COURSE SYLLABUS

Course Introduction

Course Title: Python for Interdisciplinary Data Science and Artificial Intelligence

Subject Code and Course Number: Data Science 222

Credit Hours: 3

Prerequisite Courses: DSCI 101 and CS 110 with a minimum grade of C- in each.

Instructor: Cody Hood, Ph.D.

Class Meets: Tuesday & Thursday. 1:30 PM – 2:15 PM. Hodges 221

Course Introduction: Data science requires skills not only to work with data, but also to document the steps that were used to acquire, clean, process and curate, visualize and analyze the data. In addition, being able to carry out analyses in both R and Python is an important aspect of the data science skill-set. Students will learn to program in Python and learn the Python skills equivalent to those that they learn using R in DSCI 221. The basic workflow is importing data, cleaning and processing it to then analyzing and visualizing. Data sets for the projects will be chosen from a wide variety of subject areas to meet the interests of the students in the class. Students will learn to use notebooks and scripts in various cloud and local -based platform environments. This class bring all your knowledge on interdisciplinary sciences, and more, within the context of computational applications and artificial intelligence to address a wide variety of problems in formal sciences, physical sciences, life sciences, health sciences, social sciences, and industry.

General Education Area and Learning Outcome (if relevant):
R and Python are the two programming languages used extensively in data science to produce pipelines to import, clean, transform, visualize and model large amounts of data. This course follows a similar layout as in DSCI 221 and introduces the basics using Python in different environments but using notebooks and repository managements as is common in practice. The course begins with a fast introduction to programming in Python and the fundamentals of a data science pipeline. Next some of the most widely used computational techniques are developed and analyzed within an application framework, thus using intermediate/advanced Python programming level. Finally, specific applications on interdisciplinary sciences and industry are introduced and developed through assignments. Students will work on projects using data from various sources to develop and refine their Python skills.

Faculty Contact Information

Instructor Office Location:

403B Armstrong Hall

Office Hours:

TBA

Instructor Email:

chood@mix.wvu.edu

Instructional Materials

Required Instructional Materials:

Course Format: Learning to program is a very hand-on activity. Students will be expected to study the material for each class and, when provided/requested, to try out the code in the textbook before coming to class. The class period will be spent understanding the concepts by using and developing appropriate Python code.

Required Technology: To be successful in this course, and in the data science program, students must have a laptop. Please contact the instructor if this poses a difficulty.

List of Books:

1. Algorithms to Live By: The Computer Science of Human Decisions by Brian Christian & Tom Griffiths.
2. Jake VanderPlas Python Data Science Handbook (2016) O'Reilly Media, Inc. or free, on-line at https://jakevdp.github.io/PythonDataScienceHandbook/
3. A Whirlwind Tour of Python by Jake Vanderplas. Free pdf available. The book is also available for purchase in paper form, if preferred.
4. Introduction to Python for Computational Science and Engineering by Hans Fangohr. [https://fangohr.github.io/introduction-to-python-for-computational-science-and-engineering/book.pdf]
5. Python for You and Me by Kushal Das. [https://pymbook.readthedocs.io]
6. Introduction to Machine Learning with Python: A Guide for Data Scientist by Andreas C. Muller & Sarah Guido.
7. LLM Engineer's Handbook by Julien Chaumond & Hamza Tahir.
8. Natural Language Processing with Transformers: Building Language Applications with Hugging Face.
9. Dive into Deep Learning by Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola. [http://d2l.ai]
10. Generative AI with Python and PyTorch: Navigating the AI frontier with LLMs, Stable Diffusion, and next-gen AI applications by Joseph Babcock & Raghav Bali.
11. Getting Started with PyTorch, an online course by ApXML. [https://apxml.com/courses/getting-started-with-pytorch]

The student does not require to acquire previous books; however, the avid reader can find additional insights and understanding when studying their content.

Optional Instructional Materials:

Other Sources: Data Science mainly relies on open-source software, and the good amount of code and support on coding available online. Students will learn to use these resources in class and to help each other with debugging code. Additional resources will be added on eCampus.

