

## Course Syllabus

### Course Information

Course: MIS 6V99 – Special Topics – Programming for Data Science  
Section: 5U1  
Term: Summer 2017  
Meets: Friday, 6:00 pm to 10:00 pm, JSOM 2.106

Note: Beginning Fall 2017, this same course will be taught as:  
BUAN 6340 – Programming for Data Science

### Professor Contact Information

Instructor: Kevin R. Crook  
Email: Kevin.Crook@utdallas.edu  
Office Location: Office hours will be in classroom JSOM 2.106  
Adjunct office is JSOM 3.604  
Office Hours: Friday, 5:00 pm to 6:00 pm  
First regularly scheduled office hour  
6/2/2017  
Last regularly scheduled office hour  
8/4/2017  
Special office hour for regrading requests:  
8/14/2017 4:00 PM to 7:00 PM

Teaching Assistant: Leila Hosseini  
Email: lxh132430@utdallas.edu  
Office Location: JSOM 14.315  
Office Hours: Monday, 6:00 pm to 8:00 pm  
Wednesday, 6:00 pm to 8:00 pm

(next page)

## Course Pre-requisites, Co-requisites, and/or Other Restrictions

- The following courses or equivalent knowledge (recommended):
  - BUAN 6356 – Business Analytics with R
  - MIS 6323 – Object Oriented Programming
- Knowledge of statistics
- Prior programming experience with an object-oriented language (if not Python, then Java, C++, etc.)
  - Required:
    - Variables, data types, bitwise manipulation, control structures, looping, file I/O including binary files, procedures, functions, classes, objects, inheritance, etc.
  - Recommended:
    - Data structures: stacks, queues, trees, heaps, etc.
    - Algorithms: searching, sorting, graph, etc.
    - Computational complexity – Big O Notation, etc.
- Debugging skills – as is the standard with any advanced programming course, the students must be able to debug any syntax errors, semantic errors, stack overflow, core dumps, etc. independently on their own.
- Software installation skills – as is the standard with any advanced programming course, the students must be able to install and configure software packages independently on their own and work through and resolve any issues independently on their own.
- Database skills – as a secondary activity, programming assignments may require students to read and create ERDs, create database tables, and insert, update, delete, and select data using SQL.
- Source code control skills – students will need to use GitHub for source code assignments and to store their code for the course for protection against loss

(next page)

## Course Description

This course covers many aspects of programming for data science and analytics, including syntax, handling data, data visualization, and implementation of statistical analysis models. The course will be taught using Python language and may use a different programming language as applicable.

Specifics include:

- Python Programming Fundamentals
- Graph / Network Analytics (NetworkX)
- Statistical Libraries (NumPy, SciPy)
- Data Wrangling, Data Munging, Data Cleansing (Pandas)
- Data Visualization (matplotlib)
- Web Page Screen Scraping / Parsing HTML (Beautiful Soup)
- Interfacing Python with Relational Databases using SQL (SQLAlchemy)
- Interfacing Python with Social Media (Tweepy)
- Machine Learning
  - from libraries (Scikit-Learn)
  - from source code (since Scikit-Learn is open source, we will study the actual source code for several common machine learning algorithms)
  - Scale Out (brief look at Spark MLlib)
- Natural Language Processing (NLTK)
- Deep Learning / Neural Networks (TensorFlow)
  
- Programming Labs (in class)
  - Most weeks we will have a programming lab in class based on the previous week's content
  
- Programming Assignments (outside of class)
  - Scheduling Analytics
    - Programmatically creating and optimizing an airline schedule
  - Healthcare Analytics
    - Recommender system for hospitals on Medicare ratings and patient surveys
  - Market Basket Analytics
    - Given a purchasing history of products purchased together, recommend additional product to customers making purchases
  - Social Media Analytics / Natural Language Processing / Geographic Data Visualization
    - Collecting Tweets from Twitter, performing sentiment analysis using natural language processing, and displaying the results as a geographic data visualization

### Student Learning Objectives/Outcomes

- To understand the fundamental programming concepts of data science environments including their solution design, implementation and presentation by mapping technology strategy of an enterprise to its business requirements.
- Data Extraction and filtering from a large dataset stored in a variety of different formats.
- Perform complex statistical analysis on datasets.
- Predictive modeling on datasets using advanced analytical techniques as well as using machine learning algorithms and natural language processing techniques.
- Interactively visualize large datasets and present the results as a driver for adaptive business intelligence.

