Can Jupyter Notebooks Serve Two Masters?

Amanda Birmingham

Senior Bioinformatics Engineer

Center for Computational Biology & Bioinformatics, UCSD

Reproducible Research

- Repeatability & reproducibility are key to the scientific method
  - In 1663, only Robert Boyle and Christiaan Huygens could produce a vacuum—and their findings didn't agree
- Informatics *should* be at the forefront of reproducible research
  - Doing the same thing over and over is what computers do best!

Reproducible Research

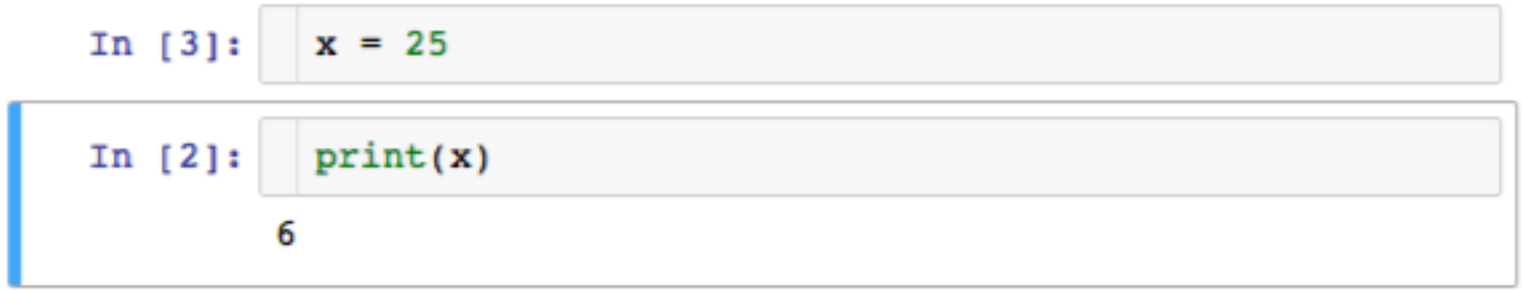
- Repeatability & reproducibility are key to the scientific method
  - In 1663, only Robert Boyle and Christiaan Huygens could produce a vacuum—and their findings didn't agree
- Informatics *should* be at the forefront of reproducible research
  - Doing the same thing over and over is what computers do best!
- Many see Jupyter Notebooks as the magic bullet for reproducibility
  - e.g., Data Carpentry offers an entire workshop on "Reproducible Research using Jupyter Notebooks"

Reproducible Research

- Repeatability & reproducibility are key to the scientific method
  - In 1663, only Robert Boyle and Christiaan Huygens could produce a vacuum—and their findings didn't agree
- Informatics *should* be at the forefront of reproducible research
  - Doing the same thing over and over is what computers do best!
- Many see Jupyter Notebooks as the magic bullet for reproducibility
  - e.g., Data Carpentry offers an entire workshop on "Reproducible Research using Jupyter Notebooks"
- But Jupyter Notebooks were explicitly designed to be **interactive**

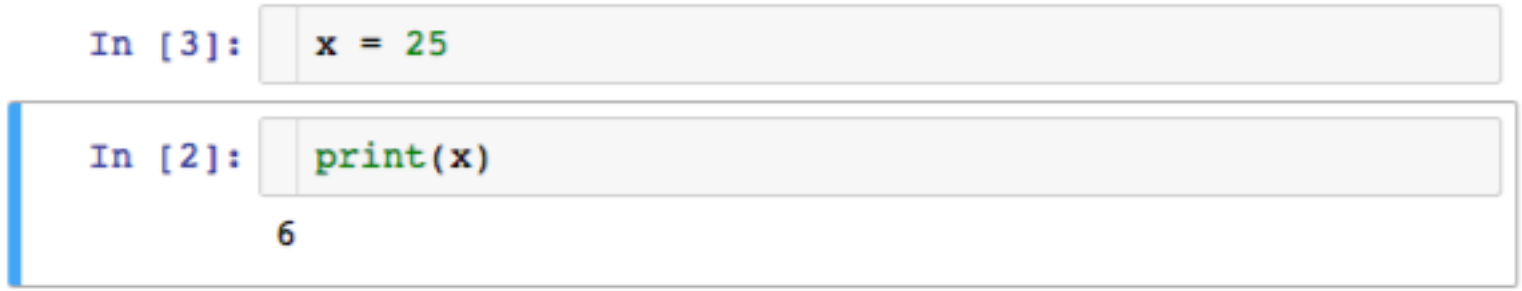
“No Man Can Serve Two Masters”

- Jupyter Notebooks’ interactivity directly compromises their reproducibility
  - Cells can be executed out of order, which is inconsistent with end-to-end rerunning
  - Changing code/variables in a notebook does NOT rerun cells that depend on that change
    - In fact, it doesn’t even clear old outputs!



“No Man Can Serve Two Masters”

- Jupyter Notebooks’ interactivity directly compromises their reproducibility
  - Cells can be executed out of order, which is inconsistent with end-to-end rerunning
  - Changing code/variables in a notebook does NOT rerun cells that depend on that change
    - In fact, it doesn’t even clear old outputs!



