

# Empirical Studies Using Visual Studio Code

Chao Liu\*

May 23, 2023

In this note, I walk you through setting up Visual Studio Code (VS Code) on Windows PCs for empirical economics or finance research. There are three reasons why I think VS Code is better than other code editors, such as Stata's own do file editor:

1. Same editor for different codes (Stata, Python, and R)
2. AI Tools to help you code
3. Seamless integration with Git

## I Preamble

**Step 1:** Download VS Code compatible with your operating system from <https://code.visualstudio.com/Download>.

**Step 2:** Sign up for a new GitHub account at <https://github.com/join>.

**Step 3:** Download Git from <https://git-scm.com/>.

**Step 4:** Set up your Git for the first time. You can follow [https://kbroman.org/github\\_tutorial/pages/first\\_time.html](https://kbroman.org/github_tutorial/pages/first_time.html).

**Step 5:** Sign in with GitHub in VS Code (in the Accounts menu in the lower left of the Activity bar).

**Step 6:** Install the official GitHub extension for VS Code, **GitHub Pull Requests and Issues**, from the extensions marketplace. This extension provides a more seamless experience when working with GitHub repositories.

---

\*Liu: Kellogg School of Management, Northwestern University. Email: [chao.liu1@kellogg.northwestern.edu](mailto:chao.liu1@kellogg.northwestern.edu).

## II Using Stata in VS Code

You can use VS Code as the do file editor for Stata.<sup>1</sup> To do so, you first install **Stata Enhanced** and **Code Runner** from the extension marketplace in VS Code. The first extension can highlight Stata syntax in VS Code, and the second allows VS Code to run custom commands.

Next you will need the `rundo` and `rundolines` programs developed by Friedrich Huebler, which can be downloaded from his blog: <https://huebler.blogspot.com/2008/04/stata.html>. If you use Stata 15 or later, then download **rundo51.zip** and **rundolines51.zip**.

You can save the contents of the two ZIP files anywhere on your computer. A suggested location is the personal ADO directory of Stata. **Notice: The path cannot have spaces (e.g., C:\Program Files\Stata17\ado\personal\rundo51 is not allowed)**. Each ZIP archive contains four files: the executable, the source code, an INI file, and a readme file.

The final step is to modify the settings of **Code Runner**. Please follow the instructions in Figure 1 to get to the settings page. Then as shown in Figure 2, you tell the extension where to find the exe files of **rundo**, **rundolines**, and **Stata** (also your Stata version). The path you provide should use double slashes and not include any spaces.

To execute your code, open a Stata window and your do file in VS Code. The default shortcut keys are **ctrl+alt+N** to run the entire do file and **ctrl+alt+K** to run the selected code. Your results will appear in your active Stata window (see Figure 3 as an example where I run selected lines 1 to 7).

---

<sup>1</sup>I thank Yikai Zhao for sharing this method. His original post can be found here: [https://github.com/wulizyk/Notebook\\_Stata/wiki/Run-STATA-in-VScode](https://github.com/wulizyk/Notebook_Stata/wiki/Run-STATA-in-VScode).

