JOURNAL OF CHEMICAL EDUCATION

pubs.acs.org/jchemeduc

Technology Report

Series of Jupyter Notebooks Using Python for an Analytical Chemistry Course

Erik J. Menke*

Cite This: J. Chem. Educ. 2020, 97, 3899−3903

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

ABSTRACT: UC Merced's upper-division analytical chemistry course has been modified to include a series of Jupyter notebooks intended to introduce chemistry students to the Python computer language. These Jupyter notebooks were designed to cover a wide variety of topics common to quantitative and instrumental analysis. Assuming no prior programming experience, the notebooks are scaffolded to introduce students the Python's syntax while applying Python to solving chemical problems. Jupyter and Python were chosen because they are free, widely available, and widely used.

KEYWORDS: *Upper-Division Undergraduate, Analytical Chemistry, Curriculum, Computer-based Learning*

Downloaded via 73.87.14.31 on November 10, 2020 at 15:31:56 (UTC).
See https://pubs.acs.org/sharingguidelines for options on how to legitimately share published articles.

INTRODUCTION

An important skill for all chemists is the ability to process, analyze, and visualize data. An important tool that helps with this are computers, and the ability to efficiently carry out this work on a computer is a critical skill for current and future chemists.[1] Although spreadsheets can help with some of these tasks, there are many problems, such as analyzing very large data sets, multidimensional analysis, and image analysis, that require more advanced tools and knowledge to solve. To help undergraduate chemistry students learn how to use computers to model and solve chemistry problems, a series of nine Jupyter notebooks, based on Python 3, have been developed that address various topics in UC Merced's analytical chemistry course. Python 3 inside Jupyter notebooks was chosen for a few reasons. The first was that both Jupyter and Python are open-source, with readily available distributions for all common operating systems.[2−7] The second is that Python is one of the most popular programming languages in the world, in particular for data analysis,[1,3,8] with a massive amount of online resources. In addition, it is becoming more and more popular within the science community, with numerous examples of Python being used for simulations and modeling,[2,7,9−15] including specifically for chemistry-related problems[1,11,12,16−21] The third is that Python is both simple and flexible, with straightforward syntax, making it relatively easy to read and learn the code.[6,7]

OBJECTIVES

UC Merced's analytical chemistry course is an upper-division course that covers topics traditionally covered in both quantitative analysis courses and instrumental analysis courses. These topics range from EDTA titrations to equilibrium calculations using activities to NMR spectroscopy to capillary zone electrophoresis. Because the course focuses on topic breadth, rather than depth, it was important to develop Jupyter notebooks that also covered a wide range of topics. Furthermore, although a programming course is a requirement of UC Merced's chemistry major, not all students take a programming course prior to analytical chemistry, and many take a language other than Python. Therefore, the Jupyter notebooks start with the assumption that the students have no prior experience with Python and starts by introducing the students to the language. With these requirements in mind, these Jupyter notebooks were developed to meet the following objectives:

1. Make students develop algorithmic thinking skills.
2. Teach students how and why computer programming is a useful tool for solving analytical chemistry problems.
3. Familiarize students with Python 3 syntax and online resources for solving problems with Python.
4. Improve student attitudes on using computers to solve chemistry problems.

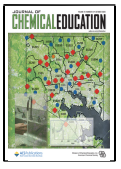
STRUCTURE AND CONTENT

Ultimately, the desire for this computational work was to serve as an effective addition to our current analytical chemistry course rather than as a stand-alone course. To that end, the nine Jupyter notebooks were created with two goals in mind. The first was to start with fairly simple Python commands and build up to more complex computations while meeting the objectives listed above. The second was to complement the material in the class. Student versions of the notebooks are available at https://github.com/erik-menke/AnalyticalProjects and are summarized below.

Notebook 1: Calibration Curves

The first notebook provided to the students focuses on introducing students to Jupyter and Python while teaching students how to construct calibration curves. Specifically,

Received: December 9, 2019
Revised: July 15, 2020
Published: September 1, 2020



ACS Publications

© 2020 American Chemical Society and Division of Chemical Education, Inc.

https://dx.doi.org/10.1021/acs.jchemed.9b01131
J. Chem. Educ. 2020, 97, 3899−3903

students are shown how to create lists, how to plot lists, and how to do a linear and nonlinear fit to the points. The students are then tasked with finding a concentration from a standard addition.

