

Syllabus

Special Topics in Probability and Statistics

OPRE001

I. COURSE FEATURES

Program: Mathematics		
Prerequisites: Calculus, differential equations		
Duration: 72 hours	Academic year: 2026/2	
Instructors: Prof. Fernando Deeke Sasse		
Email: fernando.sasse@udesc.br		

II. SUMMARY

This course provides a comprehensive introduction to modern statistical thinking, exploratory data analysis, inferential statistics, and statistical modeling. Utilizing the Python programming ecosystem—specifically Jupyter Notebooks, Pandas, Seaborn, and Statsmodels—students will bridge the gap between mathematical theory and computational application.

III. COURSE OBJECTIVES

Students must learn to:

1. Identify various data types, study designs, and sampling methods.
2. Perform exploratory data analysis (EDA) and create high-quality data visualizations.
3. Construct confidence intervals and conduct hypothesis testing for varied statistical scenarios.
4. Fit, evaluate, and interpret statistical models, including linear regression, logistic regression, multilevel models, and foundational Bayesian inference.
5. Translate statistical formulas and concepts into robust Python code.

IV. TOPICS

1. Overview of Statistics: Variable types, study design, and data collection.
2. Introduction to computational statistics: Python environments and basic data types.
3. Data structures: Principles of cleaning and manipulating datasets.
4. Pandas library fundamentals: DataFrames, reading datasets, filtering, and sorting.
5. Numerical summaries: Mean, median, standard deviation, and the empirical rule.
6. Univariate visualizations: Constructing histograms and boxplots using Matplotlib and Seaborn.
7. Multivariate data analysis: Associations between categorical and quantitative variables.
8. Advanced Seaborn: Creating interactive scatterplots and multivariate visualizations.
9. Populations and Sampling: Probability vs. non-probability sampling and sampling variance.
10. Computational sampling: Generating random data and simulating sampling distributions using NumPy.
11. Exploratory Data Analysis (EDA) consolidation and practical applications.
12. Implementation: Executing a complete EDA on a real-world dataset.

13. Foundations of statistical inference: Estimating population proportions and means.
14. Computational inference: Calculating standard errors and confidence intervals using SciPy and Statsmodels.
15. Advanced confidence intervals: Mean differences for paired data and independent groups.
16. Python automation: Scripting confidence interval calculations across multiple variables.
17. Hypothesis Testing fundamentals: Setting up tests, p -values, and significance levels.
18. Computational testing: Executing z -tests and t -tests using Statsmodels.
19. Advanced Hypothesis Testing: Evaluating differences in proportions and means (paired and independent).
20. Interpreting computational outputs: Reading and analyzing statistical summaries in Python.
21. Synthesizing statistics: The mathematical relationship between confidence intervals and hypothesis testing.
22. Analytical case studies: Formulating formal research questions and translating them into analysis code.
23. Inferential statistics consolidation and practical lab applications.
24. implementation: Structuring and computing an end-to-end inferential analysis report.
25. Introduction to statistical model fitting: Dependent vs. independent variables and inference vs. prediction.
26. Environment preparation: Setting up the Python statistics ecosystem for advanced modeling.
27. Linear Regression theory: Mathematical notation, parameters, and Ordinary Least Squares (OLS) estimation.
28. Statsmodels API for Linear Regression: Fitting models, interpreting summary tables, and plotting residuals.
29. Logistic Regression theory: Binary classification, probability notation, and interpreting log-odds.
30. Computational Logistic Regression: Fitting models, evaluating goodness-of-fit, and plotting predicted probabilities.
31. Multilevel Models: Understanding hierarchical data structures and marginal models.
32. Python implementation of Multilevel Models: Handling and interpreting mixed-effects models.
33. Foundations of Bayesian Inference: Comparing Bayesian statistical paradigms to frequentist approaches.
34. Computational Bayesian statistics: Basic modeling concepts and Python case studies.
35. Comprehensive course review: Synthesizing statistical theory with computational Python implementation.
36. Implementation: Fitting, evaluating, and presenting mathematical models applied to complex datasets.
37. Bayesian intro (priors/posterior; credible intervals; small case study)
38. Bootstrap fundamentals: resampling logic, standard errors, failure modes
39. Bootstrap for regression (coef uncertainty; prediction intervals)
40. Permutation/randomization tests (difference in means; compare to t-test)
41. Spatial data basics (stationarity; variogram intuition; empirical variogram)

