



Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

Must-Know Things In GNU R for Sharing Knowledge

- OOP and Package -

Friedrich Leisch¹ Chel Hee Lee²

¹R Development Core Team

²R Translation Team

October 28, 2011



Outline

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

1 Introduction

2 Analytic Solution

3 Objective Oriented Programming

4 S3 System

5 Interfacing Formula

6 Packaging

Dr. Friedrich Leisch



- Professor, Institute of Applied Statistics and Computing, University of Natural Resources and Life Sciences, Vienna, Austria. Head of Institute.



소개

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

Dr. Friedrich Leisch의 활동

- R project for statistical computing member of core development team since 1997, editor of R News 2001–2003, secretary general of the R Foundation since 2002.
- Omega project for statistical computing: member of core development team since 1999.
- Bioconductor project for statistical software in bioinformatics: member of core development team since 2001.



Message from Dr. Friedrich Leisch

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

Dear Korean R Users (1/3)

It is my pleasure to send my warmest regards and best wishes to you. It is exciting to hear that approximately 200 useRs will come together to hear and give talks about R in Korea.

When I joined the R project in 1996 by sending first bug reports to Ross and Robert, I guess about 200 people in the world may have known about R, and only a handful was developing it. Now - 15 years later - there are fantastic events like the one you are attending all around the globe. A really amazing development.



Message from Dr. Friedrich Leisch

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

Dear Korean R Users (2/3)

One key component of the success story behind R has been the package system which was developed by Kurt Hornik and myself. Together we also started the Comprehensive R Archive Network CRAN in 1997 and maintain it until now. The package system allows individuals outside of the R Core Team to easily contribute to the R project, it is the most fundamental infrastructure for the R community. Without it, R would not be what it is today. We started with a few dozen packages, as I write these lines there are 3363 on CRAN, several hundred on Bioconductor and who knows how many in private repositories.

GNU R Korea



Message from Dr. Friedrich Leisch

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

Dear Korean R Users (3/3)

I am honored that Chel Hee Lee has translated my tutorial on Creating R packages, which I wrote for the Compstat conference 2008 in Porto, to Korean.

I hope you will find the material useful. R is an open source project, it lives and takes its spirit from free contributions by volunteers.

I wish you all a very productive and happy conference!

*Best regards,
Fritz Leisch*

다양한 형태의 지식공유

- 일상생활에서의 대화를 통한 정보교환,
- 다양한 종류의 매체를 통하여 자신의 생각을 자유롭게 토론하는 활동,
- 학술지를 통한 연구결과 및 토의내용을 기고하는 활동.

실무자의 입장에서,

- 자체적 통계모형의 개발이 필요한 경우 (통계적 방법론),
- 물리적 한계를 극복하고자 하는 경우 (병렬 및 분산 컴퓨팅),
- 편리한 사용자 환경을 개발하고자 하는 경우 (GUI 개발), 등등

지식공유의 도구로서의 패키지 시스템

"As an article distributes scientific ideas to others, a package distributes statistical methodology to others"
(Leisch, 2009, p. 1).

성공적 R-project의 결정적 요소 중 하나 (R Development Core Team, 2010a)

- 특정운영체제에 제약없이 손쉽게 간결한 설치,
- 편리한 유지보수 및 관리,
- 공유를 위한 간편한 방법 제공,
- 효율적인 메모리 관리



자유로운 지식공유의 활동은...

- 함께 생각하는 자연스런 환경제공,
- 빠른 협력체제를 구축 및 해당분야의 손쉬운 관심인재 확보,
- 다양한 창의적 시각의 반영,
- 폭넓은 사용자에게 의한 즉각적 검증체제로 인한 하이-퀄리티 시스템 개발,
- 활용할 수 있었던 지식에 대한 보답으로서의 지식의 재생산.

Note:

현재 슬라이드는 토론이 많이 필요한 이철희만의 생각입니다.



본 발표자료는...

