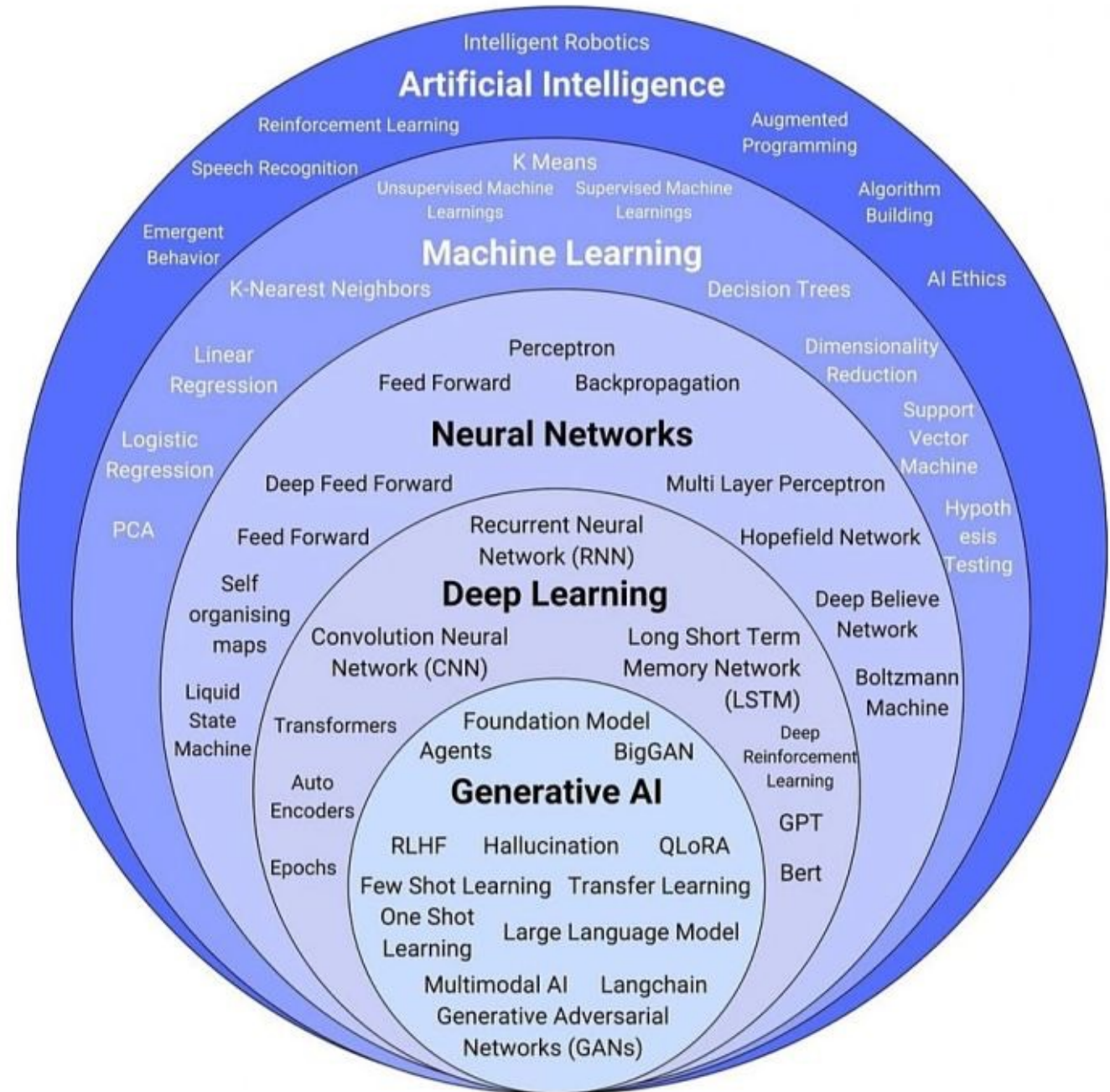
# Artificial intelligence (AI), machine learning (ML), deep learning (DL)