Course Learning Outcomes

Course Learning Outcomes:

Upon successful completion of this course students will be able to
1. Develop a reproducible workflow in Python to clean, process and visualize data.
2. Import, store, manipulate and transform data using NumPy, Pandas and other libraries.
3. Pre-process structured and semi-structured data using Python.
4. Apply basic machine learning methods provided by Scikit-learn, PyTorch and other libraries.
5. Produce enriched data visualizations using the highly customizable Python packages.
6. Explore additional Python libraries for preprocessing, manipulation, visualization and modelling of datasets in multiple data formats.
7. Use AI-based technologies to assist the data science pipeline.

Assessment

Short Descriptions of and Grading Criteria for Major Assignments/Assessments:

Assignments (60%): There will be 6 total assignments throughout the semester. The assignments will require students to explain Python code or concepts that were covered in the 1-2 weeks period prior to that assignment. The purpose of the assignments is to make sure that students are understanding the material week to week and to clear up any misconceptions. See the table in the section 'Weight/Distribution of Course Points' for a detailed breakdown of assignments and percentages.

Final Project: There will be three evaluations of the project during the semester, each evaluation measures their progress and status. Data Science consists of working on different projects where accessing, cleaning and organizing, visualizing and analyzing data are the key skills. For data science projects, it is important that reproducibility of the workflow is maintained. Grading will take into account all elements implemented through Python; such as, visualization, data accessing/cleaning/organizing/formatting/manipulation and the use of necessary mathematical/statistical aspects on the project. Students will give short presentations on the results/progress of their projects, and they will also upload the entire code in the format of a Jupyter notebook for the instructor. The file will include Python code along with narrative that discusses the data, explains the goals of the analysis, carries out the analysis and discusses the conclusions. Note that the analysis might be graphical or tidying and presenting summarized data. The end result will be (1) a Python code for the instructor and (2) presentation of final results at each evaluation of the project progress. The analysis will include graphical displays, descriptive statistics/tables and other forms of summarizing and presenting data/information.

Evaluation and grading are break down as follows,

- **Project Planning (7%)**: Students will develop a feasible plan to address a data science problem of their choosing. Although it is understandable that future activities could deviate from the original plan, please do your best to create a feasible and clear plan. Plans could be modified in the future, but the original/initial plan provide a direction in which we can focus our initial efforts.

- **Project Progress (13%)**: Students will present the current status of their project. Two assignments will help to show your findings and progress so far.
- **Project Demonstration (20%)**: Students will perform a final demonstration of their project to wider audience. Two assignments will help you to show your final findings, the tools you have created and to highlight the importance of your project.

See the table in the section 'Weight/Distribution of Course Points' for a detailed breakdown of tasks/assignments and percentages for each project's stage.

The project demonstration will be graded according to the rubric below.

	A (10 pts)	**B (8 pts)**	**C (5 pts)**	**D (3 point)**	**F (0 pts)**
Discussion of datasets	Student includes sources, problem context and relevant data details	The student includes most of the relevant dataset information, few details are missing	Some critical aspects of the dataset details are missing which make difficult its correct usage in a data science environment	Most of the details are missing and it is not clear if the dataset selection is correct.	The dataset(s) is/are not included
Goals of the analysis	The student makes important and insightful goals that match the context of the dataset at hand	The student proposes good goals for the analysis, but these are insufficient for a thoughtful analysis	Some of the proposed goals does not help to pursue the overall goal of the project	Most of details in the proposed goals are missing	Goals of the analysis are not included in the description
Visual elements	The student correctly uses plots, tables and other visual elements to explain their arguments	Some of the visual elements lack clarity. There is an area of opportunity to improve the plots, tables, etc.	A few of the expected visual elements are missing	The number of visual elements is insufficient to correctly analyze the project's findings	The visual elements are missing
	A (20 pts)	**B (15 pts)**	**C (10 pts)**	**D (5 pts)**	**F (0 pts)**
Mathematical & statistical aspects	Clear reasoning and proper use of available mathematical and statistical tools	The theory used for the modeling is in the project, but there are few missing details	There is ample are of opportunity to improve the modeling and data analytics	Modeling of the problem is unsatisfactory and most of math/stat elements are missing	Mathematical and/or statistical methods are not included in the project
	A (40 pts)	**B (30 pts)**	**C (20 pts)**	**D (10 pts)**	**F (0 pts)**
Python code	The code run smoothly in a Python environment. The numerical results match the goals of the analysis.	Most of the goals of the analysis are addressed in the Python script. Everything runs smoothly.	The Python code accomplishes most of the expected tasks. A few of them may not work properly.	Most of Python code does not run and/or does not match with the goals of the analysis	The Python script is missing.

