

Introduction to Stata Programming

Erasmus Thesis Project

Armin Hoendervangers
AEclipse ETP Committee

Current version: December 18, 2022¹

[Click here for latest version](#)

¹This is a work in progress, so the document will have incomplete sections and may contain mistakes.

Contents

1	Introduction	2
2	Information Management	3
2.1	The help command	3
2.2	scalars and matrices	5
2.3	display	5
2.4	macros	5
3	Automation	6
3.1	loops	6
4	Custom Commands	7
4.1	program	7
4.2	arguments	7
4.3	syntax	7
4.4	temporary variables	7
4.5	output	7
5	General Tips	8

—*—

1 Introduction

Reader with advanced tips and tricks for Stata. I'll add some introductory text about the reader here. Incomplete sections will have a very short description of what will be described in them. Could include some information about the thesis project, AEclipse, and maybe myself. I assume some knowledge of and/or experience with both Stata and programming in general. Example do-files will be available on the reader's [GitHub page](#). Comments and feedback are always welcome.

—*—

2 Information Management

Information is incredibly important in programming, whether it is what a command does, how its used, or what is contained in variables. I take information as a starting point, and this section is all about obtaining and storing various forms of information.

2.1 The `help` command

Perhaps the most important command in Stata, `help` allows quick access to information on *any* command Stata has. The help-files Stata provides are incredibly detailed, including information on how to use the command (its *syntax*), what the command does, its output, examples, and sometimes even the theory behind it. As useful as it is, the help-files might seem daunting at first. Understanding their structure is key to (quickly) obtain information without falling into despair. I'll highlight what I believe to be the most important parts of the help-files through an example.

If I type `help summarize`, Stata opens the window in Figure 1. In help-files, the typography on its own already gives us a lot of information.

Bold words indicate commands or options; if we want to use these, we type them exactly as they are written down. In our case, **summarize** is written in bold under the syntax heading. It is, as we know, indeed a command.

Italicised text indicates something that should be substituted. Here, *varlist* tells us that we should write down a list of variable names here – should we want to use this option.

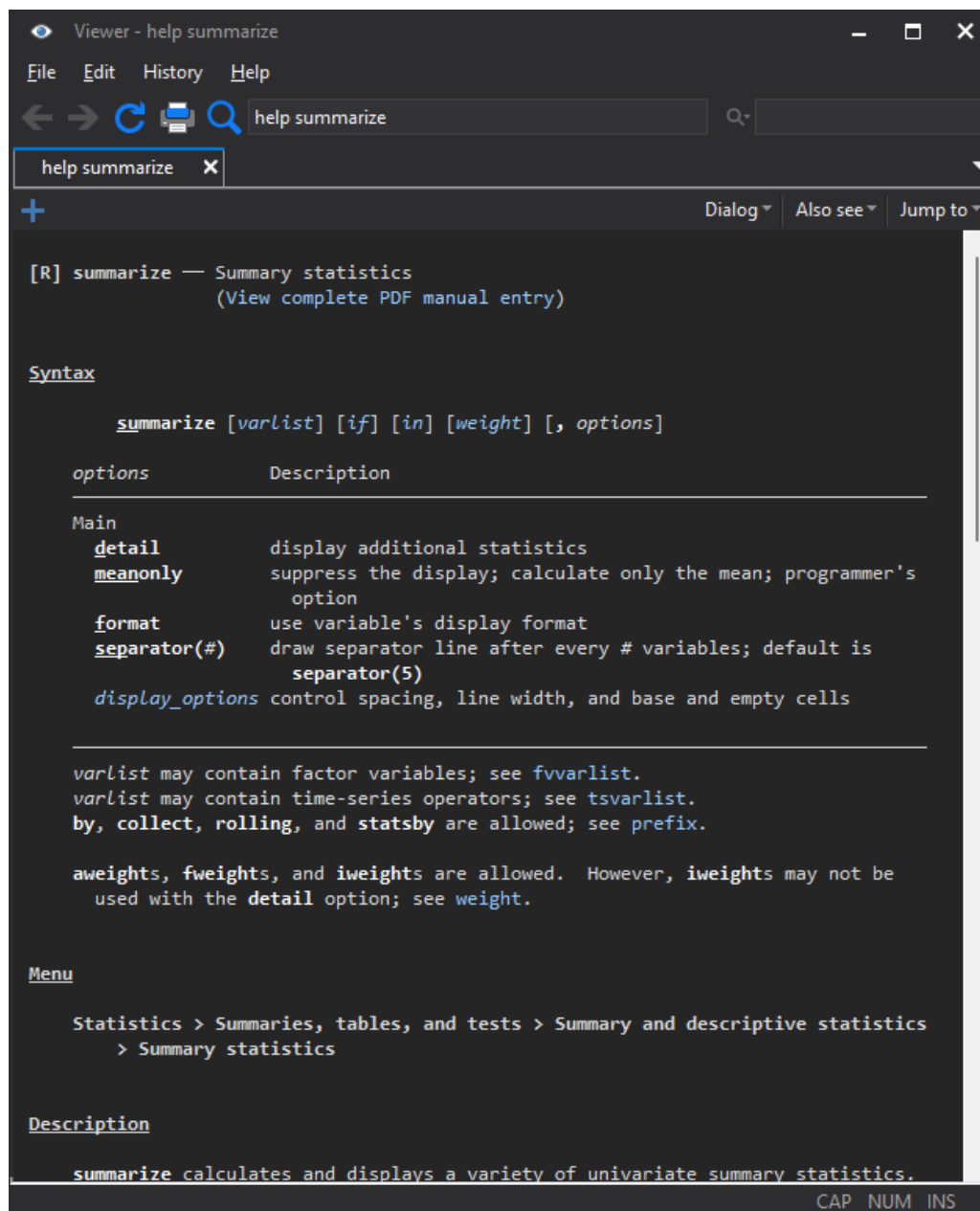
Optional arguments and functions are indicated by being [in brackets]. This means that anything that is written within brackets in the syntax is something that does not have to be specified for a command to work.

Underlined text indicates the *minimum* abbreviation of a command or option. In the case of `summarize`, I could simply write `su`. Additional letters are also allowed and how to use this is mostly personal preference. Personally, I always write `sum` as its clearer to me what that means than `su` would, but it still saves me the time and space from writing the command out in its entirety. When abbreviating commands, make sure you are familiar enough with them to remember what an abbreviation means if you open your do-files one week later. Having to look it up every time you see an abbreviation can be quite a pain.

Finally, any text in [blue](#) is a hyperlink, generally leading to more information on whatever is written down.²

² Note that the exact colour depends on Stata's colour scheme, but the default and dark schemes do use blue.

Figure 1: Help-file for summarize



2.2 scalars and matrices

storing information in scalars and matrices

2.3 display

display command

2.4 macros

globals and locals

—*—

3 Automation

This section is about different ways of automating things.

3.1 loops

different types of loops

—*—

4 Custom Commands

Section on how to write a custom command.

4.1 program

defining a program

4.2 arguments

adding user input to a program

4.3 syntax

adding standard Stata syntax to a program

4.4 temporary variables

using temporary variables

4.5 output

defining program output

—*—

5 General Tips

Things I found helpful in using Stata.