ASEM JOINT MEETING | 9 JUNE 2026

From Training Machines to Teaching Students to Think with AI

Data analysis, data science, and AI education in the age of agentic AI

Dr Johannes Bleher
Senior Lecturer in Statistics and Data Analysis
University of Hohenheim | AIDAHO Steering Committee

Methods education is about intellectual autonomy.

Understand
What does the system compute?

Question
Which assumptions and evidence matter?

Govern
Who remains accountable for use?

> **We teach scientific methods so students do not become passive subjects of automated answers.**

Refs: Long and Magerko 2020; UNESCO 2023; Parasuraman and Riley 1997

What Has Changed

AI lowers operational friction

- code generation
- method explanation
- debugging support
- documentation summaries
- workflow suggestions

Some friction carried learning signal

- What is the question?
- Which assumptions matter?
- What does the code compute?
- Why is this answer plausible?
- What would make it wrong?

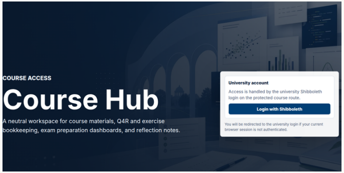
> **Observing a solution is not the same as explaining, testing, and transferring it.**

Refs: Chi et al. 1989; Kasneci et al. 2023; Dell’Acqua et al. 2023; Parker et al. 2026

Local tools show the same design principle: make practice, feedback, and transfer visible.

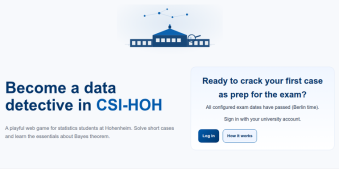
Repeated practice

Small quests, streaks, and exam preparation routines.



Course workspace

Materials, preparation, reflection, and accountability in one place.



Transfer through cases

Playful statistical problems that require interpretation.

Screenshots: teaching tools developed for my courses. Aggregate usage data are accessible only to me and the teaching team. Captured 6 June 2026.

Teaching Context

students per year
about 800
two large in-person courses

course type
statistics
introductory data analysis

setting
university
AI and data science teaching

Large introductory courses are a stress test for AI-era teaching.

- Many students meet formal data analysis for the first time.
- The hard part is not only computing an answer, but asking what the answer means.
- Teaching at scale needs feedback loops, repetition, transfer, and governance.

Refs: Hattie and Timperley 2007; Roediger and Karpicke 2006

A Learning Model

1	**Ask**	Define the question and the object of analysis.
2	**Try**	Use tools, code, and AI assistance deliberately.
3	**Explain**	Interpret the result and expose assumptions.
4	**Govern**	Check limits, risks, and responsibility.

> **The goal is not to ban assistance. It is to keep students inside the reasoning loop.**

Refs: Chi et al. 1989; Hattie and Timperley 2007; Roediger and Karpicke 2006

Learning principles need teaching infrastructure.

	Learning function	Resource for teachers
Spacing	Repeated contact with core concepts across weeks.	Scheduling, item banks, and LMS/app integration.
Challenging	Effortful tasks that require retrieval, explanation, and checking.	Authoring capacity, coding agents, and domain-level quality control.
Randomisation	Mixed contexts that force method choice, not recipe following.	Parameterised tasks, testing infrastructure, and privacy-safe logs.
Feedback	Early visibility of misconceptions.	Aggregate dashboards and teaching-team workflows.

> **Spacing, challenge and randomisation become feasible when lecturers have deployable tools, model access, support, and orchestration skills.**

Refs: Cepeda et al. 2006; Bjork and Bjork 2011; Rohrer 2012; Hattie and Timperley 2007; NIST 2023

What Digital Support Should Do

	Student side	Lecturer side
Orientation	Find the next task and see what counts as progress.	Reduce confusion about expectations and sequencing.
Feedback	Mark uncertainty, difficulty, and open questions.	Detect where explanations or discussion are needed.
Adaptation	Practise the same concept in spaced and varied contexts.	Build small course-specific apps when teaching needs change.

> **Agentic coding shifts scarce capacity: lecturers can adapt the app to the course, but only with infrastructure that supports deployment and maintenance.**

Refs: Cepeda et al. 2006; Rohrer 2012; Hattie and Timperley 2007

A Lecturer Mental Model

Teaching need → Agentic build loop → Deployed teaching app

Deploy apps	AI inference	Admin agents	Orchestrate systems

What changes Lecturers can prototype course-specific tools instead of waiting for one-size-fits-all platforms.

What is needed Hosting, authentication, model access, coding agents, administrative automation, and the skills to coordinate them.

> **Teacher flexibility depends on an institutional stack that lets ideas become maintained teaching tools.**

A check-in tool must support peer learning without becoming social scoring.

Voluntary	Students opt in; non-participation has no grading consequence.
Non-binding signal	A lightweight score or streak can encourage regular meetings, but it is not part of assessment.
Minimal check	The system confirms that a learning session happened; it does not check the purpose or content of the meeting.
No social graph	The system should not store who met whom, create rankings, or expose individual patterns; the teaching team sees aggregate data only.

