Course Overview

Advanced Statistical Methods and Models in Experimental Design

Bartosz Maćkiewicz

What this course is

- Workshop-style course: **learn by doing**
- Focus: modern statistical modeling for experimental designs in cognitive science
- Emphasis on:
    - correct model choice + justification
    - diagnostics and robustness
    - clear reporting (tables, figures, interpretation)
    - reproducible workflows

Goals

1. Learn how to *read* experimental papers
2. Learn how to *choose* the right statistical methods
3. Learn how to *perform* statistical analyses in R
4. Learn how to *describe* their results in a professional manner

Inferential statistics

In this course, we will focus on **inferential statistics**, with emphasis on **null hypothesis significance testing** (NHST).

Inferential statistics

- drawing conclusions about the population
- estimating parameters of the population
- testing hypotheses about population(s)
- examples: statistical tests, confidence intervals, etc.

Descriptive statistics

- describing a sample
- summarizing data
- plotting data
- examples: sample mean, sample standard deviation, boxplot, etc.

Overview

1. Foundations (Weeks 1–2)
    - review of classical statistical tests
    - reproducible workflows and reporting
    - power analysis and effect sizes
2. Generalized Linear Model (Weeks 3–8)
    - linear regression; nominal predictors and contrasts
    - assumptions, diagnostics, and data transformations
    - analysis of variance
    - link functions, logistic regression, Poisson regression
3. Complex experimental designs (Weeks 9–12)
    - repeated measures and mixed designs
    - hierarchical models
4. Dimension reduction and SEM (Week 13)
5. Meta-analysis (Week 14)
6. Final project workshop (Week 15)

Statistics and other disciplines

There are many similarities and overlaps between statistics and other disciplines such as Machine Learning and Data Science.

That’s OK.

Let’s consider linear models (gross simplification!):

Machine Learning

- overall performance of the model
- accuracy of predictions
- features selection
- models comparison

Statistics

- hypothesis testing
- explanation (% of variance accounted for, effect sizes)
- statistical significance of individual predictors

Two types of students

1. “Statistical Sharks”
    - They have a solid understanding of the logic of hypothesis testing.
    - They are familiar with common statistical tests and models.
    - They have some experience with analyzing data using R or other similar software.
    - They have some programming skills (preferably with R).
2. “Squirrels of Statistics”
    - They have little to no previous exposure to inferential statistics or want to consolidate knowledge.
    - They have never analyzed “real” experimental data, only toy examples.
    - They do not feel very confident in their programming skills

Some of you are Sharks, and some of you are Squirrels.

Hard problem: how to accommodate both Sharks and Squirrels?

Accommodating Sharks and Squirrels

1. “Statistical Sharks”
    - In the majority of assignments there will be more creative and slightly harder exercises that expand upon the knowledge of basics.
    - Some exercises will be explicitly marked as **harder**. If you are ahead, treat them as your “stretch goals”.
2. “Squirrels of Statistics”
    - I will provide them with additional resources on learning R as a programming language.
    - Deadlines can be made more flexible for students who need more time to catch up with material.

Assessment (continuous; no exam)

- Homework assignments (3–4 total): 50%
- Final project: 40%
- Active participation: 10%

Assignments (what you actually do)

Every week: in-class lab (workshop, required)

Most weeks include an in-class **lab worksheet**. The purpose is to make sure you actually practice the workflow:

- loading data and basic wrangling
- fitting the model we discuss in class
- checking assumptions / diagnostics
- writing short, correct interpretations

These labs are also the main way participation is assessed.

Sometimes: take-home homework (a few total)

Instead of weekly homework, there will be **a small number of take-home assignments** (a few in total across the semester).

They are longer, more complete mini-analyses.

Final project

- Submit one source file that contains both analysis and final written interpretation.
- Data access must be reproducible (built-in data, stable public source, or instructor-approved access for your own/lab data).
- Week 15 presentation materials (slides/HTML) are separate and do not replace the final report.

Academic integrity (general)

Cheating and plagiarism (including copying work from other students or internet sources without proper attribution) are serious violations of academic integrity and may be reported to the administration.

Policy on generative AI tools (ChatGPT, Copilot, Claude, etc.)

The use of Generative AI tools is **permitted for coding assistance** but **not permitted for writing the final scientific interpretation**.

Permitted uses (examples)

- Debugging help (understanding error messages, suggesting fixes)
- Syntax help (how to write a loop, how to use a function)
- Quick reminders about package usage (e.g., how to reshape data, how to create a plot)

Not permitted

- Generating the final written interpretations, discussion, or conclusions in your homework/final project report
- Submitting AI-generated text as if it were your own scientific reasoning
- Using AI to produce work you cannot explain

Disclosure requirement (mandatory)

If you used AI to generate or substantially modify **any code**, you must mark it directly in your source with an explicit comment, for example:

```
# Code block suggested by ChatGPT (then edited by me)
```

You must be able to explain your submission

You must be able to explain every line of code you submit. Inability to explain your own code will be treated as a potential academic integrity violation (and may result in a failing grade for the assignment/project).

