

# Introduction

## R applications and modelling

Arni Magnusson

Universidad de Concepción, Chile  
14–18 Jan 2013

# Topics

Statistics - data, plots, tests, models, uncertainty, simulation

R - programming, input/output, packages, help

Exercises - linear and nonlinear models, writing functions

# Schedule

## Day 1 Basic R

data, plots, tests, linear models, project management, help

## Day 2 Intermediate R

generalized linear models, advanced plots, writing functions

## Day 3 Nonlinear models

uncertainty, maximum likelihood, Hessian, MCMC

## Day 4 Programming

simulations, performance, interacting with other programs

## Day 5 Advanced R

configuration, writing packages

# Teach yourself programming in 10 years

Famous essay by Peter Norvig (NASA, Google)

It takes around 10 years to develop expertise in a given field:  
chess, paint, tennis, swim, piano, programming, . . .

Use the programming languages that your friends use

Learn programming in an interactive mode: instant feedback

Work with other programmers, contribute to open source projects

# Organizing your notes and code this week

Some ideas ...

Take personal [course notes](#) (day by day), either in a word processor, text editor, or just on paper.

Organize [long-term notes](#) (by subject) in a word processor or text editor. You can gradually build these notes over months and years, and use them as a reference.

Also save commands in [R scripts](#). Scripts are text files that contain code that can be pasted or sourced into R.

Previous commands can be accessed by pressing the up/down arrows, or by typing `history()`. Salvage commands from the history into neatly organized scripts.

# Format of the course

## Challenges

- A lot of statistics and programming concepts in a short time  
⇒ may feel too fast and confusing
- Participants have different background in statistical computing

## Approach

- Emphasis on exercises and discussion rather than lectures
- Open-ended exercises ⇒ participants can work on basic or advanced aspects, depending on their background
- Bring your own exercises and projects to class
- Work together in pairs as much as we can; we gain deeper insight when discussing with others

# Introduce ourselves

Name and workplace

Project(s) you are working on

Previous background in R and statistical computing in general

What you hope to learn this week

# Course evaluation

R scripts that you create (50%)

Short exam (50%)

# Course evaluation

## R scripts that you create (50%)

At the end of each day, email me all R code that you wrote that day. The R scripts should:

- Reach me before class starts the next day.
- Run on my computer without returning an error, but if you want you can also send a separate file with R code that doesn't quite work.
- Not include unnecessary code.
- Include some comments describing the purpose of each part of the code. Well organized code doesn't need a lot of comments. The comments should enable an experienced R programmer to read quickly through the script and understand it.

## Short exam (50%)

# Course evaluation

## R scripts that you create (50%)

### Short exam (50%)

- Thursday afternoon
- 3–5 questions that you answer by writing R functions.
- You can use the R help, manuals, books, web search, etc. during the exam.