Notebook 2: Solubility

This notebook introduces students to solving linear and nonlinear sets of equations. The motivation for the notebook is to calculate K_{sp} for AgCl, first on paper using an ICE table and then by writing a function to solve a quadratic function. Then the student is introduced to multiple equilibria, shown how to solve a set of two equations for two unknowns, and then asked to modify that approach to solve three equations for three unknowns.

Notebook 3: Statistical Analysis

This notebook covers importing files and analyzing large data sets with the Pandas Python package. Specifically, the students are tasked with importing a .csv file that contains 22 years of water quality data collected at Elkhorn Slough (starting April 1, 1996), an estuary in Monterey Bay. The data file was requested through the NOAA National Estuarine Research Reserve (NERR) Centralized Data Management Office and contains over 618,000 measurements on the temperature, specific conductivity, salinity, amount of dissolved oxygen, and pH of the water in the estuary. A similar, up-to-date data file can be downloaded from the NERR Centralized Data Management Office (https://cdmo.baruch.sc.edu/dges/). The students were shown how to import a file into a Pandas DataFrame, how to calculate some statistical analysis on the data, and how to use the time and date functionality of Pandas to calculate and visualize changes over time. The students then were tasked with graphing the monthly average pH in the estuary from for 1997, comparing this with the monthly average pH in 2007, and explaining why the pH values may differ.

Notebook 4: Activity Coefficients

This notebook is used to calculate the solubility of Hg_2Cl_2 in an aqueous solution of KNO_3, as a function of KNO_3 concentration. The students are asked to calculate the solubility of Hg_2Cl_2 in pure water, given the Debye−Hückel equation, asked to use the Debye−Hückel function to calculate activity coefficients for the Hg_2^{2+} and Cl^- ions, and then asked to graph the solubility of Hg_2Cl_2 as a function of KNO_3.

Notebook 5: Titration Curves

This notebook is used to calculate the expected titration curve for a monoprotic weak acid titrated with a strong base. The students are shown how to use material introduced in Notebook 2 to calculate the initial pH of a 51.3 mL solution of 0.131 M acetic acid and the pH of the solution as 0.0953 M NaOH is added, up to the half-equivalence point. The students are then asked to calculate the pH of the solution from the half-equivalence point to the equivalence point and then from the equivalence point up to the addition of 150 mL of 0.0953 M NaOH and then to graph the overall titration curve.

Notebook 6: EDTA

This notebook is used to calculate the $\alpha 1$, $\alpha 2$, $\alpha 3$, $\alpha 4$, and $\alpha 5$ values for EDTA in an aqueous solution from pH 0 to 14 and then to graph them all together. This is the first notebook where the students are given a series of computational tasks with minimal guidance and none of the computations already performed, with the expectation that they will use what they have learned in previous notebooks to complete the task.

Notebook 7: FTIR Spectra

This notebook introduces students to Fourier transform infrared (FTIR) spectroscopy by having the students import an interferogram collected from an FTIR and then transform it into an IR spectrum. The notebook first walks students through the process of importing the background interferogram, processing the interferogram, and then transforming it into a spectrum of intensity versus wavenumber. The students are then tasked with turning the data interferogram into a data spectrum of intensity versus wavenumber and then combining the two spectra to create a spectrum of $\%T$ versus wavenumber. Text files containing the data for the background and spectrum interferograms are provided to the students.

Notebook 8: NMR Spectra

This notebook introduces students to NMR spectroscopy, as well as how to add packages to Python. This notebook starts by having the students add the nmrglue package, an open-source package for reading proprietary NMR files and analyzing NMR spectra, to their Python environment. Then, similar to the procedure in Notebook 7, it walks the students through the process of importing an Agilent NMR data file, processing the FID, and then transforming the FID into a spectrum. The students are then tasked with turning a second Agilent NMR data file into a spectrum and discussing the differences between the two spectra. Binary files containing the FID and instrumental parameters for the two spectra (a porphyrin with and without D_2O) are provided to the students.

