Teaching With Jupyter In-Class Activities: Lessons Learned and Next Steps

David Anastasiu
San Jose State University, danastasiu@scu.edu

Follow this and additional works at: https://scholarworks.sjsu.edu/computer_eng_pub

Part of the Digital Communications and Networking Commons

Recommended Citation
Teaching With Jupyter In-Class Activities:
Lessons Learned and Next Steps
David C. Anastasiu

What is Jupyter Notebook?
- Interactive browser-based document that enables mixing rich text with mathematical equations, live data visualizations, and interactive execution of code.
- Popular with students and professionals alike from fields as diverse as Data Science, Sociology, Political Science, Physics, and Journalism.
- Supports more than 40 programming languages:
  - Python
  - MATLAB
  - Julia
  - LaTeX
  - Bash
  - SQL
  - R
  - JavaScript
  - PHP
  - Node
  - OCaml
  - F#
  - SQLite

- The benefits of using notebooks include:
  - Interactivity.
  - Analytical reproducibility.
  - Collaboration.
  - Ease of access to computing resources.
- Used by dozens of major companies and part of curriculum at many universities.

Example Jupyter Notebook Activities

CMPE 139: Database Systems I

End of semester survey given to students in 3 sections of CMPE 255 (Data Mining), over 2 semesters.
- Likert-scale survey with 10 categorical and 4 open answer questions.
- Averts response style bias by testing both positive and negative responses.
- Positive questions coded 1-5 and negative ones 5-1.
- 89 responses, evenly distributed between sections.

The Jupyter Notebook activities did not help clarify concepts in the lectures.

I found the ability to intersperse textual information with executable code in Jupyter Notebooks odd and useless.

I found it difficult to execute programs in the Jupyter Notebook environment.

The activities were too difficult and I could not finish them even if I spent the whole class time on them.

Lesson Learned from Student Feedback

End of semester survey given to students in 3 sections of CMPE 255 (Data Mining), over 2 semesters.
- Likert-scale survey with 10 categorical and 4 open answer questions.
- Averts response style bias by testing both positive and negative responses.
- Positive questions coded 1-5 and negative ones 5-1.
- 89 responses, evenly distributed between sections.

I understood material taught in class better because of the Jupyter Notebook activities.

It was helpful to have rich text descriptions of problems and concepts in the same page I was coding in.

Once installed, I found the Jupyter notebooks easy to use and intuitive.

Activities had a good degree of difficulty that kept me engaged.

What were the downsides of programming in Jupyter Notebooks? Why?
- I can not recall any downside.
- Syntax highlighting or syntax help like in other editors.

What were the most useful features of the Jupyter Notebook activities? Why?
- It helped us learn concepts better. Was a very good learning tool and very easy to use.
- [Executing] pieces of programs in real time [helps] break down complex material into understandable chunks.
- Concept followed by activity. Faster learning.
- It helped me understand the methods and algorithms mentioned in the slides in a practical way.

How, if at all, did you approach solving homework assignments for the class (or even other classes) in a different way after being exposed to Jupyter Notebook?
- Running step by step programs to make sure each part works well.
- [Getting] into the habit of writing descriptions along with the program.
- I would dissect the problems into a set of small problems, implement each of them instead of trying to solve the big problem as a whole.

What is one thing that could be improved in the use of Jupyter Notebook and/or in-class activities for this class?
- Should be more in-class with a little more time.
- Have a complete solution posted after the in-class activities are due.

In-Class Activities
- Designed to aid presentation of theoretical concepts, helping students learn through practice.
- Plan to include 1-2 Activities per class.
- Beginning of notebook introduces topic and ties in with lecture.
- Description of concepts is intermingled with demonstrations and short practical exercises.
- Students work in groups and are given 5-10 minutes to complete exercises.
- Each exercise is followed with in-class discussion analyzing proposed solutions by students in the class.

Take-Home Activities
- Some notebooks are assigned as homework assignments and provide additional opportunity for practice.
- Activities are not graded on correctness. Students are encouraged to work through activity problems.

Next Steps: Jupyter Hub, HPC, and JupyterLab

- Alleviate initial setup troubles + ensure identical setup.
- JupyterLab will soon replace Jupyter Notebook.
- Continue to add and improve activities.

Open Answer Questions and Example Answers

What were the most useful features of the Jupyter Notebook activities? Why?
- It helped us learn concepts better. Was a very good learning tool and very easy to use.
- [Executing] pieces of programs in real time [helps] break down complex material into understandable chunks.
- Concept followed by activity. Faster learning.
- It helped me understand the methods and algorithms mentioned in the slides in a practical way.

Lessions Learned From Student Feedback

End of semester survey given to students in 3 sections of CMPE 255 (Data Mining), over 2 semesters.
- Likert-scale survey with 10 categorical and 4 open answer questions.
- Averts response style bias by testing both positive and negative responses.
- Positive questions coded 1-5 and negative ones 5-1.
- 89 responses, evenly distributed between sections.

I understood material taught in class better because of the Jupyter Notebook activities.

It was helpful to have rich text descriptions of problems and concepts in the same page I was coding in.

Once installed, I found the Jupyter notebooks easy to use and intuitive.

Activities had a good degree of difficulty that kept me engaged.

What were the downsides of programming in Jupyter Notebooks? Why?
- I can not recall any downside.
- Syntax highlighting or syntax help like in other editors.

What were the most useful features of the Jupyter Notebook activities? Why?
- It helped us learn concepts better. Was a very good learning tool and very easy to use.
- [Executing] pieces of programs in real time [helps] break down complex material into understandable chunks.
- Concept followed by activity. Faster learning.
- It helped me understand the methods and algorithms mentioned in the slides in a practical way.

Next Steps: Jupyter Hub, HPC, and JupyterLab

- Alleviate initial setup troubles + ensure identical setup.

Open Answer Questions and Example Answers

What were the most useful features of the Jupyter Notebook activities? Why?
- It helped us learn concepts better. Was a very good learning tool and very easy to use.
- [Executing] pieces of programs in real time [helps] break down complex material into understandable chunks.
- Concept followed by activity. Faster learning.
- It helped me understand the methods and algorithms mentioned in the slides in a practical way.

Lessons Learned From Student Feedback

End of semester survey given to students in 3 sections of CMPE 255 (Data Mining), over 2 semesters.
- Likert-scale survey with 10 categorical and 4 open answer questions.
- Averts response style bias by testing both positive and negative responses.
- Positive questions coded 1-5 and negative ones 5-1.
- 89 responses, evenly distributed between sections.

I understood material taught in class better because of the Jupyter Notebook activities.

It was helpful to have rich text descriptions of problems and concepts in the same page I was coding in.

Once installed, I found the Jupyter notebooks easy to use and intuitive.

Activities had a good degree of difficulty that kept me engaged.

What were the downsides of programming in Jupyter Notebooks? Why?
- I can not recall any downside.
- Syntax highlighting or syntax help like in other editors.

What were the most useful features of the Jupyter Notebook activities? Why?
- It helped us learn concepts better. Was a very good learning tool and very easy to use.
- [Executing] pieces of programs in real time [helps] break down complex material into understandable chunks.
- Concept followed by activity. Faster learning.
- It helped me understand the methods and algorithms mentioned in the slides in a practical way.