- Currently, users often sacrifice one goal or the other
  - Enforce reproducibility with notebook scripting, run-only cells extension
  - Embrace interactivity in “draft” notebooks, redo for reproducibility once best path through data known

… Or Can He?

- Maybe there aren't two opposing goals, but one synthesizing goal
  - Interactivity is critical to exploratory data analysis
  - Reproducibility is central to the scientific method
  - Thus, the real goal is: **make interactive data exploration reproducible**

… Or Can He?

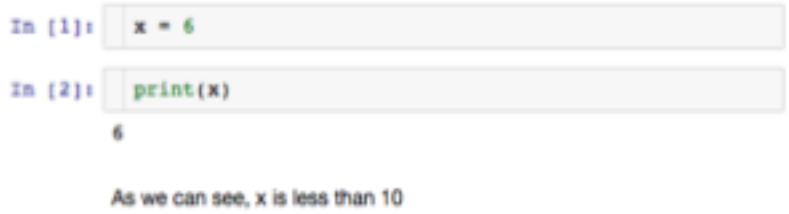
- Maybe there aren't two opposing goals, but one synthesizing goal
  - Interactivity is critical to exploratory data analysis
  - Reproducibility is central to the scientific method
  - Thus, the real goal is: **make interactive data exploration reproducible**
- Sadly, Jupyter Notebooks don't do that now
  - **But they might be able to**, with improvements!

… Or Can He?

- Maybe there aren't two opposing goals, but one synthesizing goal
  - Interactivity is critical to exploratory data analysis
  - Reproducibility is central to the scientific method
  - Thus, the real goal is: **make interactive data exploration reproducible**
- Sadly, Jupyter Notebooks don't do that now
  - **But they might be able to**, with improvements!
- Some ideas:
  - Highlight and/or clear cells "downstream" of a code change
  - Expand "undo" capability
  - Improve integration with version control systems
  - Produce record of all steps, in order run, with output and comments
    - Include re-runs as separate items
    - IPython logging (%logstart with -o) gets only part-way there, and is python-specific ☹

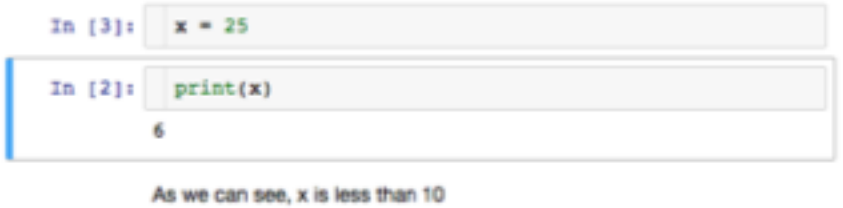
… Or Can He?

- Maybe there aren't two opposing goals, but one synthesizing goal
  - Interactivity is critical to exploratory data analysis
  - Reproducibility is central to the scientific method
  - Thus, the real goal is: **make interactive data exploration reproducible**
- Sadly, Jupyter Notebooks don't do that now
  - **But they might be able to**, with improvements!
- Some ideas:
  - Highlight and/or clear cells "downstream" of a code change
  - Expand "undo" capability
  - Improve integration with version control systems
  - Produce record of all steps, in order run, with output and comments
    - Include re-runs as separate items
    - IPython logging (%logstart with -o) gets only part-way there, and is python-specific ☹



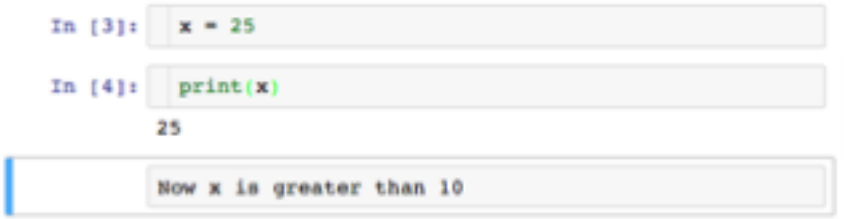
… Or Can He?

- Maybe there aren't two opposing goals, but one synthesizing goal
  - Interactivity is critical to exploratory data analysis
  - Reproducibility is central to the scientific method
  - Thus, the real goal is: **make interactive data exploration reproducible**
- Sadly, Jupyter Notebooks don't do that now
  - **But they might be able to**, with improvements!
- Some ideas:
  - Highlight and/or clear cells "downstream" of a code change
  - Expand "undo" capability
  - Improve integration with version control systems
  - Produce record of all steps, in order run, with output and comments
    - Include re-runs as separate items
    - IPython logging (%logstart with -o) gets only part-way there, and is python-specific ☹