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

실무자의 측면에서 R을 활용할 때,

- 본래의 연구활동을 돕기 위하여,
- 기술적인 지식의 습득을 보다 쉽고 빠르게 익힐 수 있도록,
- 번역 및 수정된 튜토리얼이며,
- 본래의 페이퍼를 확인하고자 하시는 분들을 위하여, Leisch (2009) 가 제공하는 예제를 재사용.

주요내용

- 작성된 함수에 클래스와 메소드 정의,
- 자유로운 모델 정의 및 수정을 위한 포물러 기능 추가,
- 도움말 작성하기,
- 제작된 패키지 설치, 테스트, 그리고 배포.



용어 정리

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

- **Package:**

R BASE 시스템 내의 코드, 데이터, 그리고 문서에 대한 확장.
즉, 애드온 (Add-on) 시스템.

- **Library:**

설치된 패키지를 포함하는 디렉토리

- **Repository:**

패키지를 다운로드 받을 수 있는 온라인 저장소

- **Source:**

프로그래머가 읽을 수 있도록 작성된 실제 코드

- **Binary:**

컴퓨터가 인식할 수 있도록 컴파일된 소스코드 (운영체제에 따라 다를 수 있음)

- **Base packages:**

R Development Core Team 에서 관리 및 배포하는 R 시스템의 핵심 코드들의 집합체

- **Recommended packages:**

R Development Core Team 에서 반드시 관리하지는 않으나, 베이스 시스템 설치에 동반되는 소스코드들.

모형 설명

y 와 x 를 각각 종속변수와 설명변수라고 가정합니다.

$$y = x'\beta + \epsilon,$$

이때, $\epsilon \sim NID(0, \sigma^2)$ 은 오차항이며, σ^2 는 분산입니다.

애널리틱 솔루션

최소제곱법을 통한 추정된 회귀계수는

$$\hat{\beta} = (X'X)^{-1}X'y$$

분산-공분산 행렬은

$$V(\hat{\beta}) = \sigma^2(X'X)^{-1}$$



회귀분석 함수정의

```
linmodEst <- function(x, y){  
  # compute QR-decomposition of x  
  qx <- qr(x);  
  
  # compute  $(x'x)^{-1} x'y$   
  coef <- solve.qr(qx, y);  
  
  # degrees of freedom and standard deviation of residuals  
  df <- nrow(x)-ncol(x);  
  sigma2 <- sum((y - x %*% coef)^2)/df;  
  
  # compute  $\sigma^2 * (x'x)^{-1}$   
  vcov <- sigma2 * chol2inv(qx$qr);  
  colnames(vcov) <- rownames(vcov) <- colnames(x);  
  list( coefficients = coef, vcov = vcov,  
        sigma = sqrt(sigma2), df = df);  
}
```

linmodEst() - 사용자정의

```
R> data(cats, package="MASS");
R> with(cats,
+ linmodEst(cbind(1, Bwt), Hwt));
```

\$coefficients

```
[1] -0.3566624 4.0340627
```

\$vcov

```
[,1]
[,2]
[1,] 0.4792475 -0.17058197
[2,] -0.1705820 0.06263081
```

\$sigma

```
[1] 1.452373
```

\$df

```
[1] 142
```

lm() - 내장함수

```
R> lm1 <- lm(Hwt~Bwt, data=cats);
R> lm1
```

Call:

```
lm(formula = Hwt ~ Bwt, data = cats)
```

Coefficients:

```
(Intercept)      Bwt
-0.3567      4.0341
```

```
R> vcov(lm1)
```

```
(Intercept)      Bwt
(Intercept)  0.4792475 -0.17058197
Bwt         -0.1705820  0.06263081
```

```
>
```




TODO...

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

어떻게 해야 일반적 내장함수와 같은 기능들을 추가할 수 있나요?

- 함수의 결과물을 자유로이 엑세스하고,
- `summary()`와 같은 제네릭 함수를 적용하며,
- 자유로이 변형이 가능한 모형식을 입력받는 기능이 필요.

무엇을 이해해야 하나요?

- 제네릭 함수,
- 객체시스템.