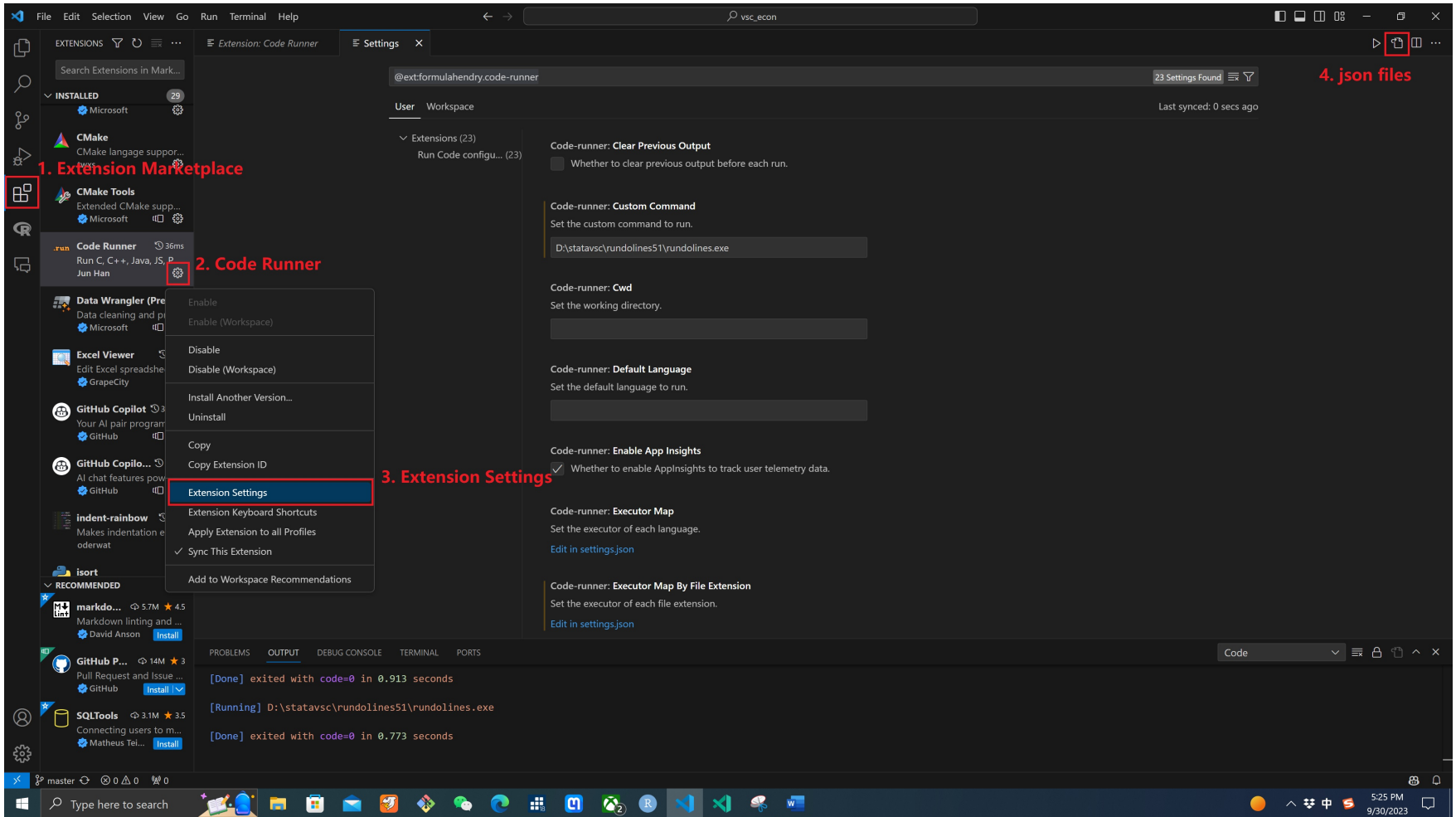


Figure 1. Find the Settings of Code Runner

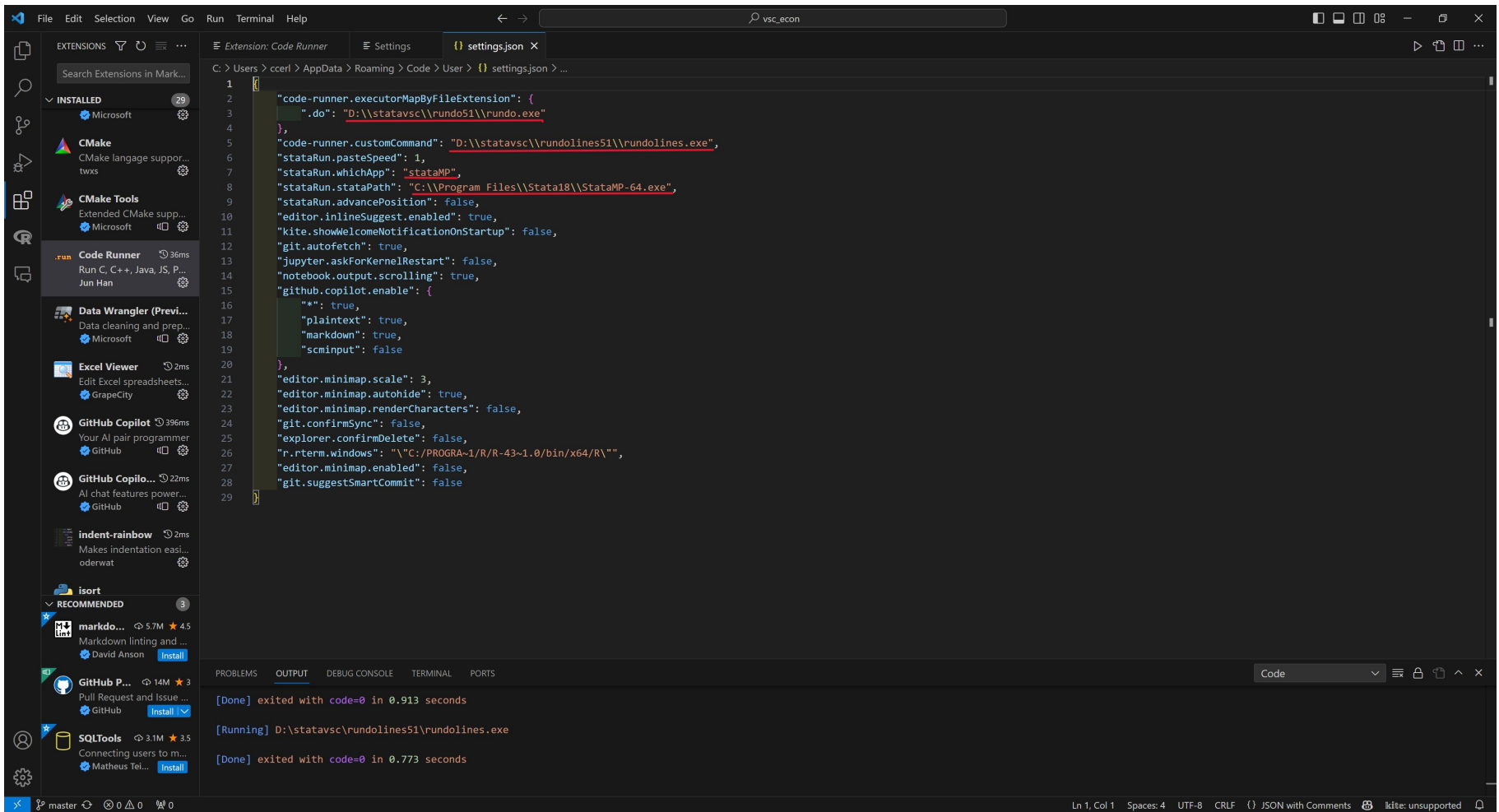


Figure 2. Adjust the Settings of Code Runner

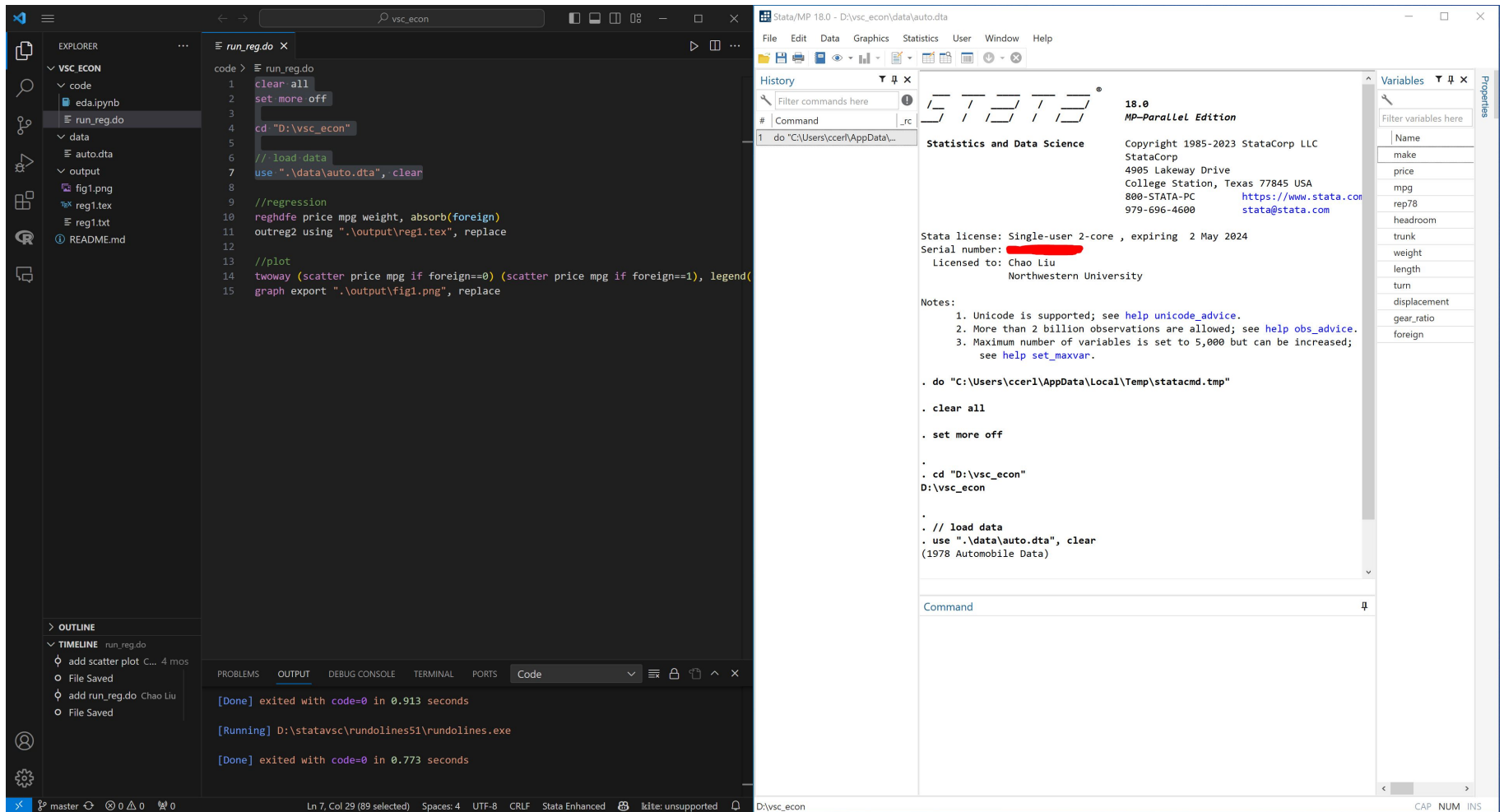


Figure 3. Run Stata Code in VS Code

