Options with the syntax command

- Another type of input are options, e.g.

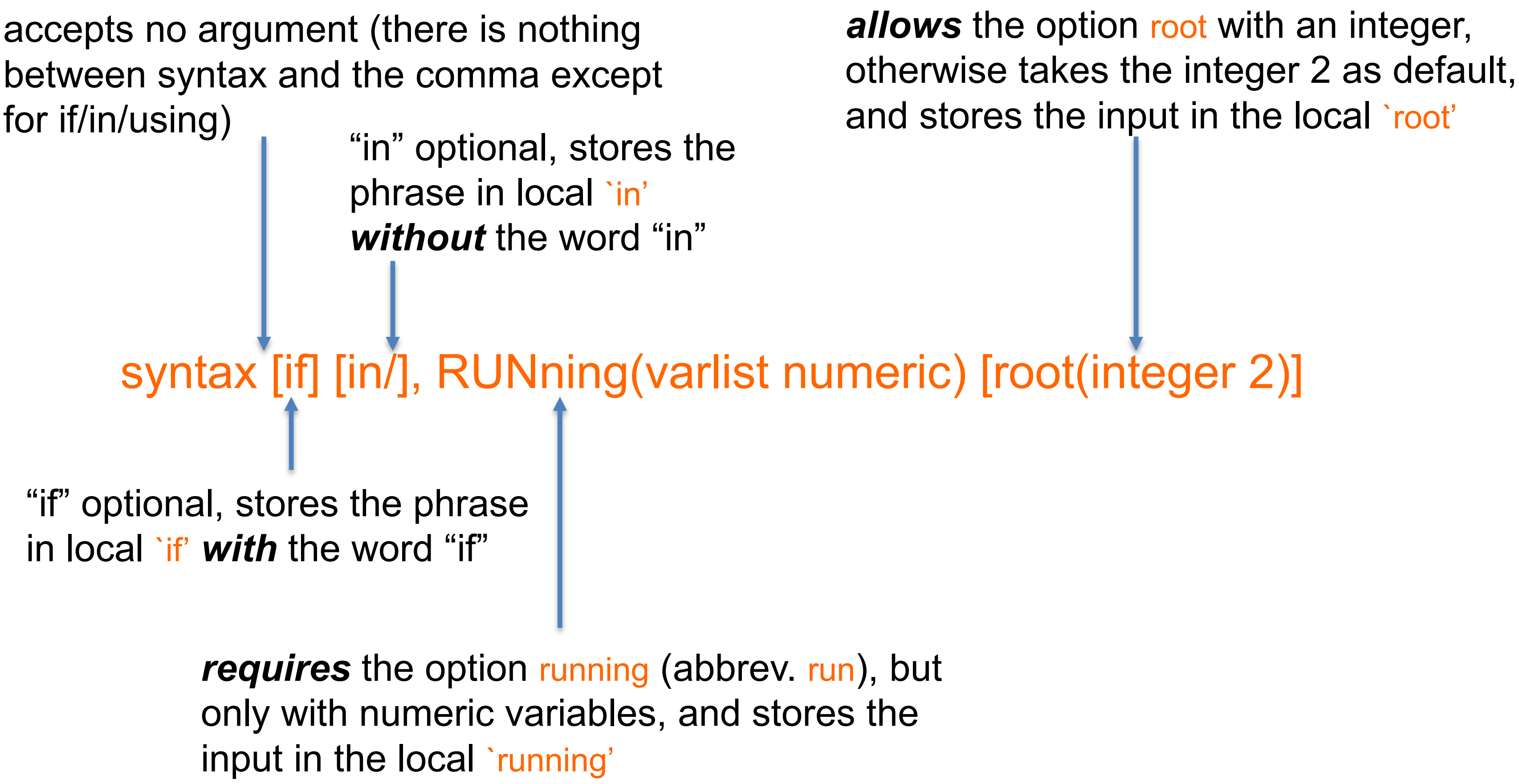
  reg income age sex if country == "France”, ***vce(cluster district)***

- The syntax command allows you to program your own options, e.g.

  syntax varlist(min=1) [if] [in], vce(namelist)

- You can specify options to be mandatory (no brackets) or optional (squared brackets)
- You can define abbreviations (abbreviation in caps)
- You can have options which are only words (e.g. replace) and options which require input (e.g. vce)
- For the latter, an input type is needed (varlist, numlist etc.) which can be amended by constraints (numeric, min/max etc.)

Options with the syntax command



accepts no argument (there is nothing between syntax and the comma except for if/in/using)

“in” optional, stores the phrase in local `in’ ***without*** the word “in”

allows the option root with an integer, otherwise takes the integer 2 as default, and stores the input in the local `root’

```
syntax [if] [in/], RUNning(varlist numeric) [root(integer 2)]
```

“if” optional, stores the phrase in local `if’ ***with*** the word “if”

requires the option running (abbrev. run), but only with numeric variables, and stores the input in the local `running’

More helpful options and commands

- Program options
  - The byable option lets the program accept the by prefix
  - The sortpreserve option tells Stata to restore the previous sorting after the program ends
- Remember that quietly suppresses the Stata output but still stores the results in r() etc. if applicable

A remark on error messages

- The syntax command comes with its own error messages for misspecification of the program syntax
- However, it might be useful to write your own error messages or warnings to prevent mistakes
- We have done this before using display and exit, but you can also include the pre-defined error codes using error
- To have the output printed red, type display in red