Objective Oriented Programming

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

객체지향 프로그래밍

- **클래스(Class):**
프로토타입의 정의
- **객체(Object):**
런타임시 클래스로부터 생성된 결과물
- **메소드(Method):**
주어진 객체를 활용하여 특정 작업을 수행하도록 정의
- **노트:** C++ 혹은 Java와 같은 순수 OOP와는 사용에 다소 차이가 있음.

In R

- **제네릭 함수(Generic function):**
객체가 가질 수 있는 다양한 종류의 클래스들에 대하여, 하나의 함수명 아래 각 클래스별 메소드를 정의해 놓은 집합체.
- 제네릭함수의 동작을 수행할 수 있는 **메소드 디스패치 (method dispatch)**를 지시하는 규칙의 정의를 객체시스템이라고 함.
- S3 (구버전), S4 (신버전), R5 (최신버전).



Summary() 함수를 통한 제네릭 함수의 이해

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

데이터 타입 - Integer

```
R> x <- rep(0:1, c(10, 20))
R> x

[1] 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 ...
R> class(x)

[1] "integer"
R> summary(x)

Min.    1st Qu.  Median    Mean 3rd Qu.    Max.
0.0000  0.0000   1.0000  0.6667  1.0000  1.0000
```

데이터 타입 - Factor

```
R> y <- as.factor(x)
R> class(y)

[1] "factor"
R> summary(y)

 0  1
10 20
```

Step 1. UseMethod() 함수를 이용하여 클래스 선언 후, 디폴트 액션을 정의함.

```
linmod <- function(x, ...){
  UseMethod("linmod");
}

linmod.default <- function(x, y, ...) {
  x <- as.matrix(x);
  y <- as.numeric(y);

  est <- linmodEst(x, y);

  est$fitted.values <- as.vector(x %*% est$coefficients);
  est$residuals <- y - est$fitted.values;
  est$call <- match.call();

  class(est) <- "linmod";
  est;
}
```

Step 2. 클래스로부터 생성된 객체에 메소드 추가

```
print.linmod <- function(x, ...){
  cat("Call:\n");
  print(x$call);
  cat("\nCcoefficients:\n");
  print(x$coefficients);
}
```

노트: 메소드 추가시 네이밍규칙 (naming convention) → 메소드.클래스

출력메소드 추가 후 결과

```
R> x <- cbind(Const=1, Bwt=cats$Bwt);
R> y <- cats$Hw;
R> mod1 <- linmod(x, y);
R> mod1;
```

```
Call:
linmod.default(x = x, y = y)
```

```
Coefficients:
      Const      Bwt
-0.3566624  4.0340627
```

Step 2-1. 추가 메소드 정의

```
summary.linmod <- function(object, ...){
  se <- sqrt(diag(object$vcov));
  tval <- coef(object) / se;
  TAB <- cbind(Estimate = coef(object), StdErr = se, t.value = tval,
               p.value = 2*pt(-abs(tval), df=object$df));
  res <- list(call=object$call, coefficients=TAB);
  class(res) <- "summary.linmod";
  res;
}
```

Step 2-2. 추가 메소드 정의

```
print.summary.linmod <- function(x, ...){
  cat("Call:\n");
  print(x$call);
  cat("\n");
  printCoefmat(x$coefficients, P.value=TRUE, has.Pvalue=TRUE);
}
```



최종 결과물 확인

```
R> summary(mod1)
```

Call:

```
linmod.default(x = x, y = y)
```

	Estimate	StdErr	t.value	p.value
Const	-0.35666	0.69228	-0.5152	0.6072
Bwt	4.03406	0.25026	16.1194	<2e-16 ***

Signif. codes:

0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

```
R> coef(summary(mod1))[,4]
```

Const	Bwt
6.072131e-01	6.969045e-34

포뮬러 메소드 이해하기

- 일반적으로 사용자로부터 입력받는 모델식

$$y \sim x1 + x2 + x3$$

- + 혹은 ~과 같은 모델연산자는 파싱에서 제외,
- 변수명 $y, x1, x2, x3$ 으로만 구성된 **모델프레임 (model.frame)**이라는 새로운 객체 생성,
- 모델프레임의 첫번째와 두번째 열을 각각 종속변수와 인터셉트로 인식,
- 변수명은 모델프레임의 **항 (terms)**라는 속성으로 사용됨.

