School of Computing
UNIVERSITY OF GEORGIA

Course Information Sheet
CSCI 3360
Data Science I

Brief Course Description (50-words or less)	This course will be a rigorous overview of methods for text mining, image processing, and scientific computing. Core concepts in supervised and unsupervised analytics, dimensionality reduction, and data visualization will be explored in depth.
Extended Course Description / Comments	Provides students with an in-depth dive into data science with the Python language. Students will learn the theoretical justifications for statistical techniques in classification, clustering, and dimensionality reduction, learn how to implement them within the Python framework, and apply them to extracting knowledge from scientific datasets. Upon completion of this course, students will be able to design an algorithmic pipeline, leverage existing Python packages and incorporate their own to implement the pipeline, and visualize the results of their computations.
Pre-Requisites	(CSCI 1301-1301L or CSCI 1360 or CSCI 1360E) and (MATH 2250 or MATH 2250E or CSCI 2150-2150L)
Required, Elective or Selected Elective	Selected Elective Course
Approved Textbook	Author(s): Joel Grus Title: Data Science from Scratch: First Principles with Python Edition: 2nd Ed., 2019 ISBN-13: 978-1492041085 Author(s): Coelho, Richert, and Brucher Title: Building Machine Learning Systems with Python Edition: 3rd ed., 2018 ISBN-13: 978-1788623223
Specific Learning Outcomes (Performance Indicators)	1. Use existing Python tools to read and preprocess raw data of various formats (text, images, binary). 2. Choose a proper statistical model for extracting knowledge from a particular dataset, given the advantages and disadvantages of the model. 3. Implement at least one algorithm from the categories of regression, classification, clustering, and convex optimization. 4. Design and document analytical pipelines to be reproducible by others. 5. Use and interpret the results of dimensionality reduction on high dimensional datasets. 6. Choose the most effective visualization to convey the knowledge learned from the data.

ABET Learning Outcomes

A. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
B. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
C. Communicate effectively in a variety of professional contexts.
D. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.
E. Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.
F. Apply computer science theory and software development fundamentals to produce computing-based solutions.

Relationship Between Student Outcomes and Learning Outcomes

	ABET Learning Outcomes						
Specific Learning Outcomes		**A**	**B**	**C**	**D**	**E**	**F**
	1	●	●				●
	2	●	●				
	3		●				●
	4		●				●
	5	●	●				
	6	●	●	●			

Major Topics Covered

1. Scientific programming with Python (Knowledge level: Usage)

   a) Understand basic Python programming language features such as indentation, modules, functions, boolean expressions, strings, control-flow, zip, and basic object-oriented programming.
   b) Be able to use various built-in Python data structures to store and access data in order to solve problems. Data structures include list, tuple, dictionary, counter, and set.
   c) Make use of various commonly used toolkit and modules, such as Jupyter notebook, pandas, numpy, sklearn, scipy, matplotlib, keras, and tensorflow, to solve appropriate data science problems.

2. Data visualization (Knowledge level: Assessment)

   a) Given a dataset, create an appropriate plot to visualize the trend, relationships, and outliers within the data.
   b) Compare and contrast various plots, including bar chart, pie chart, histogram, line plot, and scatter plot, and choose the most effective plot(s) to communicate the data.

3. Convex optimization (Knowledge level: Usage)

   a) Compute the gradient using partial derivatives when feasible and estimate the gradient using difference quotients.

b) Perform multiple steps of gradient descent towards finding the minimum when given a univariate differentiable objective function.

4. Dimensionality Reduction (Knowledge level: varies by topic)

a) Using greedy forward selection and backward elimination algorithms to find a subset of features suitable for regression and classification models (Usage).
b) Introduces feature projection algorithms such as Principal Component Analysis (Familiarity).

5. Supervised Classification (Knowledge level: Usage)

a) Demonstrate understanding of the k-nearest neighbor model by computing distances between data points to identity k nearest neighbors and conclude the appropriate class label that should be given to a data point.
b) Demonstrate understanding of the naïve bayes model by computing the appropriate conditional probabilities based on Bayes' theorem in order to classify a data point.
c) Demonstrate understanding of the decision tree model by building a decision tree classifier using the ID3 algorithm when given an appropriate dataset and use the tree to classify data points.

6. Regression (Knowledge level: Usage)

a) Fit simple and multiple linear regression models to data. Be able to interpret the coefficients and compute the R-squared metric.
b) Apply regularization techniques such as ridge and lasso regression.
c) Build polynomial regression models with interaction features on a given dataset.

7. Unsupervised clustering (Knowledge level: varies by topic)

a) Demonstrate understanding of the K-means algorithm by repeatedly computing new cluster means and reassign data points to clusters until the assignment of data points stabilizes (Usage).
b) Introduces the hierarchical clustering algorithm and apply it on a dataset (Familiarity).

8. Introduction to "big data" and "deep learning" (Knowledge level: varies by topic)

a) Understand the topology of feedforward neural networks and be able to complete the feedforward process when given the dataset, weights, biases, and activation functions (Usage).
b) Introduces the backpropagation algorithm using a simple neural network structure (Familiarity).
c) Introduces "big data", "deep learning", convolutional neural networks, and their applications in image recognition (Familiarity).

Knowledge Levels	The following is the ACM's categorization of different levels of mastery: Assessment, Usage, and Familiarity. Note that Assessment encompasses both Usage and Familiarity, and Usage encompasses Familiarity. **Familiarity:** The student understands what a concept is or what it means. This level of mastery concerns a basic awareness of a concept as opposed to expecting real facility with its application. It provides an answer to the question "What do you know about this?" **Usage:** The student is able to use or apply a concept in a concrete way. Using a concept may include, for example, appropriately using a specific concept in a program, using a particular proof technique, or performing a particular analysis. It provides an answer to the question "What do you know how to do?" **Assessment:** The student is able to consider a concept from multiple viewpoints and/or justify the selection of a particular approach to solve a problem. This level of mastery implies more than using a concept; it involves the ability to select an appropriate approach from understood alternatives. It provides an answer to the question "Why would you do that?"
Modified	1/22/2024 by Dr. Jaewoo Lee and Dr. Hao Peng
Approved	Not yet