	A (10 pts)	B (8 pts)	C (5 pts)	D (3 pts)	F (0 pts)
Conclusions and discussion	The main points were properly emphasized, and the arguments are supported by the findings	The summary of the findings and discussion was mostly clear.	The discussion and conclusions are mostly unclear. Many of the main points were missing and not clearly explained.	The student failed to discuss their findings.	Conclusions and discussions are missing.
	A (20 pts)	**B (15 pts)**	**C (10 pts)**	**D (5 point)**	**F (0 pts)**
Writing and overall presentation of content	The content is cohesive and well organized. The student effectively communicates their findings to a broader audience.	Although the content is cohesive and well organized, the student cannot properly communicate their findings.	The content lacks cohesiveness and organization, but the student is able to communicate the main ideas.	The organization of the content is poor. Student failed to communicate most of the content.	The overall writing and presentation of content is full of errors and fundamental misunderstandings.

Collected points from the aforementioned rubric (max 100) will be multiplied by the percentages in the project demonstration column (see 'Weight/Distribution of Course Points'). Thus, the same rubric applies to both tasks, project demonstration (video recording) and project demonstration (demonstration).
If necessary, this rubric structure may be modified according to the specific nature of each project, in such a case the new rubric would be shared with students in advance.

Feedback: Students will receive quick feedback throughout the course, both for individual students and for the class as a whole to highlight strengths and trouble spots in understanding the grammar of and writing code in Python. Assignments will be handled through the eCampus platform, and projects will be presented in the format given by the instructor during the class. If you are absent, you should make an appointment to meet with the instructor to receive your feedback or make additional arrangements.

Weight/Distribution of Course Points:

Total: 100%

Assignments	Project Planning	Project Progress	Project Demonstration
10%: Python Essentials	7%: Video Recording	7%: Video Recording	10%: Demonstration (Oral Presentation)
10%: Data Manipulation & Visualization		6%: Q&A Session in the Classroom	10%: Short Report including GitHub Repository
10%: Text & Graph Analysis			
10%: Signal Processing & Machine Learning			

10%: PyTorch			
10%: Parallel Comp. & LLMs			

Mid-Semester Grade:

Students will receive a midterm grade from the instructor. This grade will include the project planning (7%) and the first two assignments (20%) for a possible total of 27% of the overall course grade.

Expected Timeline of Major Assignments/Assessments and Topics/Units:

Tentative Schedule of Classes (note this is written for a generic 15-week semester):

* Topic descriptions and timelines are subject to change at the professor's discretion. Certain topics may be adjusted or not be covered as originally planned depending on classroom workflow, scheduling considerations, or unexpected classroom activities.

	Tuesday	**Thursday**
Week 1 8/21		Setting up the Integrated Development Environments (IDE), GitHub account, Overleaf account, and Virtual Environment software
Week 2 8/26-28	List Comprehension and Memory Management, Lambda Expression, and Latex typesetting system **Project Planning (deliver)**	Fundamentals of Numpy & Pandas, and Common Misconceptions
Week 3 9/2-4	Intermediate/advance Numpy and Multidimensional Operations **Assignment: Python Essentials (deliver)**	Intermediate/advance Pandas and Data Pre-processing
Week 4 9/9-11	Mixing Numpy, Pandas and Lambda Expressions **Assignment: Python Essentials (due date)**	Common Mathematical and Statistical Routines: API reference. **Project Planning (due date)** **Project Progress (deliver)**
Week 5 9/16-18	Fundamentals of Data Visualization, Latex and Text Rendering in Visualizations, and Common Misconceptions **Assignment: Data Manipulation & Visualization (deliver)**	Interactive Plotting and Maps, and Dynamic Visualizations: Animations
Week 6 9/23-25	Project Progress: Q&A Session **Assignment: Data Manipulation & Visualization (due date)**	Fundamentals of "text" as semi-structured data, and common misconceptions. Text editing and manipulation