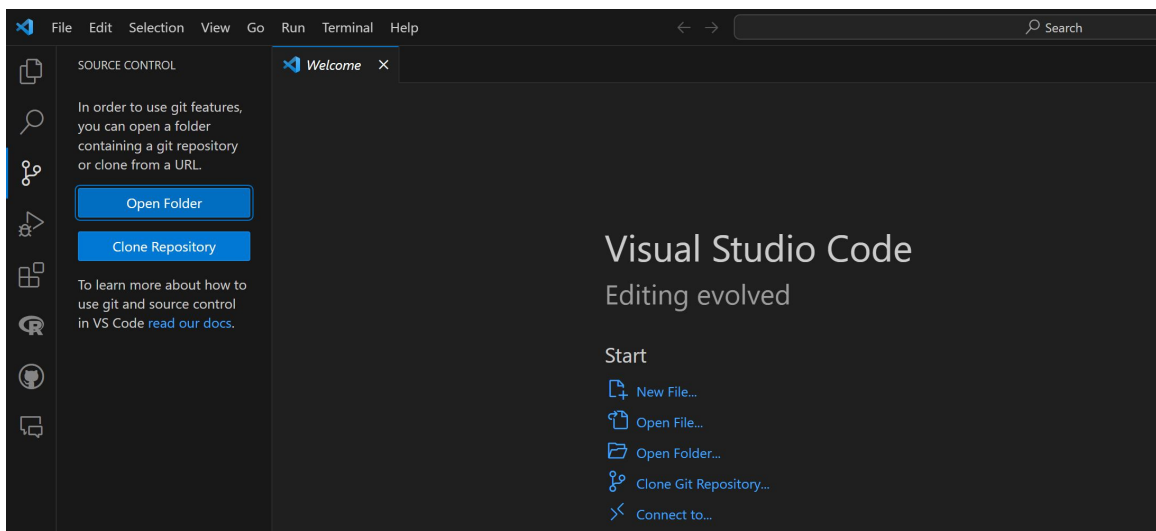


Figure 4. Open a Local Project Folder in VS Code

### III Using Git in VS Code

As more fields push open-data initiatives, having an online repository to share your data and code for replication is becoming increasingly important. Asjad Naqvi provides step-by-step instructions on how to sync your project files directly from Stata using Git commands.<sup>2</sup> Using VS Code makes this operation much easier if you have followed all steps in the preamble.

