

# Principles of Programming in Econometrics

Introduction, structure, and advanced programming techniques

Charles S. Bos

Vrije Universiteit Amsterdam  
Tinbergen Institute

`c.s.bos@vu.nl`

August 2020 – Version Python

**Separate lecture slides**

Compilation: July 27, 2020

## Overview

# Principles of Programming in Econometrics

D0: Syntax, example 2<sup>8</sup>

D1: Structure, scope

D2: Numerics, packages

D3: Optimisation, speed

## Day 2: Numerics and flow

### 9.30 Numbers and representation

- ▶ Steps, flow and structure
- ▶ Floating point numbers
- ▶ Practical Do's and Don'ts
- ▶ Packages
- ▶ Graphics

### 13.30 Practical

- ▶ Cleaning OLS program
- ▶ Loops
- ▶ Bootstrap OLS estimation
- ▶ Handling data: Inflation

## import

Enlarging the capabilities of Python beyond basic capabilities:

import Use through:

- ▶ `import package`: You'll have to use `package.func()` to access function `func()` from the package
- ▶ `import package as p`: You may use `p.func()` as shorthand
- ▶ `from package import func`: You can use `func()` directly, but no other functions from the package
- ▶ `from package import *`: You can use all functions from the package directly

Custom use:

```
import numpy as np           # Shorten numpy to np
import pandas as pd          # Etc...
import matplotlib.pyplot as plt
from lib.incmyfunc import *   # Get all my own functions directly
```

## Python packages

Package	Purpose
<code>numpy</code>	Central, linear algebra and statistical operations
<code>scipy</code>	Additional scientific python routines
<code>matplotlib.pyplot</code>	Graphical capabilities
<code>pandas</code>	Input/output, data analysis
...	Many others...

Warning: Use packages, but with care. How can you ascertain that the package computes exactly what you expect? Do you understand?

## Private modules

- ▶ Convenient to package routines into modules, for use from multiple (related) programs
- ▶ Stored in local project/lib directory, if only related to current project
- ▶ ... or stored at central python/lib directory: Use environment variable PYTHONPATH to tell Python where modules may be found; see Spyder – Tools – PYTHONPATH Manager