Notebook 9: Distillation

This notebook provides the students further practice with writing functions by having the students calculate the expected distillation phase diagram for the benzene−toluene system, with the assumption that this system behaves ideally. Similar to Notebook 6, this notebook gives the students a series of computational tasks with none of the computations already performed. This notebook starts by having the students write three functions. The first two are different variations of the Clausius−Clapeyron equation, one that returns a temperature and another that returns a pressure. The third function is an implementation of Raoult's law to calculate partial pressure. The students then work through a series of calculations and finally graph the expected distillation phase diagram.

■ IMPLEMENTATION AND ASSESSMENT

One of the baseline assumptions while developing and implementing these notebooks was that the students had little-to-no prior experience with Python. In addition, due to the resources available for the course, the students would need to work through the notebooks on their own computers but were unlikely to have Python and Jupyter installed on their computers. Therefore, prior to having the students work on the notebooks, a lecture was set aside to introduce students to the Python language and show them where to find helpful online resources.

Prior to this introductory lecture, the students were asked to download and install Anaconda, a free, open-source distribution of Python and Jupyter. During the introductory lecture on Python, the instructor spent about 20 min introducing the students to simple computational concepts and Python commands (such as basic arithmetic, variables, the print() function, etc.) as well as resources (python.org, using Google to answer specific questions, books in the library[22−27]) that might

https://dx.doi.org/10.1021/acs.jchemed.9b01131
J. Chem. Educ. 2020, 97, 3899−3903

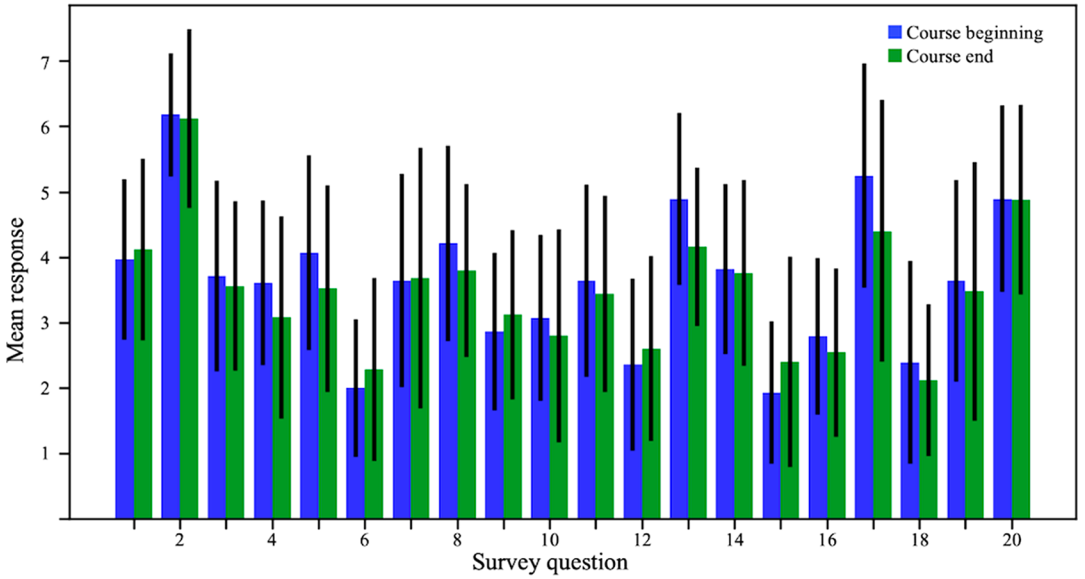


Figure 1. Survey of attitudes on using computers for chemistry showing mean responses to both pre- (blue, $N = 28$) and post- (green, $N = 25$) coursework, with black error bars showing 1 standard deviation. Individual questions are shown in Table 1.

be helpful and then asking the students to solve simple arithmetic problems while walking around the room to help students with either any Anaconda installation issues or coding questions.