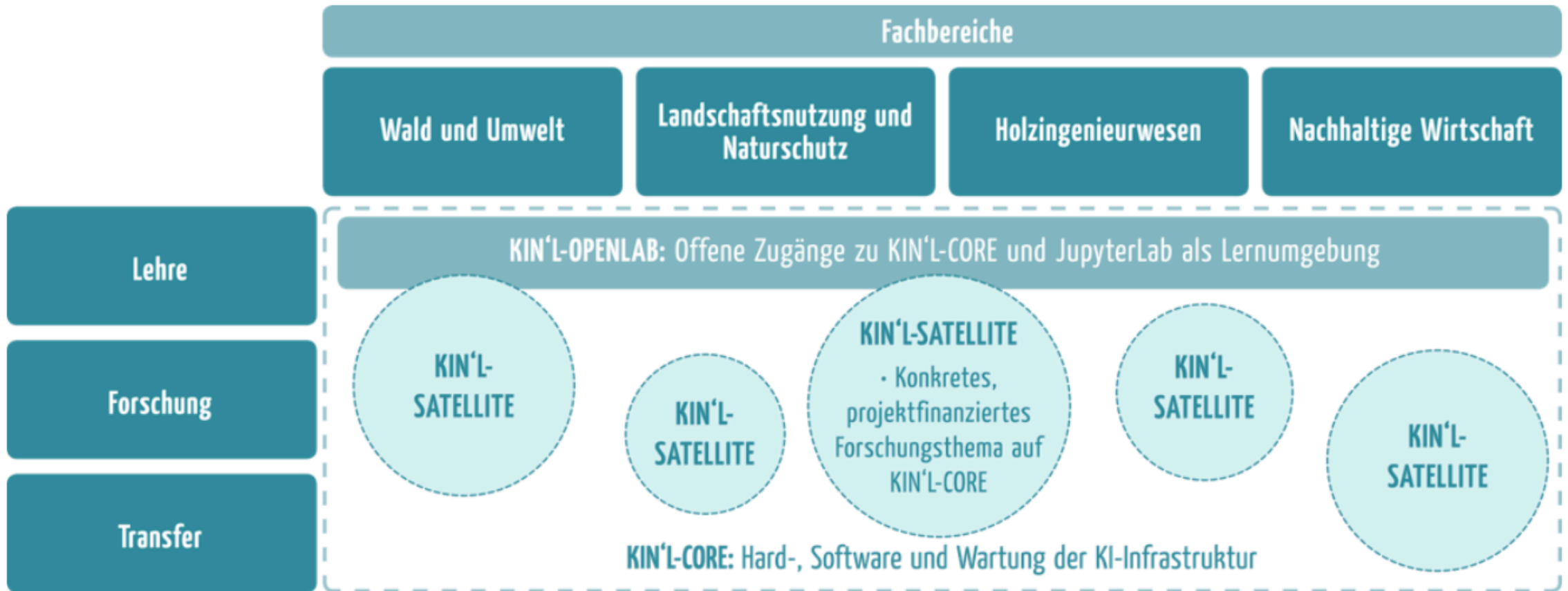
Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

## AI, ML, DL

---



# KIN'L into the HNEE



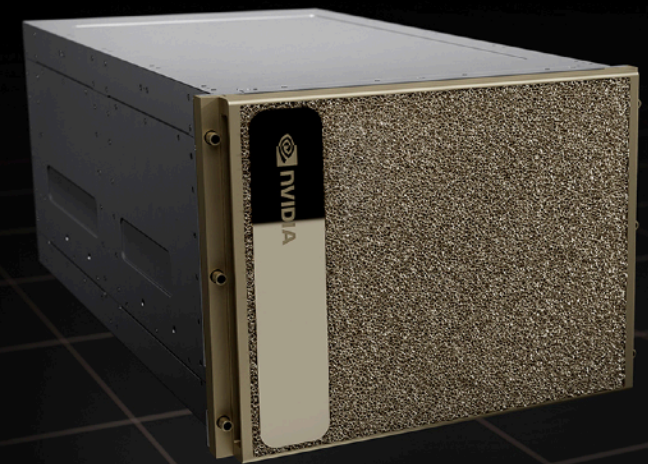
# KIN'L-CORE Hardware: NVIDIA GPUs

---

- 8x NVIDIA H100 Tensor Core GPUs
- 640GB total GPU memory
- 2TB RAM system memory
- Networking up to 400Gb/s InfiniBand/Ethernet
- Storage with 500TB NVMe SSDs