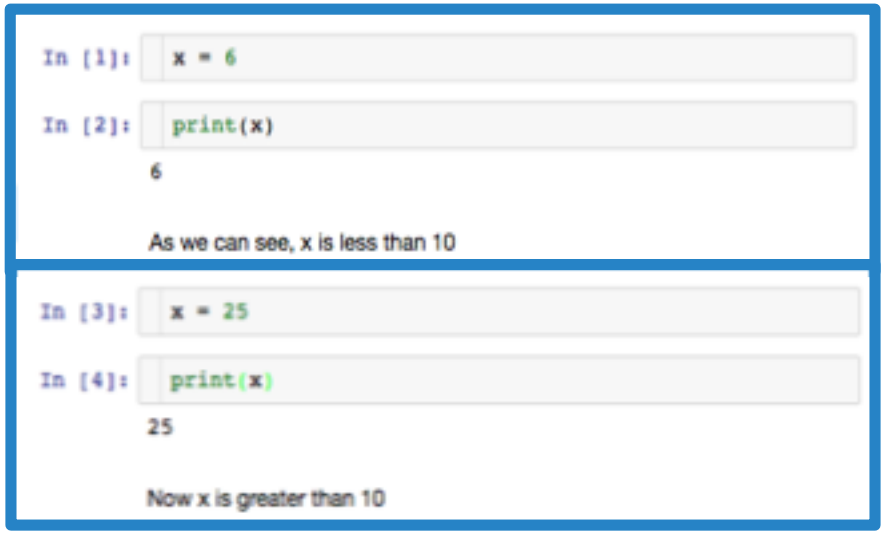
… Or Can He?

- Maybe there aren't two opposing goals, but one synthesizing goal
  - Interactivity is critical to exploratory data analysis
  - Reproducibility is central to the scientific method
  - Thus, the real goal is: **make interactive data exploration reproducible**
- Sadly, Jupyter Notebooks don't do that now
  - **But they might be able to**, with improvements!
- Some ideas:
  - Highlight and/or clear cells "downstream" of a code change
  - Expand "undo" capability
  - Improve integration with version control systems
  - Produce record of all steps, in order run, with output and comments
    - Include re-runs as separate items
    - IPython logging (%logstart with -o) gets only part-way there, and is python-specific ☹



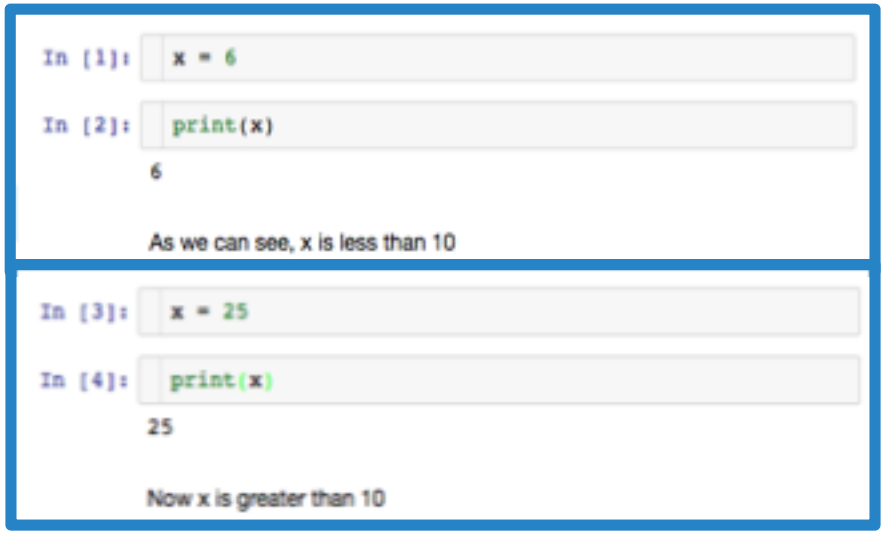
… Or Can He?

- Maybe there aren't two opposing goals, but one synthesizing goal
  - Interactivity is critical to exploratory data analysis
  - Reproducibility is central to the scientific method
  - Thus, the real goal is: **make interactive data exploration reproducible**
- Sadly, Jupyter Notebooks don't do that now
  - **But they might be able to**, with improvements!
- Some ideas:
  - Highlight and/or clear cells "downstream" of a code change
  - Expand "undo" capability
  - Improve integration with version control systems
  - Produce record of all steps, in order run, with output and comments
    - Include re-runs as separate items
    - IPython logging (%logstart with -o) gets only part-way there, and is python-specific ☹



… Or Can He?

- Maybe there aren't two opposing goals, but one synthesizing goal
  - Interactivity is critical to exploratory data analysis
  - Reproducibility is central to the scientific method
  - Thus, the real goal is: **make interactive data exploration reproducible**
- Sadly, Jupyter Notebooks don't do that now
  - **But they might be able to**, with improvements!
- Some ideas:
  - Highlight and/or clear cells "downstream" of a code change
  - Expand "undo" capability
  - Improve integration with version control systems
  - Produce record of all steps, in order run, with output and comments
    - Include re-runs as separate items
    - IPython logging (%logstart with -o) gets only part-way there, and is python-specific ☹
- Science is hard, so record-keeping should be easy!



Acknowledgments

- Fernando Perez & the Jupyter Project!
- CCBB Team
  - Katie Fisch (Director)
- Our funders
  - UC San Diego Health Sciences
  - CTRI Center for Accelerating Drug Development (CADD) – Grant UL1TR001442