In the following lecture, the students worked in groups using Excel to create calibration curves, similar to what they would be expected to do in the first notebook. The idea for this was that students had prior experience (in general chemistry, for example) working with Excel, so that questions or issues would be primarily about the calibration curves rather than implementation. Finally, in the third lecture in this series, the students worked through the first Jupyter notebook in small groups of 3 to 4 students so that those students with more experience could help those students with less experience while the instructor moved through the class helping answer questions. Finally, approximately 1 week later, the students, again in groups of 3 to 4, worked through Notebook 2 during class. All further notebooks were assigned as homework for the students, with due dates approximately 1 week after the relevant material had been introduced.

Objectives 1 (make students develop algorithmic thinking skills), 2 (teach students how and why computer programming is a useful tool for solving analytical chemistry problems), and 3 (familiarize students with Python 3 syntax and online resources for solving problems with Python) were primarily evaluated via summative assessment. The Jupyter notebooks made up 15% of the course grade, and the instructor compared the student notebooks with a notebook completed by the instructor. Each notebook was worth 10 points, with students receiving 5 points for submitting a notebook that has code but does not actually run, 7 points for submitting a notebook that runs but does not result in correct results, and 10 points for submitting a notebook that runs and yields the correct results. The majority of the students in the course received full credit for all of their notebooks, and those that did not receive full credit for all of their notebooks did receive full credit for their last two notebooks, suggesting that the students were at least making strides toward objectives 1, 2, and 3.

In addition to developing algorithmic thinking, understanding how computers are useful tools and becoming familiar with Python syntax and resources, the fourth goal for introducing this material to the class was to improve student attitudes toward using computers for analyzing and modeling chemistry. To test for changes in student attitude, a modified version of the Attitude toward the Subject of Chemistry Inventory (ASCI),[28] provided in the Supporting Information, was given anonymously on the first day of class (prior to any discussion of using computers) and in the last week of class, after all the Jupyter assignments had been graded. The results for the pre- and postwork surveys are shown in Figure 1, and the means from the survey, as well as effect sizes, are in Table 1.

As Figure 1 shows, over the course of the semester, there was little change in attitude toward using computers for chemistry. However, when we examine the effect sizes for each question, where the effect size is calculated as the difference of the means divided by the standard deviation,[29] we can see small effects (defined as an effect size between 0.20 and 0.50) for 10 questions and a medium effect (defined as an effect size between 0.50 and 0.80) for one question. Whereas some of these effects are positive (question 6 and 12, for example) and others negative (questions 4 and 8, for example), all of the change is toward a more negative attitude. For example, question 4 shows a shift from simple to complicated, whereas question 6 shows a shift from good to bad. Surprisingly, the largest effect is seen for question 13, with a shift from attractive to disgusting.

Because these shifts are small, however, it is impossible to say unequivocally that they are due to changing attitudes resulting from the Jupyter exercises. Furthermore, due to changes in course enrollment after the first week and differences in attendance during the pre- and post-survey administration, there were differences in the surveyed population. Finally, whereas there was a shift toward more negative attitudes, overall both the pre- and post-survey populations held primarily positive attitudes toward using computers for chemistry, in particular, questions 2 (computers are beneficial), 15 (computers are worthwhile), and 18 (computers are safe).

https://dx.doi.org/10.1021/acs.jchemed.9b01131
J. Chem. Educ. 2020, 97, 3899−3903

Table 1. Comparison of Survey Question Pre- and Post-semester Means and Effect Sizes

