C S 8A: INTRODUCTION TO DATA SCIENCE

Foothill College Course Outline of Record

Heading	Value
Effective Term:	Summer 2024
Units:	4.5
Hours:	4 lecture, 2 laboratory per week (72 total per quarter)
Advisory:	Students will benefit from some experience with computer programming or statistics; demonstrated proficiency in English by placement via multiple measures OR through an equivalent placement process OR completion of ESLL 125 & ESLL 249.
Degree & Credit Status:	Degree-Applicable Credit Course
Foothill GE:	Non-GE
Transferable:	CSU/UC
Grade Type:	Letter Grade (Request for Pass/No Pass)
Repeatability:	Not Repeatable

Student Learning Outcomes

- AÊsuccessfulÊstudent will be able transform raw data to a more interpretable format byÊcreating tables,Êcharts, and plots using a modernÊsoftware language.
- AÊsuccessfulÊstudent will be able to analyze data usingÊsimulation models andÊstatistical techniquesÊsuch asÊcalculation ofÊsummaryÊstatistics,Êcalculation ofÊconfidence intervals, and regression, and will be able to interpret findings from these techniques.
- AÊsuccessfulÊstudent will be able to explain key dataÊscienceÊconceptsÊsuch asÊcorrelation vs.Êcausation, randomness,Êsampling, and uncertainty.

Description

Introduction to the fundamental concepts and computational skills needed to understand and analyze data arising from real-world phenomena. Topics include key data science concepts such as correlation vs. causation, randomness, sampling, uncertainty, predictive models, and classification. Using a tool such as Jupyter notebooks, students write code for transformation and use of data tables, simulation models, and A/B testing.

Course Objectives

The student will be able to:

1. Write and execute code in an environment such as Jupyter notebook.
2. Use expressions, variables, comparisons, control statements, iteration, arrays, and function calls in writing a computer program.
3. Transform raw data into tables and manipulate data tables using a package such as pandas, babypandas, or datascience.
4. Create and interpret a histogram, bar chart, line plot, and scatter plot.
5. Define and use a function in a computer program.
6. Group data by one or more attributes, apply a function ("split-apply-combine"), and interpret the results.
7. Join structured data tables.
8. Calculate probability that an event occurs and describe the situations where probabilities are added vs. multiplied.
9. Explain randomness, sampling, probability distributions, and sample mean at an introductory level.
10. Describe simulation models and the use of bootstrap.
11. Design, perform, and interpret hypothesis tests using simulation models.
12. Describe the meaning of variability in data.
13. Describe the relationship between sample size, accuracy of an estimate, and margin of error in light of the central limit theorem.
14. Calculate and interpret confidence intervals.
15. Interpret correlation coefficients.
16. Describe how linear and logistic regression can be used for predictive models.
17. Describe the general workings of classification.
18. Distinguish between causation measured through randomized experiments vs. association observed and describe why trends do not necessarily describe causal scenarios.

Course Content

1. Observational and experimental data
   a. Treatment/variable/feature, observation, outcome, association
   b. Treatment group, control group, randomization, randomized controlled experiment/trial
   c. Comparison, causality
2. Use of an environment like Jupyter notebook for writing and executing code
3. Introduction to programming
   a. Expressions
   b. Named variables
   c. Call expressions
   d. Numeric and string data types
   e. Comparisons
   f. Arrays
   g. Conditional statements
   h. Iteration
4. Tables
   a. Reading data into a table from a file
   b. Selecting columns
   c. Selecting rows by index or feature
   d. Sorting tables
5. Data visualization
   a. Scatter plots, line plots, and bar charts
   b. Best practices
   c. Binning data
   d. Histograms
   e. Plotting more than one category with scatter plots, line plots, and bar charts
6. Functions

   a. Signature
   b. Docstring
   c. Body
   d. Return statement
7. Applying functions to data tables
   a. Applying a function to a column
   b. Classifying by one variable (split-apply-combine)
   c. Computing counts, summary statistics, or other operations by group
   d. Classifying by more than one variable
   e. Creating pivot tables
   f. Combining information from two or more tables using inner, outer, left, or right join functions
8. Chance
   a. Probability as a fraction
   b. Multiplying probabilities
   c. Adding probabilities
   d. Probability of at least one event
   e. Randomness
   f. Use of random number generator
9. Sampling and empirical distributions
   a. Sampling at random vs. deterministically
   b. Sampling with and without replacement
   c. Law of averages
   d. Creating a histogram of sampled values
   e. Uniform distribution
   f. Simulations using random sampling
10. Testing hypotheses
   a. Comparing simulation results of numeric variables to expected distributions
   b. Comparing simulation results of categorical variables to expected distributions
   c. Statistical bias
   d. Null vs. alternative hypotheses
   e. Test statistics
   f. P-values
11. Comparing samples
   a. Observational analysis with hypothesis testing
   b. Randomized controlled experiments
   c. Meta-analysis
12. Estimation
   a. Percentiles
   b. Bootstrap
   c. Confidence intervals
13. Central tendency and variability
   a. Mean
   b. Variability
   c. Standard deviation
   d. Normal distribution
   e. Central limit theorem
14. Regression
   a. Correlation
   b. Linear regression
   c. Least squares
   d. Residuals
   e. Regression for prediction and inference
   f. Fitted values
   g. Interpretation of regression coefficients and confidence intervals
15. Classification
   a. Training and testing datasets
   b. Classifier examples: nearest neighbor and decision trees
   c. Measuring accuracy
16. Conditional probability
17. Examples used throughout course
   a. Economic data
   b. Geographic data
   c. Document collections
   d. Social networks
   e. Public health
   f. Sports
   g. Law
   h. Medicine
   i. Science
   j. Literature
18. Other data science issues
   a. Social and legal issues around data analysis
   b. Privacy
   c. Data ownership