(next page)

## Required Textbooks and Materials

### Laptop:

Students are required to have a Windows 10 (64-bit) laptop, legitimately licensed copy, which they need to bring to class. Some of the software may run on older versions of Windows, but it will be at your own risk and responsibility. The software will run on 4 GiB of RAM, but more memory is better.

If you have a Mac, you can run the software used in this class on the Mac, but please be advised that students in the past have had major issues with the Mac and the Mac lack of competent technical support. If you use a Mac, you do so at your own risk and must resolve any technical issues on your own.

### Software:

This course will use the Anaconda managed data science distribution of Python and associated libraries. We will use Python 3.x. It can be downloaded and installed at the following link: <https://www.continuum.io/downloads>

### Textbooks:

For the textbooks for this course, many are available online for free from the UTD library, <http://www.utdallas.edu/library>. However, the library places a limit on the number of students who can view the textbooks simultaneously. Many of the textbooks are also available from Safari Books, [safaribooksonline.com](http://safaribooksonline.com).

The following textbooks are required for this course:

#### **Python Data Science Handbook**

By Jake VanderPlax, O'Reilly Media, Inc.

Free edition in Jupiter Notebook format:

<https://github.com/jakevdp/PythonDataScienceHandbook>

#### **Hands-On Machine Learning with Scikit-Learn and Tensorflow**

By Aurelien Geron, O'Reilly Media, Inc.

#### **Introducing Python**

By Bill Lubanovic, O'Reilly Media, Inc.

## Assignments & Academic Calendar

Week Number Class Date	Topics Covered in Class	Assignments Due
Week 1 June 2	Python Programming Fundamentals	
Week 2 June 9	Python Programming Fundamentals  Graph / Network Analytics (NetworkX)	
Week 3 June 16	Statistical Libraries (NumPy, SciPy)  Data Visualization (matplotlib)	
Week 4 June 23	Data Wrangling, Data Munging, Data Cleansing (Pandas)	
Week 5 June 30	Data Wrangling, Data Munging, Data Cleansing (Pandas)  Web Page Screen Scraping / Parsing HTML (Beautiful Soup)  Interfacing Python with Relational Databases using SQL (SQLAlchemy)  Interfacing Python with Social Media (Tweepy)	Programming Assignment #1 Due Sunday, July 2 at 11:59 PM
Week 6 July 7	Machine Learning (Scikit-Learn, Source Code, MLLib)	
Week 7 July 14	Machine Learning (Scikit-Learn, Source Code, MLLib)	
Week 8 July 21	Natural Language Processing (NLTK)	Programming Assignment #2 Due Sunday, July 23 At 11:59 PM
Week 9 July 28	Deep Learning / Neural Networks (TensorFlow)	

<p>Week 10 August 4</p>	<p>Deep Learning / Neural Networks (TensorFlow)</p>	<p>Programming Assignment #3 Due Sunday, August 6 At 11:59 PM</p> <p>Note: no work for any lab nor assignments 1, 2, and 3 can be submitted for credit after Sunday, August 6 at 11:59 PM due to the end of the semester</p>
<p>Final Exam</p>	<p>Final Exam will be a take home programming assignment</p>	<p>Final Exam will be a take home programming assignment Due Sunday, August 13 at 11:59 PM No late submissions will be accepted</p>
<p>Special Office Hours for Regrade Requests</p>	<p>Regrade requests can only be made in person during these special office hours:</p> <p>Monday, August 14 4:00 PM to 7:00 PM</p>	

(next page)

<b>Grading Policy</b>
-----------------------

**Rank Grading System** - this course will use the rank grading system.

**Programming Labs** – must be completed in class at the appointed time to earn any credit. Each programming lab will have a grading rubric detailing how many points students can earn for that lab. Students will be ranked from one to the number of students in the course based on the total number of points earned. In the case of any ties, ties will be broken based on average completion time of all programming labs.

