

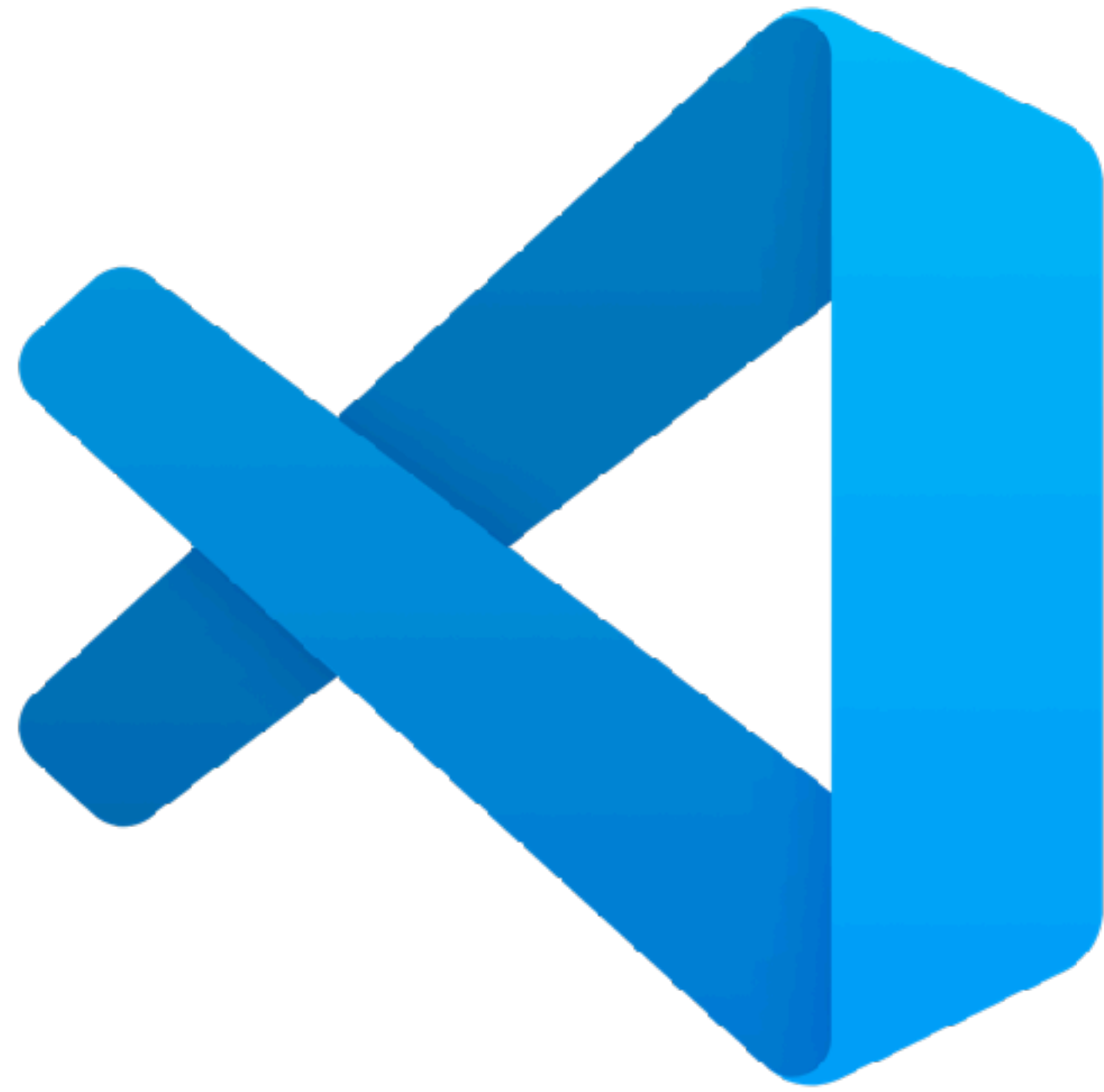
AST1501 - Introduction to Research

Jo Bovy

Using coding tools

Using coding tools

- In pretty much any field of astrophysics, you'll be doing *a lot* of coding
- While significant intellectual effort goes into developing codes, coding involves many less intellectual tasks
- If you can use tools to make those tasks easier, use them!



Code editors

Use a good code editor!

- Hard to overstate the importance of using a good code editor
- And really learning how to use it well
- VS Code generally considered to be the best currently (but can change quickly...)
- Generally, good code editors will do things like:
 - Auto-save
 - Keep track of indentation
 - Advanced search and replace tools
 - ...



