Fortgeschrittene Statistische Software für Nebenfachstudierende

SoSe 2026 | 6 ECTS

Overview

This course builds on Einführung in die Statistische Software für Nebenfachstudierende to develop the skills needed to design and carry out a complete data analysis independently. Where Einführung focuses on building foundational R skills through individual exercises, this course shifts emphasis toward collaborative, project-based work — integrating those skills across a complete analysis pipeline and adding the workflow practices (version control, reproducible reporting, critical use of AI tools) needed for real-world data work. Rather than working through R systematically, students learn to reason about data analysis as a whole — identifying what needs to be done at each stage of a pipeline, making and justifying decisions, and recognising where things can go wrong silently.

The course has two parts. The first six weeks establish foundational programming and workflow practices: writing functions, debugging, version control with Git and GitHub, and reproducible reporting with Quarto. The second part is project-based: small groups work with real datasets over seven weeks, applying and extending their skills across themes including data acquisition, cleaning, analysis, and communication.

AI assistance (LLMs) is an explicit part of the course. Students are expected to use it critically — as a tool that requires checking and judgment, not a substitute for understanding.

Prerequisites: Students should have completed Einführung in die Statistische Software für Nebenfachstudierende or have equivalent experience. This course assumes familiarity with: basic R syntax and data types, importing and manipulating data frames, data wrangling (filtering, reshaping, joining datasets), simple visualisation with ggplot2, descriptive statistics, and writing basic R scripts. No advanced statistical or programming background is assumed.

Course Rationale

This course takes a deliberate departure from a tools-first approach to statistical programming. Rather than systematically teaching R syntax or specific packages, the course treats data analysis as a design and reasoning problem — one in which tools, including AI assistants, serve as means to an end.

The central premise is that effective data analysts are distinguished not by how much syntax they have memorised, but by their ability to reason clearly about data, make

deliberate decisions at each stage of an analysis pipeline, and recognise when something might be silently wrong. Students who can articulate why a particular transformation is appropriate, what assumptions underlie a statistical choice, and how a visualisation could mislead are better equipped to use any tool — R packages, LLMs, or otherwise — than students who have simply been walked through them.

The course is structured in two halves that serve different learning functions. The first six weeks introduce foundational concepts and practices at a pace that is intentionally ambitious. Students should expect to encounter more ideas than they can fully absorb in the moment. This is by design: the goal is to build an initial mental model of the data analysis lifecycle — what the pieces are, how they connect, and what questions are worth asking. Partial understanding at this stage is not failure; it is the starting point.

The second half gives students the space to practice, consolidate, and extend what they have encountered. Each week is organised around a different theme of tasks drawn from typical data-driven workflows — loosely based on the R4DS cycle, but not treated as strictly linear stages, and not exhaustive of every possible analysis context. The aim in each week is to practice planning and executing tasks within that theme, as well as evaluating and integrating results, artefacts, and insights across themes and collaborators. Working in small groups on real datasets, students coordinate with and learn from one another to craft a meaningful data-driven report or story. LLM assistance is explicitly available and encouraged, but within a framework: students are expected to reason from first principles before delegating to a language model, evaluate outputs critically, and be able to explain and defend every decision in their final analyses.

Throughout, statistical workflow and communication practices — version control with Git and GitHub, reproducible reporting with Quarto — are treated as non-negotiable professional skills, not optional extras. The oral examination reflects this emphasis: students are assessed on their ability to reason about their work, not merely to reproduce it.

Learning Objectives

By the end of this course, students will be able to:

Foundational programming and workflow skills

1. Write well-structured R scripts and functions, and use Git, GitHub, and Quarto to maintain reproducible, version-controlled analyses.
2. Diagnose and resolve errors systematically, and seek out R packages and functions independently as analysis needs arise.

Data analysis pipeline design

3. Describe the key stages of a data analysis pipeline and the decisions that arise at each stage, from data acquisition through to communication.

4. Import, inspect, clean, link, and transform real-world datasets, documenting the assumptions embedded in each step.
5. Construct statistically appropriate visualisations and recognise common ways that graphics can mislead.

Critical reasoning and quality awareness

6. Identify warning signs of questionable analytical practices — including hidden assumptions, untested data quality issues, and silent errors — and design pipelines that make these visible rather than hiding them.
7. Use LLM assistance as a starting point rather than an answer, applying first-principles reasoning to check, question, and take responsibility for generated output.

Communication and collaboration

8. Present and defend analytical decisions clearly — in written reports, visualisations, and verbally — including the assumptions made and alternatives considered.
9. Synthesis and communicate statistical findings in audience and venue appropriate formats – reports, articles, interactive dashboards etc.

Course Format

The class consists of weekly lectures where new concepts and tools will be introduced, and hands-on practicals where you will have the opportunity to practice these new skills.

Weekly lectures: Thursday, 10:00-12:00 Schellingstr. 3 (S) / S 001

Weekly practicals: Friday, 14:00-16:00 Schellingstr. 3 (S) / S 004

Group project

From Week 6 onwards, the practicals support an optional but strongly recommended group project. In groups of 3–4, students choose a real dataset, carry out a data analysis, and produce a data story or report published as a Quarto website on GitHub. The project runs in parallel with the lecture themes — each week's content maps directly onto a stage of the project workflow.

Examination Requirement: Oral Examination

The examination format for this subject is Oral Examination. The examination questions will be based on lecture content, practical exercises and the optional but recommended group project.

Students who complete the project will be examined on their own work. Students who do not submit will be examined on an unseen dataset and analysis context.

Course outline

Part 0: Recap & Preliminaries

- W01 (17 Apr): Introduction

Part 1: Statistical Programming Foundations (W2–5)

- W02 (24 Apr): Scripts, Functions & Refactoring
- W03 (01 May): Debugging
- W04 (08 May): Version Control & Collaborative Coding
- W05 (15 May): Quarto websites
- W06 (22 May): R Packages ~~Open datasets~~

Part 2: Working with Real World Data (W6–10)

Obtaining & understanding data

- W07 (29 May): Open Data & Initial data analysis
- No class (05 Jun)

Analysing & presenting findings

- W08 (12 Jun): Data Cleaning
- W09 (19 Jun): Analysis strategies
- W10 (26 Jun): Statistical Communication & Visualisation

Part 3: Advanced Topics & Summary (W11–13)

- W11 (03 Jul): Interactive vis & data storytelling
- W12 (10 Jul): Multi-lingual analysis (Python/R)
- W13 (17 Jul): Review/summary/Outro

Instructors

This course is taught by Dr. Cynthia A. Huang, Leonhard Kestel, and Lisa Bondo Andersen.

Contact

In case you have any questions or problems you can approach us during practicals or contact us via statprog@stat.uni-muenchen.de.

Acknowledgements

This course draws on many open source teaching materials, including but not limited to:

- Business Analytics courses developed by Monash NUMBATs
- R4DS textbook
- ModernDive labs
- RStats WTF Debugging

- various discussions, blog posts, talks from Nick Tierney, Mine Cetinkaya-Rundel and others.
- Prior versions of this course at SODA-LMU

This syllabus was drafted with AI assistance. The course overview, rationale, objectives and group project summary sections were expanded using Claude Sonnet 4.6 from preliminary notes written by the instructors.