Draft: Syllabus, Spring 2016, for OIDD 311: Business Computer Languages: Scripting for Business Analytics 12:00–1:20 M & W, JMHH **380** Canvas:

Professor Steven O. Kimbrough, Instructor
Office hours: 565 JMHH, 9:30–11:00 and 1:30–3:00 Wednesdays, and by appointment

October 22, 2016

1 Class Description

OPIM 311, "Scripting for Business Analytics," may be framed as follows. *Business analytics* refers to modeling and analysis undertaken for purposes of management and supporting decision making. This course is about business analytics and exploratory computing in support of business analytics, it is about using MATLAB® and Python for these purposes, and the treatment is introductory.

The business analytics concept has received wide uptake and substantial attention in the business press. In fact, the *Zeitgeist* embraces analytics. Business school gurus tout it.[1] Consultants and practitioners proclaim it, repeating "big data" at every turn.[2] Geeks stun the establishment with impressively accurate predictions.[3] Jobs are plentiful even in a down economy and professional organizations start offering certification programs, sponsor conferences on business analytics, and publish specialized news magazines on the subject.[4] Although hype abounds, there are real achievements at hand.

The varieties of techniques and methods are numerous and growing, including simple equational models, constrained optimization models, probabilistic models, visualization, data analysis, various kinds of heuristics (and families of heuristics, called metaheuristics), machine learning (supervised and unsupervised), data mining, text mining, agent-based models, simulation, multiobjective optimization and decision making, and much more. Elementary modeling of this sort

[1] See, for example, "Creating Value with Business Analytics" (Kiron and Shockley, 2011), *Competing on Analytics: The New Science of Winning* (Davenport and Harris, 2007) and *Super Crunchers: Why Thinking-by-Numbers Is the New Way to Be Smart* (Ayres, 2007).

[2] See, for example, "Big data: The next frontier for competition" (McKinsey & Company, 2012) and *Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics* (Franks, 2012) and *The New York Times* announces "The Age of Big Data" (Lohr, 2012).

[3] See, for example, *The Signal and the Noise* (Silver, 2012) and various poll aggregators who performed so impressively in the 2012 U.S. presidential election.

[4] See INFORMS, `http://www.informs.org`; `http://informs.informs.org/Build-Your-Career/Analytics-Certification`; `http://meetings2.informs.org/analytics2013/`; `http://www.analytics-magazine.org/`.

can be undertaken in Excel and other spreadsheet programs, but "industrial strength" applications typically use more sophisticated tools, based on scripting languages.

Scripting languages are programming languages that are designed to be learned easily and to be used for special purposes, rather than for large-scale application programming. This course focuses on the special purposes associated with business analytics and teaches MATLAB and Python in this context. MATLAB and Python are widely used in practice (both in management and in engineering), as are the business analytic methods covered in the course.

Prior programming experience is useful, but neither required nor presumed for this course.

2 Texts

- Required text: *Automate the Boring Stuff with Python: Practical Programming for Total Beginners* Sweigart (2015).
- Recommended text: *MATLAB® Primer,* Davis (2011).
- MATLAB documentation, available at `http://www.mathworks.com/help/matlab/` and as PDFs on Canvas.
  - *MATLAB® Primer, R2014b* (MathWorks, 2014c), available as a PDF on Canvas, *getstart.pdf.*

    This will be our main text for MATLAB. The others are supplemental.
  - *MATLAB® Programming Fundamentals, R2014b* (MathWorks, 2014d), available as a PDF on Canvas, *matlab_prog.pdf.*
  - *MATLAB® Data Import and Export, R2014b* (MathWorks, 2014a), available as a PDF on Canvas, *import_export.pdf.*
  - *MATLAB® Graphics, R2014b* (MathWorks, 2014b), available as a PDF on Canvas, *graphg.pdf.*
- *An Introduction to Doing Analytics with MATLAB®: Exploratory Computation and Business Analytics* by Steven Orla Kimbrough (Kimbrough, 2014a). Referred to as the *DAMbook*, it is handed out as a PDF, *DAMbook-311.pdf.*
- *Python Real Fast* by Steven O. Kimbrough (Kimbrough, 2014b). Referred to as the *PRFbook*, it is handed out as a PDF as the file *PRFbook.pdf.*

In addition, various other readings will be assigned, especially various from various MATLAB manuals. These will generally be handed out or made available online.

MATLAB is a commercial product (installed on the Wharton lab machines). The main URL for MATLAB is `http://www.mathworks.com` (and see the Appendix B.2 of the *DAMbook* for related links).There is an open-source product that is in many ways similar to MATLAB: GNU Octave, `http://www.gnu.org/software/octave/`. There are student/educational versions for sale at a hefty discount for MATLAB. Alternatively, you can install Octave on your own computers for free.

Python is an open-source, "free" product. `http://www.python.org/` and `http://docs.python.org/tutorial/`. The following passage, from the Python Web site, is accurate.

> Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms. `http://docs.python.org/tutorial/`

There are several different builds of Python available. You can download and install versions from the official Python Web site: `http://www.python.org/`. In addition to version 3.1.2 from there, we'll be working with a "Python 3" build from Continuum Analytics `http://continuum.io/`, and their Anaconda Server and IDE, Spyder, `http://docs.continuum.io/anaconda/ide_integration.html` and `http://docs.continuum.io/anaconda-launcher/index.html`. I encourage you to download this version (and everything that comes with it) for your own computers. It is free for educational purposes.

In the spirit of open source, free software, there is much Python documentation freely available. We'll use some of it. A recommended, but not required, text is: *Programming in Python 3: A Complete Introduction to the Python Language*, (Summerfield, 2009). Also, *Python Essential Reference* by David M. Beazley (Beazley, 2009) is very useful for the day-to-day Python programmer.

Three other useful books: Bird et al. (2009) is an excellent introduction to . . . natural language processing with Python, and generally excellent all around. Beyond the scope of this course, but recommended anyway. Kreibich (2010) for a general introduction to SQLite. Friedl (2006) for an authoritative discussion of regular expressions.

3 Grades

Grading will be based on several components, roughly as follows.

45% Class participation and in-class exercises.

35% Three case assignments. One a small group assignment (2-3 people), 15%. One an individual case project, done in MATLAB, due after the end of the semester, 10%. One an individual case project, done in Python, due duing the semester, 10%.

20% Quizzes (2, 10% each).

Most of all, I want to see you engaged and involved in the class.

4 Class Schedule

1. Introduction and overview of the course.
   Reading (before class): (Sweigart, 2015, chaper 1), "Python Basics."
2. Python: Flow Control.
   Reading (before class): (Sweigart, 2015, chaper 2), "Flow Control."
3. Python: Functions.
   Reading (before class): (Sweigart, 2015, chaper 3), "Functions."
4. Python: Lists.
   Reading (before class): (Sweigart, 2015, chaper 4), "Lists."

5. Python: Dictionaries and Structuring Data.

   Reading (before class): (Sweigart, 2015, chaper 5), "Dictionaries and Structuring Data."

6. Python: Manipulating Strings.

   Reading (before class): (Sweigart, 2015, chaper 6), "Manipulating Strings."

7. Python: Pattern Matching with Regular Expressions.

   Reading (before class): (Sweigart, 2015, chaper 7), "Pattern Matching with Regular Expressions."

8. Python: Reading and Writing Files.

   Reading (before class): (Sweigart, 2015, chaper 8), "Reading and Writing Files."

9. Python: Organizing Files.

   Reading (before class): (Sweigart, 2015, chaper 9), "Organizing Files."

10. Python: Debugging.

    Reading (before class): (Sweigart, 2015, chaper 10), "Debugging."

11. Python: Web Scraping.

    Reading (before class): (Sweigart, 2015, chaper 11), "Web Scraping."

12. Python: Working with Excel Spreadsheets.

    Reading (before class): (Sweigart, 2015, chaper 12), "Working with Excel Spreadsheets."

13. Python: Working with CSV Files and JSON Data.

    Reading (before class): (Sweigart, 2015, chaper 14), "Working with CSV Files and JSON Data."

14. Python: pandas.

    Reading (before class): ***PRFbook***

*** * * Spring Break Saturday-Sunday March 7–15, 2016. *****

15. Python: matplotlib.

    Reading (before class): ***PRFbook***

16. Python: Quiz. **Quiz #1**

17. Getting started with MATLAB.

    Read: "Preface" of the *DAMbook* (Kimbrough, 2014a, pages xi–xiii).

    Today we will draw upon chapter 1 of the *DAMbook*, "Quick Start in MATLAB".

    Read: *MATLAB® Primer,* chapter 1, "Quick Start" (MathWorks, 2014c).

18. User-defined functions and flow control.

    Read: Chapter 3 of the *DAMbook*, "User-Defined Functions" and "Scripts and Functions" in *MATLAB® Primer,* (MathWorks, 2014c, chapter 5, "Programming").

Read: Chapter 4 of the *DAMbook*, "Control Flow". Recommended: (Davis, 2011, chapter 7) and "Control Flow" in *MATLAB® Primer, R2012b,* (MathWorks, 2014c, chapter 5, "Programming").

Recommended reading: Chapters "Simple Knapsack Models" and "Evolutionary Algorithms" in the *DAMbook*.

19. Arrays and operations on them, 1.

    Read: Chapter 5 of the *DAMbook*, "Arrays and Operations on Them, 1"; (Davis, 2011, chapters 3–5). Recommended: "Matrices and Arrays" and "Array Indexing" in *MATLAB® Primer,* (MathWorks, 2014c, chapter 1, "Quick Start").

