

# INTL450/550: Advanced Data Analysis in Python

## Introduction

David Carlson

February 4, 2019

# Contact Information

## Instructor

- David Carlson
- Office: CASE 140
- Email: [dcarlson@ku.edu.tr](mailto:dcarlson@ku.edu.tr)
- Office Hours: TUES 11.30 – 12.30; WED 1.30 – 2.30 (or by appointment)

# Course Objectives

- Learn Python 3 syntax

# Course Objectives

- Learn Python 3 syntax
- Understand basic programming concepts

# Course Objectives

- Learn Python 3 syntax
- Understand basic programming concepts
- Understand advanced data analysis problems and the needed tools to solve them

# Course Objectives

- Learn Python 3 syntax
- Understand basic programming concepts
- Understand advanced data analysis problems and the needed tools to solve them
- Establish a basic understanding of advanced machine learning concepts and algorithms

# Course Objectives

- Learn Python 3 syntax
- Understand basic programming concepts
- Understand advanced data analysis problems and the needed tools to solve them
- Establish a basic understanding of advanced machine learning concepts and algorithms
- Improve upon academic and professional writing

# Course Objectives

- Learn Python 3 syntax
- Understand basic programming concepts
- Understand advanced data analysis problems and the needed tools to solve them
- Establish a basic understanding of advanced machine learning concepts and algorithms
- Improve upon academic and professional writing
- Identify and properly analyze a question in a relevant field



# Course Objectives

- Learn Python 3 syntax
- Understand basic programming concepts
- Understand advanced data analysis problems and the needed tools to solve them
- Establish a basic understanding of advanced machine learning concepts and algorithms
- Improve upon academic and professional writing
- Identify and properly analyze a question in a relevant field
- Contribute to statistical software development

# Why Python?

- Great for beginners and advanced use

# Why Python?

- Great for beginners and advanced use
  - ▶ Easily readable code

# Why Python?

- Great for beginners and advanced use
  - ▶ Easily readable code
  - ▶ Indentation forces organization

# Why Python?

- Great for beginners and advanced use
  - ▶ Easily readable code
  - ▶ Indentation forces organization
  - ▶ Online resources

# Why Python?

- Great for beginners and advanced use
  - ▶ Easily readable code
  - ▶ Indentation forces organization
  - ▶ Online resources
- Widely used, especially in scientific computing

# Why Python?

- Great for beginners and advanced use
  - ▶ Easily readable code
  - ▶ Indentation forces organization
  - ▶ Online resources
- Widely used, especially in scientific computing
- Powerful

# Why Python?

- Great for beginners and advanced use
  - ▶ Easily readable code
  - ▶ Indentation forces organization
  - ▶ Online resources
- Widely used, especially in scientific computing
- Powerful
  - ▶ Machine learning modules



# Why Python?

- Great for beginners and advanced use
  - ▶ Easily readable code
  - ▶ Indentation forces organization
  - ▶ Online resources
- Widely used, especially in scientific computing
- Powerful
  - ▶ Machine learning modules
- Open-source

# Why Python?

- Great for beginners and advanced use
  - ▶ Easily readable code
  - ▶ Indentation forces organization
  - ▶ Online resources
- Widely used, especially in scientific computing
- Powerful
  - ▶ Machine learning modules
- Open-source
- In demand

# Required Background

- No programming experience needed

# Required Background

- No programming experience needed
- Understanding of generalized linear models assumed

# Required Background

- No programming experience needed
- Understanding of generalized linear models assumed
  - ▶ Maximum likelihood

# Required Background

- No programming experience needed
- Understanding of generalized linear models assumed
  - ▶ Maximum likelihood
  - ▶ Linear algebra

# Required Background

- No programming experience needed
- Understanding of generalized linear models assumed
  - ▶ Maximum likelihood
  - ▶ Linear algebra
  - ▶ Basic probability

# Required Background

- No programming experience needed
- Understanding of generalized linear models assumed
  - ▶ Maximum likelihood
  - ▶ Linear algebra
  - ▶ Basic probability
  - ▶ Linear modeling assumptions



# Course Outline

- Familiarize with Python 3 and advanced data analysis techniques

# Course Outline

- Familiarize with Python 3 and advanced data analysis techniques
- Should be able to read any book after the course to learn new modeling strategies

# Course Outline

- Familiarize with Python 3 and advanced data analysis techniques
- Should be able to read any book after the course to learn new modeling strategies
- Python syntax and programming concepts (data types, functions, loops, recursion, classes, inheritance)