Suppose you have a project folder stored in your local disk. You can open this folder in VS Code (see Figure 4) and initialize a repository which will enable source control features powered by git. I prefer to directly publish the folder to a GitHub private repository, as shown in Figure 5.

Let's see what will happen if I modify the file in this folder. There is only one Stata do file which has two lines: "*clear all*" and "*set more off*". Now I add one command "*sysuse auto, clear*" to this file and save it. You will see this pending change in the left activity panel (see Figure 6). What you need to do is to stage this change (click the plus sign), add a message for this change, commit this change, and sync this change (see Figure 7). Congratulations! You have successfully utilized the power of source control.

---

<sup>2</sup>Please see <https://medium.com/the-stata-guide/stata-and-github-integration-8c87ddf9784a>.

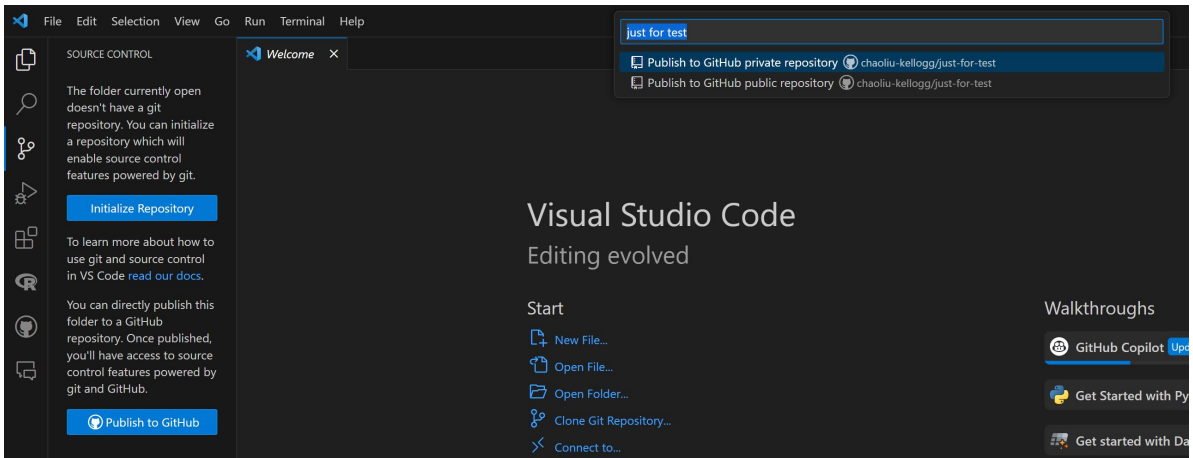


Figure 5. Publish to GitHub Private Repository

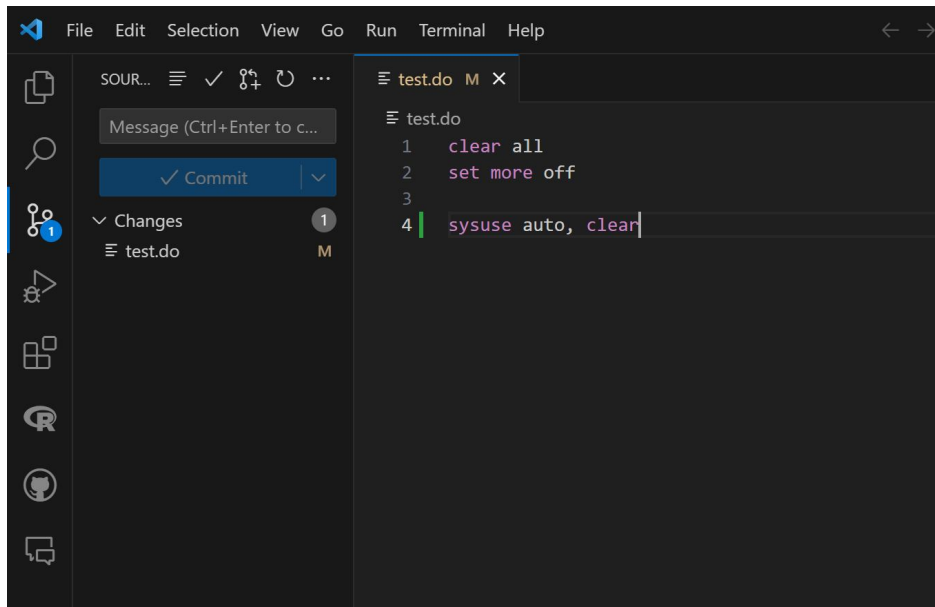