Step 3. 모델식 입력기능 추가

```
linmod.formula <- function(formula, data=list(), ...) {
  mf <- model.frame(formula=formula, data=data);
  x <- model.matrix(attr(mf, "terms"), data=mf);
  y <- model.response(mf);

  est <- linmod.default(x, y, ...);
  est$call <- match.call();
  est$formula <- formula;
  est;
}
```

결과확인

```
R> summary(linmod(Hwt~Bwt*Sex, data=cats))
```

Call:

```
linmod.formula(formula = Hwt ~ Bwt * Sex, data = cats)
```

	Estimate	StdErr	t.value	p.value
(Intercept)	2.98131	1.84284	1.6178	0.1079605
Bwt	2.63641	0.77590	3.3979	0.0008846 ***
SexM	-4.16540	2.06176	-2.0203	0.0452578 *
Bwt:SexM	1.67626	0.83733	2.0019	0.0472246 *

Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1

따라하기

- 바탕화면에 **tmp** 라는 임시 폴더 생성
- **linmod.R**이라는 새로운 파일을 **tmp**내에 생성
- 위에서 작성한 모든 함수들(`linmodEst`, `linmod`, `linmod.default`, `print.linmod`, `summary.linmod`, `print.summary.linmod`, `linmod.formula`)을 **linmod.R**에 기록후 저장.
- R 을 실행하고 **tmp**를 워킹디렉토리로 지정
- 패키지 생성을 위해 **package.skeleton()**함수 사용

```
R> package.skeleton(name="linmod", code_files="linmod.R")
Creating directories ...
Creating DESCRIPTION ...
Creating Read-and-delete-me ...
Copying code files ...
Making help files ...
Done.
Further steps are described in './linmod/Read-and-delete-me'.
```



패키지 만들기

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

실행절차

• tmp 디렉토리 확인해보기

```
chel@gnustats:~$ ls -R ~/Desktop/tmp/  
/home/chel/Desktop/tmp/:  
linmod  linmod.R  
  
/home/chel/Desktop/tmp/linmod:  
DESCRIPTION  man  R  Read-and-delete-me  
  
/home/chel/Desktop/tmp/linmod/man:  
linmod.default.Rd  linmod-package.Rd  print.summary.linmod.Rd  
linmodEst.Rd       predict.linmod.Rd    summary.linmod.Rd  
linmod.formula.Rd  print.linmod.Rd  
  
/home/chel/Desktop/tmp/linmod/R:  
linmod.R  
chel@gnustats:~$
```



실행절차

- **DESCRIPTION**파일 확인해보기

```
chel@gnustats:~$ cat ~/Desktop/tmp/linmod/DESCRIPTION
Package: linmod
Type: Package
Title: What the package does (short line)
Version: 1.0
Date: 2011-09-22
Author: Who wrote it
Maintainer: Who to complain to <yourfault@somewhere.net>
Description: More about what it does (maybe more than one line)
License: What license is it under?
LazyLoad: yes
chel@gnustats:~$
```

- **DESCRIPTION** 파일은 데비안리눅스에서 개발한 데비안 컨트롤 파일 형식 (Debian-control-file format)을 따릅니다.

- 패키지에 관련된 사항을 입력합니다.

실행절차

- **man** 디렉토리는 작성한 함수들에 대한 도움말을 저장합니다.
- 파일명이 **.Rd**와 함께 끝나는 것은 R Documentation를 의미합니다.

```
chel@gnustats:~$ head -10 ~/Desktop/tmp/linmod/man/linmodEst.Rd
\name{linmodEst}
\alias{linmodEst}
%- Also NEED an '\alias' for EACH other topic documented here.
\title{
linmodEst
%% ~~function to do ... ~~
}
\description{
%% ~~ A concise (1-5 lines) description of what the function does. ~~
}
chel@gnustats:~$
```

- 다른 사용자를 위하여 도움말을 성의있게 작성합니다.