20. Arrays and operations on them, 2.

    Read: Chapter 6 of the *DAMbook*, "Arrays and Operations on Them, 2;" (Davis, 2011, chapters 3–5). Recommended for those interested (not required for the course): "Language Fundamentals" and "Linear Algebra" in *MATLAB® Primer,* (MathWorks, 2014c, chapter 2; chapter 3).

21. Beyond Numeric Arrays and I/O.

    Read: Chapters 7 and 8 of the *DAMbook*, "Beyond Numeric Arrays" and "I/O: Interacting with Files." Recommended: (Davis, 2011, chapter 8). For reference purposes: (MathWorks, 2014a).

22. Cell Arrays, Etc., and I/O.

    Read: Handouts from the *DAMbook*. For reference purposes: (MathWorks, 2014d).

23. Q&A for Quiz #2. Discussion of small group projects. MATLAB assignment handed out and discussed.

    Python case due.

24. Quiz #2.

    Quiz #2

25. MATLAB: Using the toolboxes 1.

26. MATLAB: Using the toolboxes 2.

27. Python: NumPy.

    Read in *PRFbook* chapter 17.

28. Last day of class. Optional student presentations and/or presentation on special topics.

Small group assignment hand-ins due: 5 p.m. Sunday, May 1, 2016.

Small group hand-ins.

MATLAB case assignment due: 5 p.m. Wednesday, May 4, 2016.

MATLAB case due.

5 Calendar, Spring 2016

Reading days: April 28–9, 2016. Final examinations: May 2–10, 2016.

	0	1	2
0	—	W: 2016-02-17	W: 2016-03-30
1	W: 2016-01-13	M: 2016-02-22	M: 2016-04-04
2	W: 2016-01-20	W: 2016-02-24	W: 2016-04-06
3	M: 2016-01-25	M: 2016-02-29	M: 2016-04-11
4	W: 2016-01-27	W: 2016-03-02	W: 2016-04-13
5	M: 2016-02-01	M: 2016-03-14	M: 2016-04-18
6	W: 2016-02-03	W: 2016-03-16	W: 2016-04-20
7	M: 2016-02-08	M: 2016-03-21	M: 2016-04-25
8	W: 2016-02-10	W: 2016-03-23	W: 2016-04-27
9	M: 2016-02-15	M: 2016-03-28	—

Table 1: Class number :: date correlation, for Monday (M) and Wednesday (W) classes, spring 2016.

References

Ayres, I. (2007). *Super Crunchers: Why Thinking-by-Numbers Is the New Way to Be Smart*. Bantam Books, New York, NY.

Beazley, D. M. (2009). *Python: Essential Reference*. Addison-Wesley, Upper Saddle River, NJ, fourth edition.

Bird, S., Klein, E., and Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly, Sebastopol, CA.

Davenport, T. H. and Harris, J. G. (2007). *Competing on Analytics: The New Science of Winning*. Harvard Business School Press, Boston, MA.

Davis, T. A. (2011). *MATLAB® Primer*. CRC Press, Boca Raton, eighth edition.

Franks, B. (2012). *Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics*. John Wiley & Sons, Hoboken, NJ.

Friedl, J. E. (2006). *Mastering Regular Expressions*. O'Reilly & Associates, Inc., Sebastopol, California, third edition.

Kimbrough, S. O. (2014a). *An Introduction to Business Analytics with MATLAB®: Exploratory Computation and Business Analytics*. Draft manuscript. File: DAMbook-master.pdf. In process and changeable.

Kimbrough, S. O. (2014b). *Python Real Fast*. Draft manuscript.

Kiron, D. and Shockley, R. (2011). Creating business value with analytics. *MIT Sloan Management Review*, 53(1):56–64.

Kreibich, J. A. (2010). *Using SQLite: Small. Fast. Reliable. Choose Any Three.* O'Reilly, Sebastopol, CA.

Lohr, S. (2012). The age of big data. *The New York Times*.

MathWorks (2014a). *MATLAB® Data Import and Export, R2014b*. The MathWorks, Inc., Natick, MA. File: import_export.pdf.

MathWorks (2014b). *MATLAB® Graphics, R2014b*. The MathWorks, Inc., Natick, MA. File: graphg.pdf.

MathWorks (2014c). *MATLAB® Primer, R2014b*. The MathWorks, Inc., Natick, MA. File: getstart.pdf.

MathWorks (2014d). *MATLAB® Programming Fundamentals, R2014b*. The MathWorks, Inc., Natick, MA. File: matlab_prog.pdf.

McKinsey & Company (2012). Big data: The next frontier for competition. `http://www.mckinsey.com/Features/Big_Data` Accessed: 2012-12-25.

Silver, N. (2012). *The Signal and the Noise: Why So Many Predictions Fail—but Some Don't*. The Penguin Press, New York, NY.

Summerfield, M. (2009). *Programming in Python 3: A Complete Introduction to the Python Language*. Addison-Wesley, Upper Saddle River, NJ.

Sweigart, A. (2015). *Automate the Boring Stuff with Python: Practical Programming for Total Beginners*. No Starch Press.