**Programming Assignments** - each programming assignment will have a grading rubric detailing how many points students can earn for that assignment. Students will be ranked from one to the number of students in the course based on the total number of points earned. In the case of any ties, ties will be broken based on average completion time of all programming assignments.

**Final Exam** – the final exam will be a programming assignment in class and will have a grading rubric detailing how many points students can earn. Students will be ranked from one to the number of students in the course based on points earned. In the case of any ties, ties will be broken based on completion time of the final exam.

**Semester Ranking** - will be a weighted average of the student’s ranking in each of the following categories with any ties broken based on average completion time of all programming labs:

Programming Labs	40%
Programming Assignments	40%
Final Exam	20%
<b>Total</b>	<b>100 %</b>

**Semester Letter Grades** - will be assigned based on the Semester Ranking no harsher than the following:

A	Top 30%
A-	Next 20%
B+	Next 20%
B	Next 10%
B-	Next 5%
C+	Next 5%
C	Next 5%
C-	Next 5%
F	< 60% of Programming Labs Working < 60% of Programming Assignments Working Regardless of Semester Ranking



## Course & Instructor Policies

- **Professional Conduct** - expected at all times from all students. Examples of unprofessional conduct includes, but is not limited to:
  - Trying the “game the system”
  - Coming late to class (without good reason)
  - Leaving early from class (without good reason)
  - Talking in class
  - Working on other subjects while in class
  - Using electronic devices for other than the course materials
- **Make-up Programming Labs** – restricted to documented emergencies with sufficient written proof provided. Make-up programming labs will differ in content from the regular programming lab.
- **Make-up Exams** – restricted to documented emergencies with sufficient written proof provided. Make-up exams will differ in content from the regular exam.
- **Lab, Assignment, Exam Content** – Content for labs, assignments, and/or exams may include:
  - Reading assignments, even if not covered in class
  - Anything covered in class, even material only discussed verbally
- **Extra Credit** – no extra credit assignments are available.
- **Class Attendance** – required, except for legitimate emergencies. Please be on-time and remain until class is dismissed. Vacations, visa trips, and other travel plans are not legitimate emergencies and should be scheduled during official breaks.
- **Late Work** – Each lab or assignment will give instructions if late work will be accepted, and if so, what penalties will be applied. One second after the due time counts as late. For technical difficulties, students must submit proof such as a screen print and immediately email a copy with the assignment to the instructor within 1 minute of missing the deadline. No work may be submitted for credit after Sunday, August 6 at 11:59 PM due to the end of the semester.
- **Instructor Response Policy** - Instructor is only part time and has limited time to answer emails. Instructor will check email twice a week, so it’s best to ask questions in class, visit the TA’s office hours, or visit the instructor’s office hours.
- **Regrade Requests** – Regrade requests must be made in person during a special office hours convened for that purpose (see schedule above). Everything for the entire semester will be regraded and students may receive lower than initial scores.

### **Comet Creed**

*This creed was voted on by the UT Dallas student body in 2014. It is a standard that Comets choose to live by and encourage others to do the same:*

*“As a Comet, I pledge honesty, integrity, and service in all that I do.”*

### **UT Dallas Syllabus Policies and Procedures**

The information contained in the following link constitutes the University’s policies and procedures segment of the course syllabus.

Please go to <http://go.utdallas.edu/syllabus-policies> for these policies.

***The descriptions and timelines contained in this syllabus are subject to change at the discretion of the Professor.***