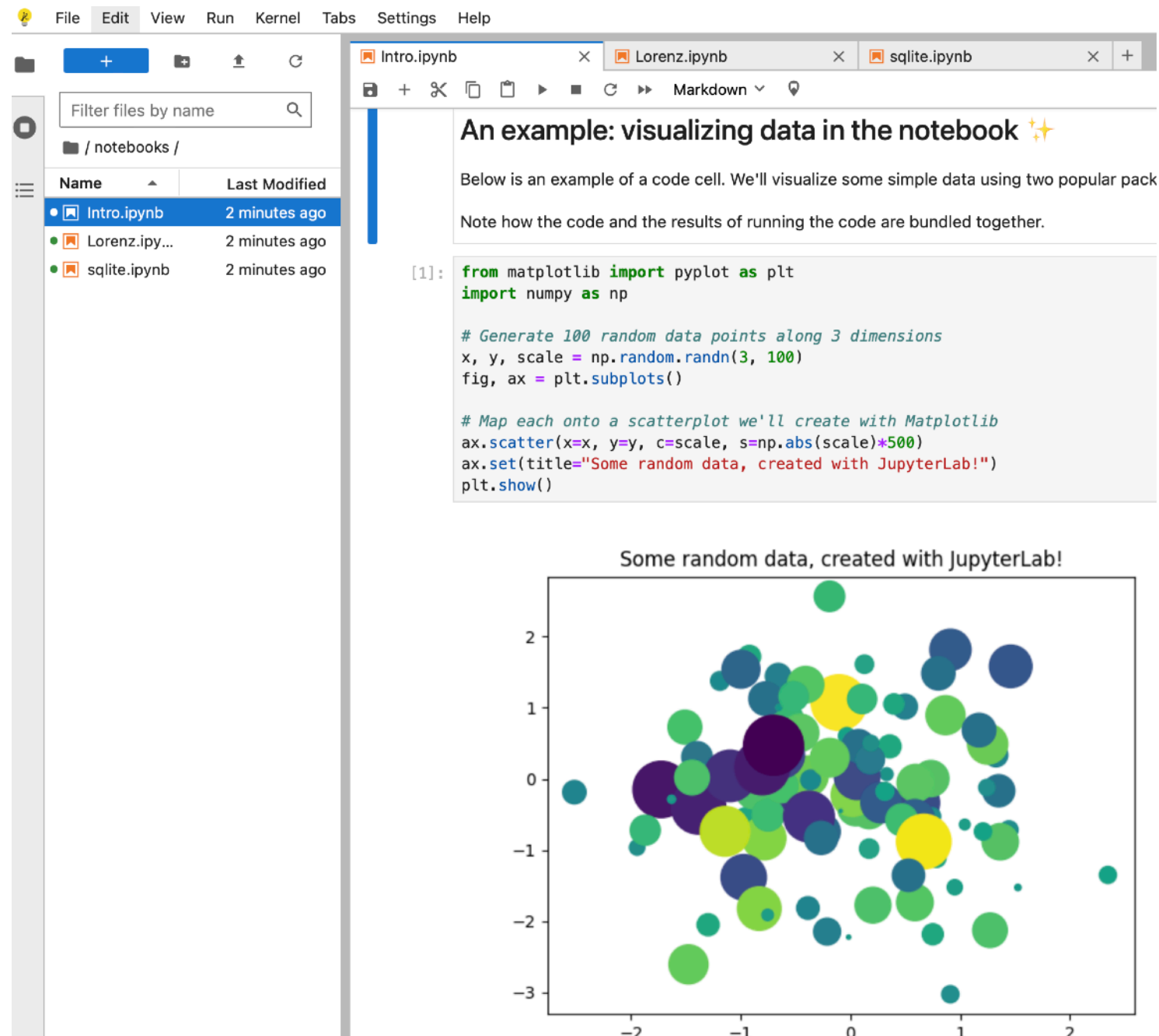
## **NVIDIA DGX H100**

The gold standard for AI infrastructure.