Lab Content

1. Familiarization with an environment such as Jupyter
   a. Navigating the environment
   b. Running code
   c. Reading and understanding error messages
2. Expressions
   a. Using mathematical expressions
   b. Defining variables
3. Table operations
   a. Finding total number of columns and rows
   b. Filtering by columns and rows
   c. Creating tables by typing in values or by reading from files
4. Data types and creating and extending tables
   a. String methods
   b. Converting between string and numeric data types
   c. Creating, operating on, and indexing arrays
5. Functions and visualizations
   a. Calling functions
   b. Defining functions
   c. Making functions that call other functions
   d. Applying functions to columns of a table
6. Visualizations
   a. Creating a histogram
   b. Creating a line plot
   c. Creating a scatter plot
7. Conditional statements, iteration, simulation
   a. Writing conditional statements
   b. Creating loops
   c. Generating a random choice

   d. Producing random samples
   e. Building a simulation
8. A/B testing
   a. Designing a simulation
   b. Choosing and applying a test statistic
   c. Interpreting the result
9. Sample means
   a. Determining a sample mean from the results of a simulation
   b. Varying parameters in a simulation to demonstrate concepts related to the Central Limit Theorem
   c. Using bootstrapping to produce confidence intervals
10. Regression
   a. Assessing correlation
   b. Fitting a best fit line to a scatter plot
   c. Using bootstrapping to produce a confidence interval on best fit line slope
11. Conditional probability
12. Other
   a. Importing code modules or libraries

Special Facilities and/or Equipment

1. Instructor access to a cloud provider such as Google Cloud, Microsoft Azure, Amazon EC2, or IBM Cloud.
2. A Kubernetes-based deployment of JupyterHub or similar and an assignment server that loads assignments into the students' environment.
3. Student access to a computer lab with the latest version of Anaconda or similar and an appropriate web browser installed.
4. Website or course management system with an assignment posting component and a forum component (where students can discuss course material and receive help from the instructor). This applies to all sections, including on-campus (i.e., face-to-face) offerings.
5. When taught via distance learning, the college will provide a fully functional and maintained course management system through which the instructor and students can interact.
6. When taught via distance learning, students must have currently existing email accounts and ongoing access to computers with internet capabilities.

Method(s) of Evaluation

Methods of Evaluation may include but are not limited to the following:

Tests and quizzes
Laboratory assignments and projects which include source code, sample runs, and documentation
Written homework
Final examination

Method(s) of Instruction

Methods of Instruction may include but are not limited to the following:

Lectures which include data science concepts, example code, and analysis of data science examples
Online labs (for all sections, including those meeting face-to-face/on-campus), consisting of:
1. A programming assignment webpage located on a college-hosted course management system or other department-approved internet environment. Here, the students will review the specification of each programming assignment and submit their completed lab work
2. A discussion webpage located on a college-hosted course management system or other department-approved internet environment. Here, students can request assistance from the instructor and interact publicly with other class members
Detailed review of programming assignments which includes model solutions and specific comments on the student submissions
In-person or online discussion which engages students and instructor in an ongoing dialog pertaining to all aspects of designing, implementing, and analyzing programs
When course is taught fully online:
1. Instructor-authored lecture materials, handouts, syllabus, assignments, tests, and other relevant course material will be delivered through a college-hosted course management system or other department-approved internet environment
2. Additional instructional guidelines for this course are listed in the attached addendum of CS department online practices

Representative Text(s) and Other Materials

Adhikari, Ani, John DeNero, and David Wagner. Computational and Inferential Thinking: The Foundations of Data Science. 2022.

Types and/or Examples of Required Reading, Writing, and Outside of Class Assignments

1. Reading
   a. Textbook assigned reading averaging 30 pages per week
   b. Reading the supplied handouts and modules averaging 10 pages per week
   c. Reading online resources as directed by instructor though links pertinent to programming
   d. Reading library and reference material directed by instructor through course handouts
2. Writing
   a. Writing technical prose documentation that supports and describes the programs that are submitted for grades

Discipline(s)

Computer Science