VS Code

- VS Code is a particularly useful code editor with many useful features
- Many of these are built in, but many others are the result of an extensive system of *extensions*
- Non-exhaustive list of things you can do easily
 - Edit, display, and run Jupyter Notebooks in different conda environments
 - Preview Markdown documents
 - View PDFs
 - Debugging
 - Use Docker
 - Search entire codebases or sub-directories
 - Work on a remote server

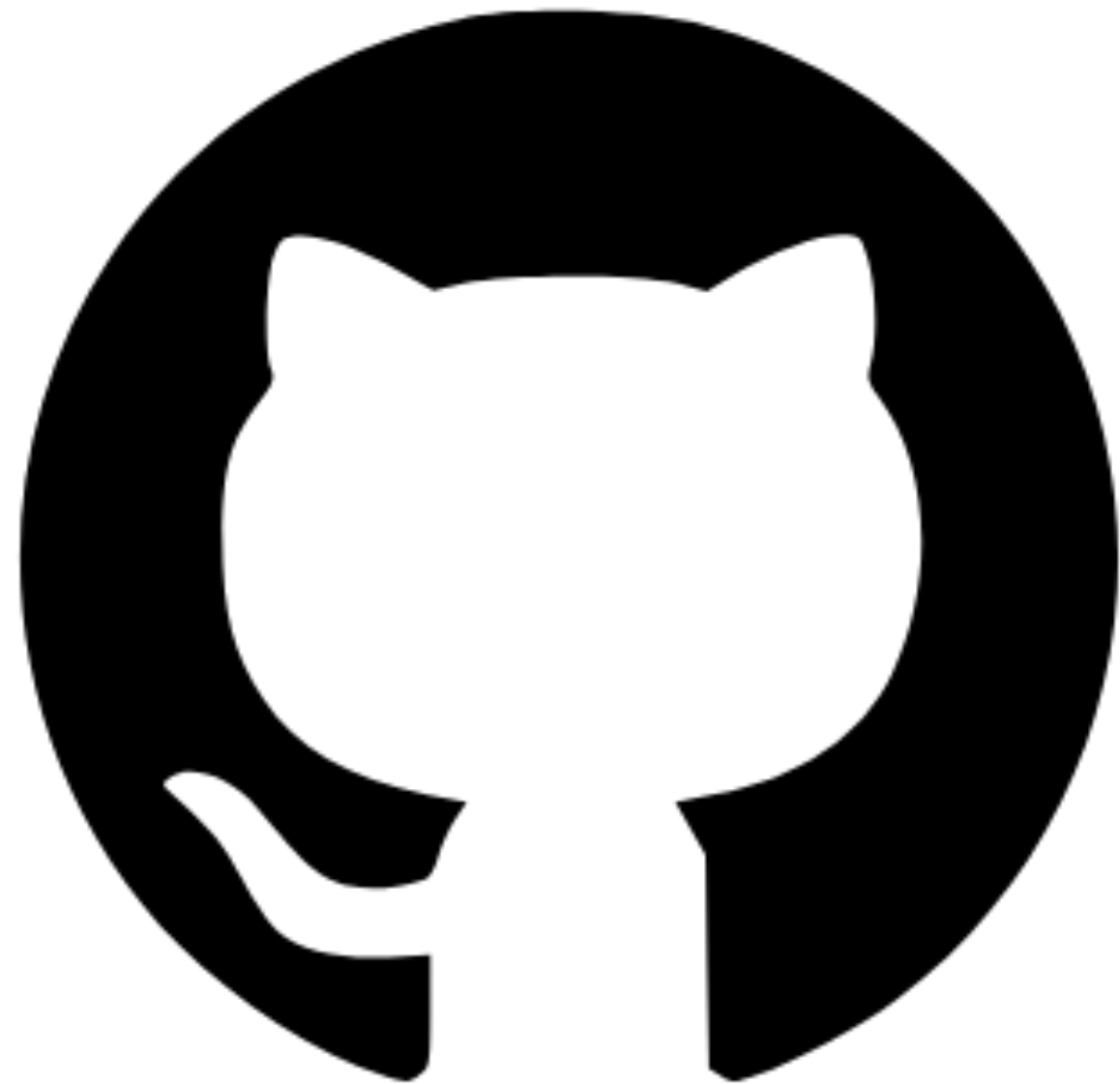


VS Code IntelliSense

- IntelliSense: from the docs

“general term for various code editing features including: code completion, parameter info, quick info, and member lists. IntelliSense features are sometimes called by other names such as "code completion", "content assist", and "code hinting.”

- For Python, provide by extension (Python, Pylance)
- Brief tour



GitHub Copilot



GitHub Copilot

- AI tool to help in coding, similar to ChatGPT
 - Included for free in student GitHub education pack
 - Integrated in VS Code (through extension)
 - There is also a command line version: GitHub CLI
- Different ways of using it:
 - Inline suggestions
 - Prompting to write longer code
 - Completions
 - Chat
 - Ask questions about code
 - Re-format code



GitHub Copilot vs. ChatGPT

- GitHub Copilot now allows users to use a variety of models, including the latest OpenAI models, but also Anthropic's Claude (which is often better at coding)
- Advantage of GitHub Copilot is its seamless integration into VS Code —> don't leave your editor
- And you get access to advanced models for free!
- That said, I have found that ChatGPT, even the free version, is sometimes significantly better at providing coding solutions
- Your mileage may vary
- Conclusion: generally easiest to just use GitHub Copilot, but if you don't get a satisfactory answer, maybe consider ChatGPT



GitHub Copilot caveats

- Always check that what it suggests makes sense, don't blindly trust it
- When re-formatting multiple lines, can be difficult to check that everything makes sense.
- Generally always on (and somewhat difficult to turn off), also works on non-Python files: Markdown, LaTeX, ...
- That can be useful in some contexts, but beware of issues of plagiarism, academic misconduct: best to turn it off for any text you write for papers, dissertation, ... (can disable by default for specific file types)

Working in a git repository then makes it easier to look at changes