# Course Outline

- Familiarize with Python 3 and advanced data analysis techniques
- Should be able to read any book after the course to learn new modeling strategies
- Python syntax and programming concepts (data types, functions, loops, recursion, classes, inheritance)
- Data base management, creation, manipulation, and visualization

# Course Outline

- Familiarize with Python 3 and advanced data analysis techniques
- Should be able to read any book after the course to learn new modeling strategies
- Python syntax and programming concepts (data types, functions, loops, recursion, classes, inheritance)
- Data base management, creation, manipulation, and visualization
- Overview of Bayesian statistics; calling Stan through Python

# Course Outline

- Familiarize with Python 3 and advanced data analysis techniques
- Should be able to read any book after the course to learn new modeling strategies
- Python syntax and programming concepts (data types, functions, loops, recursion, classes, inheritance)
- Data base management, creation, manipulation, and visualization
- Overview of Bayesian statistics; calling Stan through Python
- Machine learning models

# Course Outline

- Familiarize with Python 3 and advanced data analysis techniques
- Should be able to read any book after the course to learn new modeling strategies
- Python syntax and programming concepts (data types, functions, loops, recursion, classes, inheritance)
- Data base management, creation, manipulation, and visualization
- Overview of Bayesian statistics; calling Stan through Python
- Machine learning models
- Projects and presentations

# Readings

- 2 books; can be read for free online



# Readings

- 2 books; can be read for free online
- Do not recommend buying them

# Readings

- 2 books; can be read for free online
- Do not recommend buying them
- Shaw, Zed A. 2017. *Learn Python 3 the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code (Zed Shaws Hard Way Series)*. 1st Edition. Addison-Wesley. Available at: <https://learnpythonthehardway.org/python3/>

# Readings

- 2 books; can be read for free online
- Do not recommend buying them
- Shaw, Zed A. 2017. *Learn Python 3 the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code (Zed Shaws Hard Way Series)*. 1st Edition. Addison-Wesley. Available at: <https://learnpythonthehardway.org/python3/>
- VanderPlas, Jake. 2016. *Python Data Science Handbook: Essential Tools for Working with Data*. O'Reilly Media. Available at: <https://jakevdp.github.io/PythonDataScienceHandbook/>

# Readings

- 2 books; can be read for free online
- Do not recommend buying them
- Shaw, Zed A. 2017. *Learn Python 3 the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code (Zed Shaws Hard Way Series)*. 1st Edition. Addison-Wesley. Available at: <https://learnpythonthehardway.org/python3/>
- VanderPlas, Jake. 2016. *Python Data Science Handbook: Essential Tools for Working with Data*. O'Reilly Media. Available at: <https://jakevdp.github.io/PythonDataScienceHandbook/>
- Additional readings listed on the syllabus on Blackboard

# Readings

- 2 books; can be read for free online
- Do not recommend buying them
- Shaw, Zed A. 2017. *Learn Python 3 the Hard Way: A Very Simple Introduction to the Terrifyingly Beautiful World of Computers and Code (Zed Shaws Hard Way Series)*. 1st Edition. Addison-Wesley. Available at: <https://learnpythonthehardway.org/python3/>
- VanderPlas, Jake. 2016. *Python Data Science Handbook: Essential Tools for Working with Data*. O'Reilly Media. Available at: <https://jakevdp.github.io/PythonDataScienceHandbook/>
- Additional readings listed on the syllabus on Blackboard
- Articles will be posted on Blackboard

# Homework

- Both graded and ungraded homework

# Homework

- Both graded and ungraded homework
- All listed on syllabus on Blackboard

# Homework

- Both graded and ungraded homework
- All listed on syllabus on Blackboard
- Graded homework will be posted as a pdf on Blackboard and GitHub



# Homework

- Both graded and ungraded homework
- All listed on syllabus on Blackboard
- Graded homework will be posted as a pdf on Blackboard and GitHub
- Collaboration is encouraged, but every keystroke must be your own

# Homework

- Both graded and ungraded homework
- All listed on syllabus on Blackboard
- Graded homework will be posted as a pdf on Blackboard and GitHub
- Collaboration is encouraged, but every keystroke must be your own
- Graded homework: 50%

# Homework

- Both graded and ungraded homework
- All listed on syllabus on Blackboard
- Graded homework will be posted as a pdf on Blackboard and GitHub
- Collaboration is encouraged, but every keystroke must be your own
- Graded homework: 50%
- All homework, when applicable, needs to be done on git