Figure 6. Stage Changes

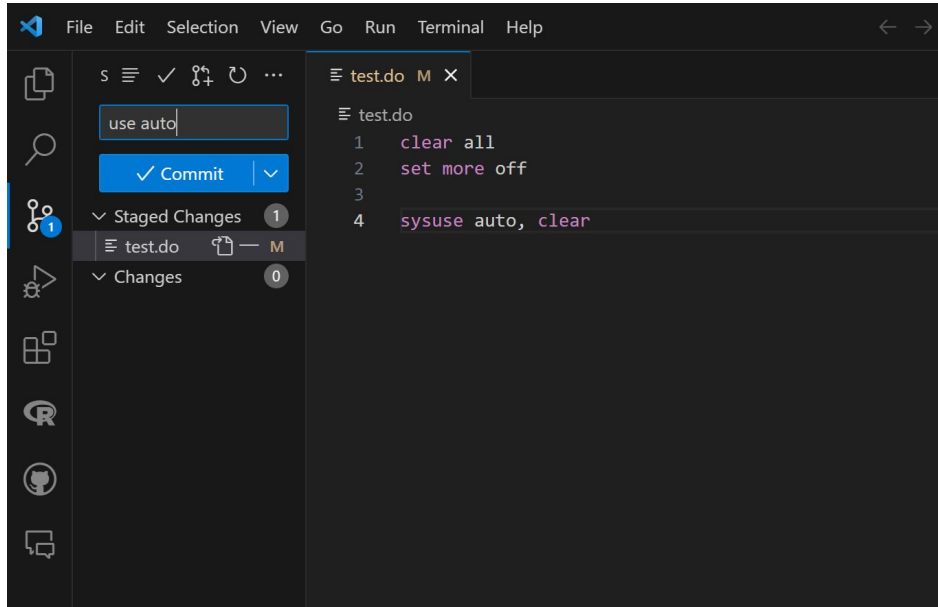


Figure 7. Commit Changes

## IV Using GitHub Copilot in VS Code

GitHub Copilot is an AI pair programmer that helps you write code faster and with less work. To use GitHub Copilot you must have an active GitHub Copilot subscription. There are two extensions in the VS Code marketplace: **GitHub Copilot Chat** and **GitHub Copilot**.

### A GitHub Copilot Chat

The use of GitHub Copilot Chat is very similar to that of ChatGPT. Figure 8 shows an example where I ask the AI pair programmer to explain the selected code. You can use this tool to understand the codes of published papers.

### B GitHub Copilot

This method offers a more seamless way to use AI. You simply need to type your requirement as a code comment and then press enter. GitHub Copilot will display its suggestion in grey. If you are satisfied with its code, press tab and the code will autocomplete. In Figure 9, I aim to plot the histogram of a variable, and the AI tool can provide me with the appropriate code.