# KIN'L-OPENLAB Software: JupyterLab

- Web-based interactive development environment
- Flexible interface to configure and arrange workflows
- Modular design to enrich functionality



The screenshot displays the JupyterLab web interface. On the left, a file browser shows a directory named 'notebooks' containing three files: 'Intro.ipynb', 'Lorenz.ipynb', and 'sqlite.ipynb', all modified '2 minutes ago'. The main area shows a code cell in 'Intro.ipynb' with the following Python code:

```
[1]: from matplotlib import pyplot as plt
import numpy as np

# Generate 100 random data points along 3 dimensions
x, y, scale = np.random.randn(3, 100)
fig, ax = plt.subplots()

# Map each onto a scatterplot we'll create with Matplotlib
ax.scatter(x=x, y=y, c=scale, s=np.abs(scale)*500)
ax.set(title="Some random data, created with JupyterLab!")
plt.show()
```

Below the code cell, a scatter plot titled "Some random data, created with JupyterLab!" is displayed. The plot shows 100 data points in a 2D space, where the x and y coordinates are random values between -2 and 2. The size and color of each point are determined by a third random value, 'scale', ranging from approximately -3 to 3. The points are scattered across the plot area, with larger and more vibrant points (yellow, green, blue, purple) representing higher absolute values of the 'scale' parameter.

# Python

---



## Python (programming language)

---

[Article](#) [Talk](#)

---

From Wikipedia, the free encyclopedia

**Python** is a [high-level, general-purpose programming language](#). Its design philosophy emphasizes [code readability](#) with the use of [significant indentation](#).<sup>[33]</sup>

Python is [dynamically typed](#) and [garbage-collected](#). It supports multiple [programming paradigms](#), including [structured](#) (particularly [procedural](#)), [object-oriented](#) and [functional programming](#). It is often described as a "batteries included" language due to its comprehensive [standard library](#).<sup>[34][35]</sup>

[Guido van Rossum](#) began working on Python in the late 1980s as a successor to the [ABC programming language](#) and first released it in 1991 as Python 0.9.0.<sup>[36]</sup> Python 2.0 was released in 2000. Python 3.0, released in 2008, was a major revision not completely [backward-compatible](#) with earlier versions. Python 2.7.18, released in 2020, was the last release of Python 2.<sup>[37]</sup>

Python consistently ranks as one of the most popular programming languages, and has gained widespread use in the [machine learning](#) community.<sup>[38][39][40][41]</sup>



# Deep learning libraries

● TensorFlow  
Software

● PyTorch  
Computer application

+ Add comparison

Germany ▼

Past 5 years ▼

All categories ▼

Web Search ▼

Interest over time ⓘ



# Hands-on, practical examples

---

- A JupyterLab example of handwritten digits recognition from the MNIST database using PyTorch
- Local Large Language Models like Llama3 contrasted with ChatGPT

# Let's talk!

---

- We want you to be part of the **KIN'L-Satellites**
- AI use cases in your:
  - Teaching
  - Research
  - Transfer
- Potential to apply for grants and funding
- Other ideas

# Acknowledgments

---

Funding reference 13FH009KI2

SPONSORED BY THE



Federal Ministry  
of Education  
and Research



**FORSCHUNG AN  
FACHHOCHSCHULEN**

# Any further questions? Please get in touch with us!

---

Email

**[Jorge.Ruiz@hnee.de](mailto:Jorge.Ruiz@hnee.de)**

Webpage

**<https://www.hnee.de/kinl>**

(Work in progress)