> **The goal is to lower coordination costs for peer learning, not to evaluate students' social behaviour.**

Refs: Johnson and Johnson 2009; Tinto 1997; Pointon-Haas et al. 2024; van Kessel et al. 2025; NIST 2023

Agentic AI Changes The Skill Target

Earlier target	Write code	Run code	Report output	
AI-era target	**Frame problem**	**Steer agent**	**Audit output**	**Accountability**

> **The decisive competence is understanding the analytical object well enough to direct, evaluate, and govern automated work.**

Refs: Long and Magerko 2020; Parasuraman and Riley 1997; Dell'Acqua et al. 2023

A Conference Anecdote

"It is still my writing, because it is still my thinking."

What changed	A researcher described using Claude to turn a difficult analytical problem into a workable draft over lunch.
What did not change	The ideas, prior reading, modelling judgement, and quality control still came from the researcher.
Teaching lesson	AI can compress execution time, but only when the user can ask the right questions and audit the answer.

> **Agency shifts from producing every sentence manually to directing and judging the intellectual work.**

The evidence supports a conditional claim, not a simple automation story.

Productivity	Some writing and support tasks become faster. Effects vary by worker and task.
Limits	Performance can fall outside the tool's competence frontier, especially when users cannot audit outputs.
Complements	Returns depend on workflows, training, data, and governance, not on tool access alone.

Refs: Dell'Acqua et al. 2023; Noy and Zhang 2023; Brynjolfsson et al. 2023; Acemoglu and Restrepo 2018; Brynjolfsson et al. 2018

What Teachers Need From Universities

Deployment	Hosting, authentication, monitoring, privacy-preserving logs, and reliable app updates.
AI inference	Model access for coding agents, feedback agents, and administrative support tasks.
Orchestration skills	Lecturers need enough technical judgement to specify, test, and maintain these systems.
Governance	Privacy, assessment rules, transparency, accountability, and clear ownership.

> **The scarce resource is not only technology. It is the capacity to turn teaching ideas into trustworthy infrastructure.**

Refs: UNESCO 2023; Brynjolfsson et al. 2018; NIST 2023

The Bridge To Governance

Teaching perspective

- Students need autonomy, not dependence.
- Methods education must train judgement.
- Learning systems need privacy-aware feedback.

Institutional perspective

- Governance is capacity, not a document.
- Adoption without oversight creates hidden risk.
- Universities need shared infrastructure and rules.

> **AI education is civic capacity: students need judgement to shape AI for public value, especially where power, inequality, and misuse are at stake.**

Refs: UNESCO 2021, 2023; NIST 2023; OECD 2019/2024; Weuts et al. 2025

Three Practical Principles

1	**Keep students in the loop**	Every task should require framing, interpretation, and critique.
2	**Build visible feedback**	Learning systems should surface uncertainty, difficulty, and misconceptions early.
3	**Agent infrastructure**	Agents require investment, rules, training, and accountable deployment.

Refs: UNESCO 2023; Parasuraman and Riley 1997; Hattie and Timperley 2007; NIST 2023

Teach students to think with AI, not to let AI think for them.

They need enough understanding to ask sharper questions, evaluate outputs, and use AI to extend their own thinking.

Dr Johannes Bleher | University of Hohenheim | AIDAHO

Refs: Synthesis of sources listed in the appendix

Selected References

Learning and education

Long and Magerko (2020), AI literacy, CHI.

UNESCO (2023), Generative AI in education.

Parasuraman and Riley (1997), automation, Human Factors.

Chi et al. (1989), self-explanation, Cognitive Science.

Kasneci et al. (2023), LLMs in education.

Cepeda et al. (2006), spacing, Psychological Bulletin.

Bjork and Bjork (2011), desirable difficulties.

Rohrer (2012), interleaving, Educational Psychology Review.

Hattie and Timperley (2007), feedback.

Roediger and Karpicke (2006), retrieval practice.

Parker et al. (2026), longitudinal GenAI use in teacher education, Computers and Education Open.

Johnson and Johnson (2009), cooperative learning.

Tinto (1997), learning communities.

Pointon-Haas et al. (2024), peer support and well-being.

van Kessel et al. (2025), belonging and outcomes, Australian Educational Researcher.

Work, productivity, governance

Dell'Acqua et al. (2023), jagged frontier, HBS Working Paper.

Noy and Zhang (2023), productivity effects, Science.

Brynjolfsson, Li and Raymond (2023), generative AI at work, NBER.

Acemoglu and Restrepo (2018), automation and work, NBER.

Brynjolfsson, Rock and Syverson (2018), productivity J-curve, NBER.

NIST (2023), AI Risk Management Framework.

OECD AI Principles (2019/2024); UNESCO (2021), AI ethics. Weuts et al. (2025), AI governance in higher education, arXiv.