Why this policy exists (the learning perspective)

Large Language Models can be useful for small coding tasks and quick reminders, but they can also block the development of the skills that matter most in statistics and modeling:

- reading documentation
- debugging
- recognizing when output is nonsense
- interpreting models responsibly

If you outsource those steps, you may end up with results you cannot defend or explain.

Webpage

https://adv-stat.kursy.bartoszmackiewicz.pl

The page is used for:

1. Additional resources (books, articles, tutorials, etc.)
2. Assignments
3. Datasets that we'll be working with during classes

Where the rules live

See the course pages (website):

- Syllabus
- Assessment and grading
- Attendance rules
- How to Submit Assignments - Quarto (.qmd) Basics
- Tooling (R, RStudio, Quarto)

Exercise: first look at the worksheets!

1. Download the worksheet from the webpage.
2. Fill in the blanks with your solutions.
3. Upload the worksheet.

Review time!

Logic of NHST

1. Begin with a **research hypothesis**.
2. Set up the **null hypothesis**.
3. Construct the **sampling distribution** of the particular statistic on the assumption that H_0 is true.
4. **Collect** some data.
5. **Compare** the sample statistic to that distribution.
6. Decide to reject or retain H_0 based on the probability, under H_0, of observing a sample statistic as extreme as the one obtained.

Example

Here we will present an example based on James Bond who insisted that martinis should be shaken rather than stirred.

Let's consider a hypothetical experiment to determine whether Mr. Bond can tell the difference between a shaken and a stirred martini. Suppose we gave Mr. Bond a series of 16 taste tests.

In each test, we flipped a fair coin to determine whether to stir or shake the martini. Then we presented the martini to Mr. Bond and asked him to decide whether it was shaken or stirred. Let's say Mr. Bond was correct on 13 of the 16 taste tests.

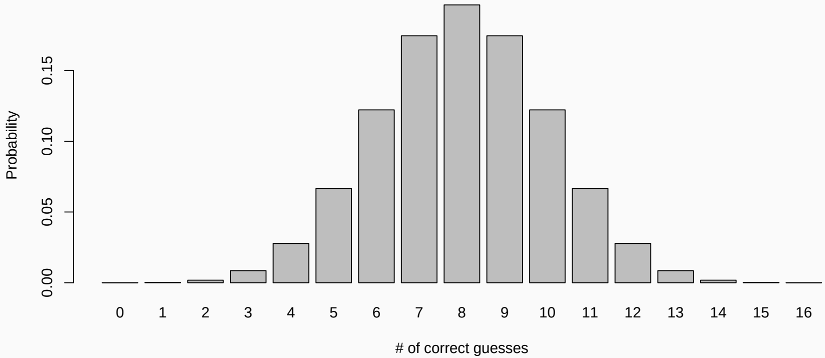
Does this prove that Mr. Bond has at least some ability to tell whether the martini was shaken or stirred?

NHST in action

1. Begin with a **research hypothesis**.
   - Mr. Bond has some ability to tell whether the martini was shaken or stirred.
2. Set up the **null hypothesis**.
   - Mr. Bond is no better than chance (H_0: $Pr = 0.5$)
3. Construct the **sampling distribution** of the particular statistic on the assumption that H_0 is true.
   - binomial distribution: $B(16, 0.5)$
4. Collect some data.
   - on 13 taste tests Mr. Bond was correct
5. Compare the sample statistic to that distribution.
   - what is $Pr(13, 5, 0.5)$?
6. Reject or retain H_0, depending on the probability, under H_0, of a sample statistic as extreme as the one we have obtained.
   - R to the rescue!

R in action

```
barplot(dbinom(0:16, 16, 0.5),
        names.arg = 0:16,
        ylab = "Probability",
        xlab=  "# of correct guesses"
        )
```



R in action

```r
# using density function
sum(dbinom(13:16, 16, 0.5))
```

```
[1] 0.01063538
```

```r
# using cumulative distribution function
pbinom(13-1, 16, 0.5, lower.tail = FALSE)
```

```
[1] 0.01063538
```

Type I error

The first kind of error is the mistaken rejection of a null hypothesis as the result of a test procedure. This kind of error is called a type I error (false positive) and is sometimes called an error of the first kind.

In the context of the Mr. Bond example, a Type I error would occur if we incorrectly conclude that he can distinguish between stirred and shaken martinis when, in fact, he cannot.

Type II error

The second kind of error is the mistaken acceptance of the null hypothesis as the result of a test procedure. This sort of error is called a type II error (false negative) and is also referred to as an error of the second kind.

In the context of the Mr. Bond example, a Type II error would occur if we fail to recognize his ability to distinguish between stirred and shaken martinis when he actually possesses this ability