- 42. Ordinary kriging pipeline
- 43. Spatial statistics and Kriging: variograms (nugget/sill/range), ordinary kriging, prediction variance, cross-validation

V. LEARNING METHODOLOGY

The course employs a hybrid pedagogical approach that bridges rigorous mathematical theory with applied computational practice. The learning process is driven by the following core methodologies:

- **Theoretical Foundations:** Direct instruction detailing the mathematical principles behind statistical inference, probability distributions, and predictive modeling. This ensures a deep understanding of the mechanics underlying statistical algorithms before they are applied via software.
- **Computational Laboratory Sessions:** Hands-on, guided practice using interactive environments like Jupyter Notebooks and Google Colab. Students will actively translate theoretical formulas into functional Python code using libraries such as Pandas, NumPy, and Statsmodels.
- **Problem-Based Learning (Resolução de Problemas):** The curriculum heavily emphasizes applied problem-solving. Students will be tasked with analyzing real-world datasets, requiring them to formulate formal research questions, select the appropriate statistical frameworks, and execute the analysis computationally.
- **Project-Oriented Development:** Learning is consolidated through iterative, end-to-end projects. Students will progress from performing basic exploratory data analysis to fitting and evaluating complex multivariable models, ultimately producing comprehensive, reproducible analytical reports.

VI. GRADING

Assignments: 20%
Midterm exam: 20%
Final exam: 60%

IV. BIBLIOGRAPHY

1. MONTGOMERY, Douglas C.; RUNGER, George C. Applied statistics and probability for engineers. 5th ed. Hoboken: Wiley, 2011. 768 p
2. NETER, John; WASSERMAN, William. Applied linear statistical models: regression, analysis of variance, and experimental designs, 3rd ed. Boston: Irwin, c1990. 1181p.
3. LAWSON, John. Basic experimental strategies and data analysis for science and engineering. Boca Raton: CRC Press, Taylor & Francis Group, [2017] 1 online resource.
4. PEARL, Judea; GLYMOUR, Madelyn; JEWELL, Nicholas P. Causal inference in statistics: a primer. Chichester, West Sussex, United Kingdom: J. Wiley & Sons, 2016. xvii, 136 p.
5. EVERITT, Brian. Chance Rules: an informal guide to probability, risk and statistics.

- New York, NY: Springer New York: Springer New York, 2008. VI, 142 p
6. Downey, Allen B. Think Bayes: Bayesian Statistics in Python ,2nd ed., O'Reilly, 2021, <https://allendowney.github.io/ThinkBayes2/>
 7. JAMES, Gareth; WITTEN, Daniela; HASTIE, Trevor; TIBSHIRANI, Robert; TAYLOR, Jonathan. An introduction to statistical learning with applications in Python. Cham: Springer, 2023.
 8. VANDERPLAS, Jake. Python data science handbook: essential tools for working with data. Sebastopol: O'Reilly Media, 2016.
 9. DIEZ, David M.; CETINKAYA-RUNDEL, Mine; BARR, Christopher D. OpenIntro statistics. 4th ed. [S.l.]: OpenIntro, Inc., 2019.
 10. DAVIDSON-PILON, Cameron. Probabilistic programming & Bayesian methods for hackers. Boston: Addison-Wesley Professional, 2015.
 11. THE PANDAS DEVELOPMENT TEAM. Pandas: powerful Python data analysis toolkit. [S.l.]: NumFOCUS, 2024.
 12. WASKOM, Michael L. Seaborn: statistical data visualization. [S.l.]: NumFOCUS, 2021.
 13. SEABOLD, Skipper; PERKTOLD, Josef. Statsmodels: econometric and statistical modeling with Python. Austin: Proceedings of the 9th Python in Science Conference, 2010.