Week 7 9/30-10/2	Text analysis, preprocessing and language translation	Graphs and Networks as semi-structured data via NetworkX. Fundamentals, Creating and Modifying Graph Structures, and Network Visualization
Week 8 10/7	Graph Operations, Computing Graphs Properties, and Traversal and Searching **Assignment: Text and Graph Analysis (deliver)**	
Week 9 10/14-16	Image Processing and OpenCV **Assignment: Text and Graph Analysis (due date)**	
Week 10 10/21-23	Algorithms for Supervised Classification using Scikit-Learn	
Week 11 10/28-30	Non-Linear and Linear Dimensionality Reduction using Scikit-Learn **Assignment: Signal Processing & Machine Learning (deliver)**	
Week 12 11/4-6	Basics of Neural Networks in PyTorch **Assignment: Signal Processing & Machine Learning (due date)** **Assignment: PyTorch (deliver)**	
Week 13 11/11-13	Basics of Large Language Models. Using the Google Gemini API **Assignment: PyTorch (due date)** **Assignment: Parallel Computing and LLMs (deliver)**	
Week 14 11/18-20	Interdisciplinary Applications: Mathematics & Statistics **Assignment: Parallel Computing and LLMs (due date)**	
Week 15 12/2-4	Interdisciplinary Applications: Health Sciences	
Week 16 12/9-11	**Project Demonstration (due date)**	

Note: The time allocation each week is approximately as follows; lecture, discussion and Q&A (40%), analysis of programming (40%), and additional group discussion (20%). Final project sessions focus on presentations, brainstorming and feedback from students and the instructor.

Final Grading Scale:

Final Grading Scale	A	B	C	D	F
Percentage	90 – 100%	80 – 89.9%	70 – 79.9%	60 – 69.9%	0 – 59.9%

Topic descriptions and timelines are subject to change at the professor's discretion. Certain topics may be adjusted or not be covered as originally planned depending on classroom workflow, scheduling considerations, or unexpected classroom activities.

Course and Institutional Policies

Attendance Policy:

Data Science is a very interactive field, and class will be interactive and will supplement material from the textbooks and online sources. To get the most benefit from the course and to succeed, you should actively attend all the classes. The material builds on concepts and programming, and it is easy to fall behind. A substantial portion of the grade is based on assignments (30%) which requires the skills obtained in class. Similarly, programming can be learned and practiced during class which is a necessary skill to develop a successful final project; the project is worth the remaining grade (70%). Overall grades are highly likely to suffer due to lack of attendance.

Participation Policy:

My goal as the instructor is to create an open and informal environment to help students actively participate while addressing a variety of viewpoints and skillsets. Whenever you have a question during class, please feel free to raise your hand and ask away. While we all experience some shyness, this class is designed to help students learn how to be active in the classroom. If you feel you will have any trouble participating in class, please see me at the start of the semester. Students will be provided needed assistance on a regular basis by attending my office hours. A lot of this class is writing code, and we all learn to write by working together and helping each other debug code. The class will be interactive, and asking for help as well as providing help is the best way for everyone to learn.

Late Assignment and Missed Exam Policy:

Each student will take the assignments and hand in projects and give oral presentations on the dates set for the entire class. Exceptions to this are under circumstances described in the WVU attendance policy. If you know you will be absent, please notify the instructor (via email) and plan on making arrangements before the scheduled date. Similarly, projects should be handed in before the due date, and arrangements should be made for the oral presentations with the instructor prior to the scheduled presentation day. If you have an emergency that prevents you from giving the instructor prior

notification, please get in touch as quickly as possible. In this case student responsibility for make-up work as per the Attendance Policy will be followed and full credit will be given for make-up work completed by the timeline deemed appropriate by the instructor.

Institutional Policies:
Students are responsible for reviewing policies on inclusivity, academic integrity, incompletes, sale of course materials, sexual misconduct, adverse weather, as well as student evaluation of instruction, and days of special concern/religious holiday statements.