# Homework

- Both graded and ungraded homework
- All listed on syllabus on Blackboard
- Graded homework will be posted as a pdf on Blackboard and GitHub
- Collaboration is encouraged, but every keystroke must be your own
- Graded homework: 50%
- All homework, when applicable, needs to be done on git
- Look at other's homework as a last resort, and cite their work

# Final Project

- Can be done in groups (2—4 students) or individually; reflected in grading

# Final Project

- Can be done in groups (2—4 students) or individually; reflected in grading
- Replication code and report appropriate for your field

# Final Project

- Can be done in groups (2—4 students) or individually; reflected in grading
- Replication code and report appropriate for your field
- Hypothesis, data report, methods with explanation, findings

# Final Project

- Can be done in groups (2—4 students) or individually; reflected in grading
- Replication code and report appropriate for your field
- Hypothesis, data report, methods with explanation, findings
- Null results are fine



# Final Project

- Can be done in groups (2—4 students) or individually; reflected in grading
- Replication code and report appropriate for your field
- Hypothesis, data report, methods with explanation, findings
- Null results are fine
- Exploratory work generally discouraged

# Final Project

- Can be done in groups (2—4 students) or individually; reflected in grading
- Replication code and report appropriate for your field
- Hypothesis, data report, methods with explanation, findings
- Null results are fine
- Exploratory work generally discouraged
- Benchmarks listed in the syllabus

# Final Project

- Can be done in groups (2—4 students) or individually; reflected in grading
- Replication code and report appropriate for your field
- Hypothesis, data report, methods with explanation, findings
- Null results are fine
- Exploratory work generally discouraged
- Benchmarks listed in the syllabus
- Presentation (time permitting) ungraded

# Final Project

- Can be done in groups (2—4 students) or individually; reflected in grading
- Replication code and report appropriate for your field
- Hypothesis, data report, methods with explanation, findings
- Null results are fine
- Exploratory work generally discouraged
- Benchmarks listed in the syllabus
- Presentation (time permitting) ungraded
- Worth 50% of grade

- Version control

# GitHub

- Version control
- Open-source development

# GitHub

- Version control
- Open-source development
- Collaboration

# GitHub

- Version control
- Open-source development
- Collaboration
- Let others know your skill



- Version control
- Open-source development
- Collaboration
- Let others know your skill
- Beautiful merging of collaborative work

# GitHub

- Version control
- Open-source development
- Collaboration
- Let others know your skill
- Beautiful merging of collaborative work
- Branches for sub-projects

# GitHub

- Version control
- Open-source development
- Collaboration
- Let others know your skill
- Beautiful merging of collaborative work
- Branches for sub-projects
- Free (unless you want private repos)

## To Do Before Wednesday

- Read <https://git-scm.com/docs/user-manual.html> up until the section “Exploring Git history”

# To Do Before Wednesday

- Read <https://git-scm.com/docs/user-manual.html> up until the section “Exploring Git history”
- Shaw, Appendix

# To Do Before Wednesday

- Read <https://git-scm.com/docs/user-manual.html> up until the section “Exploring Git history”
- Shaw, Appendix
- Shaw, Exercises 0 — 15

# To Do Before Wednesday

- Read <https://git-scm.com/docs/user-manual.html> up until the section “Exploring Git history”
- Shaw, Appendix
- Shaw, Exercises 0 — 15
- Install Anaconda  
<https://docs.anaconda.com/anaconda/install/>

# To Do Before Wednesday

- Read <https://git-scm.com/docs/user-manual.html> up until the section “Exploring Git history”
- Shaw, Appendix
- Shaw, Exercises 0 — 15
- Install Anaconda  
<https://docs.anaconda.com/anaconda/install/>
- Sign up for a free GitHub account <https://github.com/>



# To Do Before Wednesday

- Read <https://git-scm.com/docs/user-manual.html> up until the section “Exploring Git history”
- Shaw, Appendix
- Shaw, Exercises 0 — 15
- Install Anaconda  
<https://docs.anaconda.com/anaconda/install/>
- Sign up for a free GitHub account <https://github.com/>
- Install git <https://git-scm.com/downloads>

# To Do Before Wednesday

- Read <https://git-scm.com/docs/user-manual.html> up until the section “Exploring Git history”
- Shaw, Appendix
- Shaw, Exercises 0 — 15
- Install Anaconda  
<https://docs.anaconda.com/anaconda/install/>
- Sign up for a free GitHub account <https://github.com/>
- Install git <https://git-scm.com/downloads>
- Create a public repository called PythonCourse, and add me (carlson9) as a collaborator