Workshop: High-performance computing for economists

Lars Vilhuber[1] John M. Abowd[1] Richard Mansfield[1]
Kevin L. McKinney

[1]Cornell University, Economics Department,

August 20-22, 2013: Day 1

What do you learn in a Ph.D. program?

What do you learn in a Ph.D. program?

How to learn...

Goal of this class

Goal of this class

To open new doors, to be able to conceive of problems that you didn't think had a feasible solution.

Goal of this class

To open new doors, to be able to conceive of problems that you didn't think had a feasible solution.

To broaden your knowledge about what you do NOT know

Overview

Day 1

- Programming basics (Lars)

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.
  - A clean sequence of programs

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.
  - A clean sequence of programs
  - NX, SSH, Linux, request an account on cluster

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.
  - A clean sequence of programs
  - NX, SSH, Linux, request an account on cluster
  - Basic scripting

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.
  - A clean sequence of programs
  - NX, SSH, Linux, request an account on cluster
  - Basic scripting
- Basics of version control (Lars)

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.
  - A clean sequence of programs
  - NX, SSH, Linux, request an account on cluster
  - Basic scripting
- Basics of version control (Lars)
  - File-system based version control

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.
  - A clean sequence of programs
  - NX, SSH, Linux, request an account on cluster
  - Basic scripting
- Basics of version control (Lars)
  - File-system based version control
  - More formal version control (Subversion, Git)

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.
  - A clean sequence of programs
  - NX, SSH, Linux, request an account on cluster
  - Basic scripting
- Basics of version control (Lars)
  - File-system based version control
  - More formal version control (Subversion, Git)
  - Working with servers

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.
  - A clean sequence of programs
  - NX, SSH, Linux, request an account on cluster
  - Basic scripting
- Basics of version control (Lars)
  - File-system based version control
  - More formal version control (Subversion, Git)
  - Working with servers
  - Setting up infrastructure at Cornell

Overview

Day 1

- Programming basics (Lars)
  - Choosing an editor
  - How to structure programs, texts, etc.
  - A clean sequence of programs
  - NX, SSH, Linux, request an account on cluster
  - Basic scripting
- Basics of version control (Lars)
  - File-system based version control
  - More formal version control (Subversion, Git)
  - Working with servers
  - Setting up infrastructure at Cornell
- Subroutines: the example of function programming in R (Lars)

Overview

Day 2

Overview

Day 3

Choosing an editor

... or system

Separate editors and systems

- MS Word and math editor
- LibreOffice
- LaTeX(TeXstudio, TeXMaker, Scientific Workplace, TeXWorks+Miktex, etc.)
- NotePad++ (Windows)
- Gedit, (X)Emacs (Linux)

Integrating programming and running

- IDE (Eclipse, ActiveState Komodo, etc.)
- Native programming GUIs (SAS, Matlab, Stata)
- Gedit, (X)Emacs

Integrating programs and text/results

- SWeave (integrates LaTeX and R)
- RStudio (GUI to R and SWeave)

Basic scripting in Linux

A basic loop on the command line

```
for (( i; i<10; i++ ))
do
 echo $i
done
for i in 1 3 7 99
do
 echo $i
done
```

Source: [1]

Now let's try it out

Next section

Next section