

# 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

## Target of course

- ▶ Learn
- ▶ structured
- ▶ programming
- ▶ and organisation
- ▶ (in Python/Julia/Matlab/Ox or other language)

Not only: Learn more syntax... (mostly today)

Remarks:

- ▶ Structure: Central to this course
- ▶ Small steps, simplifying tasks
- ▶ Hopefully resulting in: Robustness!
- ▶ Efficiency: Not of first interest... (Value of time?)
- ▶ Language: Theory is language agnostic

## Target of course II

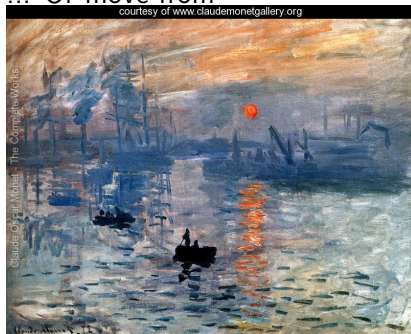
... Or move from

courtesy of [www.claudemonetgallery.org](http://www.claudemonetgallery.org)

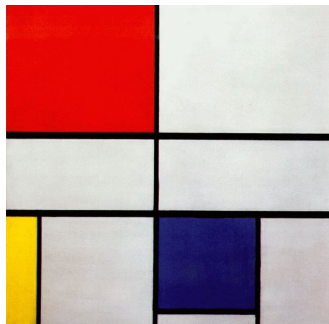


## Target of course II

... Or move from



to



(Maybe discuss at end of first day?...)

# Syntax

What is 'syntax'?

- ▶ Set of rules
- ▶ Define how program 'functions'
- ▶ Should give clear, non-ambiguous, description of steps taken
- ▶ Depends on the language

Today:

- ▶ Learn basic Python syntax
- ▶ Learn to read manual/web/google for further syntax!

## Syntax II

What is not 'syntax'?

- ▶ Rule-book on how to program
- ▶ Choice between packages
- ▶ Complete overview

For clarity:

- ▶ We will *not* cover all of Python
- ▶ We make a (conservative) *choice* of packages (`numpy`, `scipy`, `pandas`, `matplotlib`)
- ▶ We focus on structure, principle, guiding thoughts
- ▶ ... and then you should be able to do the hard work

## Overview

# Principles of Programming in Econometrics

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

D1: Structure, scope

D2: Numerics, packages

D3: Optimisation, speed

## Day 0: Syntax

### 9.30 Introduction

Example:  $2^8$

Elements

Main concepts

Closing thoughts

Revisit E0

### 13.30 Practical (at VU, main building)

- ▶ Checking variables, types, conversion and functions
- ▶ Implementing Backsubstitution



# Day 1: Structure

## 9.30 Introduction

- ▶ Programming in theory
- ▶ Science, data, hypothesis, model, **estimation**

## Structure & Blocks (Droste)

### Further concepts of

- ▶ Data/Variables/Types
- ▶ Functions
- ▶ Scope, globals

## 13.30 Practical

- ▶ Regression: Simulate data
- ▶ Regression: Estimate model

## 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

## Day 3: Optimisation

### 9.30 Optimization (minimize)

- ▶ Idea behind optimization
- ▶ Gauss-Newton/Newton-Raphson
- ▶ Stream/order of function calls
- ▶ Standard deviations
- ▶ Restrictions
- ▶ Speed

### 13.30 Practical

- ▶ Regression: Maximize likelihood
- ▶ GARCH-M: Intro and likelihood

## Evaluation

- ▶ No old-fashioned exam
- ▶ Range of exercises, to try out during course
- ▶ Short voluntary final exercise (see VU Canvas, **TBA**). If you hand it in, you may receive some comments/hints on programming style.

Main message: Work for your own interest, later courses will be simpler if you make good use of this course...