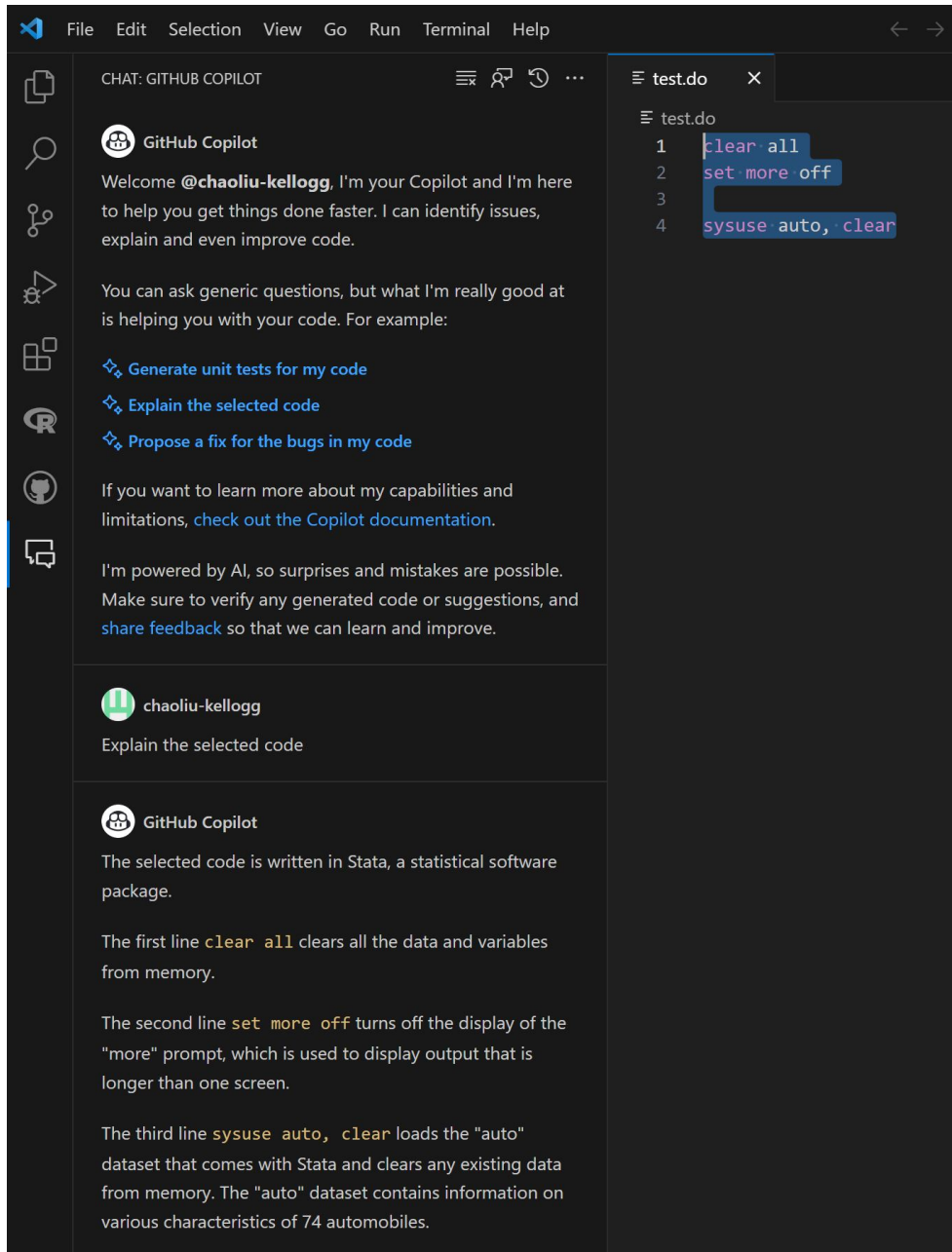
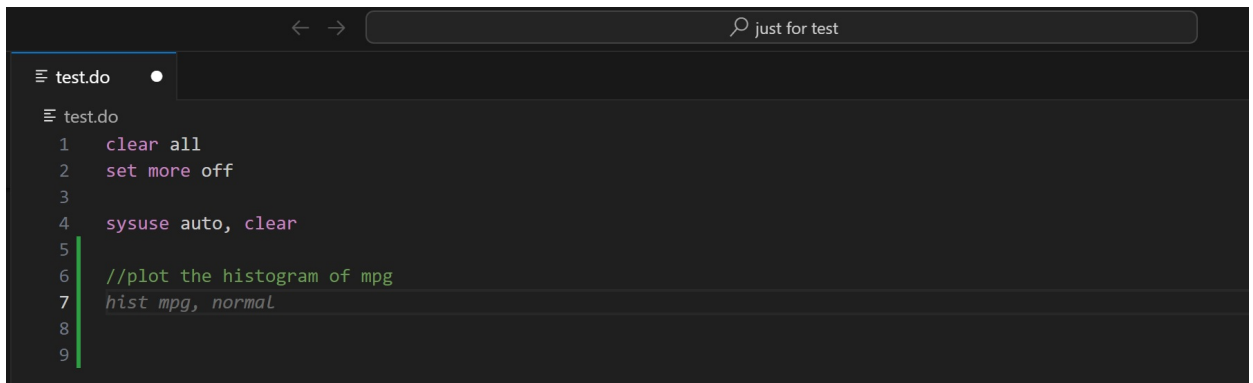


Figure 8. GitHub Copilot Chat



```
test.do
test.do
1 clear all
2 set more off
3
4 sysuse auto, clear
5
6 //plot the histogram of mpg
7 hist mpg, normal
8
9
```

Figure 9. GitHub Copilot