word pairs for response to this prompt: using computers for chemistry is...		students' mean response values[a]		
first word in pair [1]	second word in pair [7]	pre-semester (N = 28)	post-semester (N = 25)	effect size[b]
easy	hard	4.0	4.1	0.1
worthless	beneficial	6.2	6.1	0.0
exciting	boring	3.7	3.6	−0.1
complicated	simple	3.6	3.1	−0.3
confusing	clear	4.1	3.5	−0.3
good	bad	2.0	2.3	0.2
satisfying	frustrating	3.6	3.7	0.0
scary	fun	4.2	3.8	−0.3
comprehensible	incomprehensible	2.9	3.1	0.2
challenging	not challenging	3.1	2.8	−0.2
pleasant	unpleasant	3.6	3.4	−0.1
interesting	dull	2.4	2.6	0.2
disgusting	attractive	4.9	4.2	−0.6
comfortable	uncomfortable	3.8	3.8	0.0
worthwhile	useless	1.9	2.4	0.3
work	play	2.8	2.5	−0.2
chaotic	organized	5.3	4.4	−0.4
safe	dangerous	2.4	2.1	−0.2
tense	relaxed	3.6	3.5	−0.1
insecure	secure	4.9	4.9	0.0

[a]The scale for the survey questions has a range of 1−7, with 1 being the first word of the pair, 7 being the second word of the pair, and the middle position indicating that the respondent was undecided or had no feelings related to those terms. [b]Calculated as the difference of the means divided by the standard deviation. An effect size between 0.20 and 0.50 is considered small; an effect size between 0.50 and 0.80 is considered medium; an effect size >0.80 is considered large. See ref 29.

SUMMARY

Nine Jupyter notebooks have been developed to supplement an undergraduate analytical course, covering topics ranging from solubility and equilibrium to processing spectra. The notebooks have been scaffolded so that students are given more computationally complex tasks as the course progresses. The students' attitudes toward computers and computation for chemistry were overall very positive, although their attitude shifted slightly more negative over the course of the semester.

ASSOCIATED CONTENT

Supporting Information

Student versions of the notebooks, text files containing the background and sample IR interferogram data for Notebook 7, and binary files containing the FID and parameter data for Notebook 8 are available at https://github.com/erik-menke/AnalyticalProjects. Completed notebooks with solutions for instructors can be requested by emailing the author. The Supporting Information is available at https://pubs.acs.org/doi/10.1021/acs.jchemed.9b01131.

Static version of the revised Attitude toward the Subject of Chemistry Inventory (PDF)

AUTHOR INFORMATION

Corresponding Author

Erik J. Menke − *Department of Chemistry and Chemical Biology, University of California, Merced, Merced, California 95343, United States;* orcid.org/0000-0002-7819-2167; Email: emenke@ucmerced.edu

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jchemed.9b01131

Notes

The author declares no competing financial interest.

ACKNOWLEDGMENTS

The author thanks the students for the feedback, hard work, and enthusiasm for trying something new and different in this course.

REFERENCES

(1) Weiss, C. J. Scientific Computing for Chemists: An Undergraduate Course in Simulations, Data Processing, and Visualization. *J. Chem. Educ.* **2017**, *94* (5), 592−597.

(2) Hill, C. *Learning Scientific Programming with Python*; Cambridge University Press: Cambridge, UK, 2016.

(3) McKinney, W. *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and Ipython*, 2nd ed.; O'Reilly Media: Sebastopol, CA, 2018

(4) McKinney, W. Data Structures for Statistical Computing in Python. In *Proceedings of the 9th Python in Science Conference*; van der Walt, S., Millman, J., Eds.; Austin, TX, 2010; p 51.

(5) Pérez, P.; Granger, B. E. IPython: A System for Interactive Scientific Computing. *Comput. Sci. Eng.* **2007**, *9* (3), 21.

(6) Perkel, J. M. Pick Up Python. *Nature* **2015**, *518*, 125.

(7) Rossant, C. *Learning IPython for Interactive Computing and Data Visualization*; Packt Publishing: Birmingham, UK, 2013.

(8) Nelli, F. *Python Data Analytics*; Apress: Berkeley, CA, 2015.

(9) Beddard, G. S. Solution of the Schrödinger equation for one-dimensional anharmonic potentials: an undergraduate computational experiment. *J. Chem. Educ.* **2011**, *88* (7), 929.

(10) Hunter, J. D. Matplotlib: A 2D Graphics Environment. *Comput. Sci. Eng.* **2007**, *9* (3), 90.

(11) Srnec, M. N.; Upadhyay, S.; Madura, J. D. Teaching Reciprocal Space to Undergraduates via Theory and Code Components of an IPython Notebook. *J. Chem. Educ.* **2016**, *93* (12), 2106.

(12) Srnec, M. N.; Upadhyay, S.; Madura, J. D. A Python Program for Solving Schrödinger's Equation in Undergraduate Physical Chemistry. *J. Chem. Educ.* **2017**, *94* (6), 813−815.

(13) Van Der Walt, S.; Colbert, S. C.; Varoquaux, G. The NumPy Array: A Structure for Efficient Numerical Computation. *Comput. Sci. Eng.* **2011**, *13* (2), 22.

(14) Van der Walt, S.; Schönberger, J. L.; Nunez-Iglesias, J.; Boulogne, F.; Warner, J. D.; Yager, N.; Gouillart, E.; Yu, T. Scikit-Image: Image Processing in Python. *PeerJ* **2014**, *2*, e453.

(15) Cruzeiro, V. W. D.; Gao, X.; Kleiman, V. D. Implementing New Educational Platforms in the Classroom: An Interactive Approach to the Particle in a Box Problem. *J. Chem. Educ.* **2019**, *96* (8), 1663−1670.

(16) Green, M.; Chen, X. Data Functionalization for Gas Chromatography in Python. *J. Chem. Educ.* **2020**, *97* (4), 1172−1175.

(17) Tan, S. W. B.; Naraharisetti, P. K.; Chin, S. K.; Lee, L. Y. Simple Visual-Aided Automated Titration Using the Python Programming Language. *J. Chem. Educ.* **2020**, *97* (3), 850−854.

(18) Navarre, E. C. Extensible Interface for a Compact Spectrophotometer for Teaching Molecular Absorption in the Undergraduate Laboratory. *J. Chem. Educ.* **2020**, *97* (5), 1500−1503.

(19) Vargas, S.; Zamirpour, S.; Menon, S.; Rothman, A.; Häse, F.; Tamayo-Mendoza, T.; Romero, J.; Sim, S.; Menke, T.; Aspuru-Guzik, A. Team-Based Learning for Scientific Computing and Automated

Experimentation: Visualization of Colored Reactions. *J. Chem. Educ.* **2020**, *97* (3), 689−694.
(20) Weiss, C. J. Introduction to Stochastic Simulations for Chemical and Physical Processes: Principles and Applications. *J. Chem. Educ.* **2017**, *94* (12), 1904−1910.
(21) Smith, D. G. A.; Burns, L. A.; Sirianni, D. A.; Nascimento, D. R.; Kumar, A.; James, A. M.; Schriber, J. B.; Zhang, T.; Zhang, B.; Abbott, A. S.; et al. Psi4NumPy: An Interactive Quantum Chemistry Programming Environment for Reference Implementations and Rapid Development. *J. Chem. Theory Comput.* **2018**, *14* (7), 3504−3511.
(22) Lee, K. D. *Python Programming Fundamentals*; Springer: London, 2011.
(23) Walters, G. *The Python Quick Syntax Reference*; Apress: Berkeley, CA, 2014.
(24) Stewart, J. *Python for Scientists*, 2nd ed.; Cambridge University Press: Cambridge, UK, 2017.
(25) McKinney, W. *Python for Data Analysis*; O'Reilly Media: Sebastopol, CA, 2013.
(26) Nelli, F. *Python Data Analytics: With Pandas, NumPy, and Matplotlib*, 2nd ed.; Apress: New York, 2018.
(27) Boschetti, A.; Massaron, L. *Python Data Science Essentials: A Practitioner's Guide Covering Essential Data Science Principles, Tools, and Techniques*, 3rd ed.; Packt Publishing Ltd.: Birmingham, UK, 2018.
(28) Bauer, C. F. Attitude toward Chemistry: A Semantic Differential Instrument for Assessing Curriculum Impacts. *J. Chem. Educ.* **2008**, *85* (10), 1440.
(29) Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed.; L. Erlbaum Associates: Hillsdale, NJ, 1988.

https://dx.doi.org/10.1021/acs.jchemed.9b01131
J. Chem. Educ. 2020, 97, 3899−3903