실행절차

• .tar.gz파일 생성하여 패키지 완성 후 배포하기

```
chel@gnustats:~$ cd Desktop/tmp
chel@gnustats:~/Desktop/tmp$ R CMD build linmod
* checking for file 'linmod/DESCRIPTION' ... OK
* preparing 'linmod':
* checking DESCRIPTION meta-information ... OK
* checking for LF line-endings in source and make files
* checking for empty or unneeded directories
* building 'linmod_1.0.tar.gz'
```

```
chel@gnustats:~/Desktop/tmp$ ls
linmod linmod_1.0.tar.gz linmod.R
```

- 만약, 크로스 컴파일을 원하신다면 **R CMD build -binary**를 이용하세요.

설치 후 테스트 해보기

- 먼저 라이브러리 경로를 파악합니다.

```
R> .libPaths()
[1] "/home/chel/R/i686-pc-linux-gnu-library/2.12"
[2] "/usr/local/lib/R/site-library"
[3] "/usr/lib/R/site-library"
[4] "/usr/lib/R/library"
```

- 어떠한 라이브러리도 설치되지 않았음을 확인합니다.

```
chel@gnustats:~$ ls -R ./R/i686-pc-linux-gnu-library/2.12
./R/i686-pc-linux-gnu-library/2.12:
chel@gnustats:~$
```

- 또한, 패키지한 함수를 사용할 수 있는지 확인합니다.

```
R> data(cats, package="MASS")
R> linmod(Hwt~Bwt*Sex, data=cats)
Error: could not find function "linmod"
R>
```

- 아직 설치 전이므로 사용할 수 없습니다.



설치 후 테스트 해보기

- 아래와 같이 설치합니다.

```
chel@gnustats:~$ cd Desktop/tmp
chel@gnustats:~/Desktop/tmp$ R CMD INSTALL -l ~/R/i686-pc-linux-gnu-library/2.
* installing *source* package 'linmod' ...
** R
** preparing package for lazy loading
** help
*** installing help indices
** building package indices ...
** testing if installed package can be loaded

* DONE (linmod)
chel@gnustats:~/Desktop/tmp$
```



설치 후 테스트 해보기

- 라이브러리를 확인합니다.

```
chel@gnustats:~$ ls -R ~/R/i686-pc-linux-gnu-library/2.12  
/home/chel/R/i686-pc-linux-gnu-library/2.12:
```

```
linmod  R.css
```

```
/home/chel/R/i686-pc-linux-gnu-library/2.12/linmod:
```

```
DESCRIPTION  help  html  INDEX  Meta  R
```

```
/home/chel/R/i686-pc-linux-gnu-library/2.12/linmod/help:
```

```
aliases.rds  AnIndex  linmod.rdb  linmod.rdx  paths.rds
```

```
/home/chel/R/i686-pc-linux-gnu-library/2.12/linmod/html:
```

```
00Index.html
```

```
/home/chel/R/i686-pc-linux-gnu-library/2.12/linmod/Meta:
```

```
hsearch.rds  links.rds  package.rds  Rd.rds
```

```
/home/chel/R/i686-pc-linux-gnu-library/2.12/linmod/R:
```

```
linmod  linmod.rdb  linmod.rdx
```

```
chel@gnustats:~$
```

설치 후 테스트 해보기

- R에서 직접 수행해봅니다.

```
R> library(linmod)
R> data(cats, package="MASS")
R> linmod(Hwt~Bwt*Sex, data=cats)
Call:
linmod.formula(formula = Hwt ~ Bwt * Sex, data = cats)
```

Coefficients:

(Intercept)	Bwt	SexM	Bwt:SexM
2.981312	2.636414	-4.165400	1.676265

```
R> summary(linmod(Hwt~Bwt*Sex, data=cats))
```

```
Call:
linmod.formula(formula = Hwt ~ Bwt * Sex, data = cats)
```

	Estimate	StdErr	t.value	p.value
(Intercept)	2.98131	1.84284	1.6178	0.1079605
Bwt	2.63641	0.77590	3.3979	0.0008846 ***
SexM	-4.16540	2.06176	-2.0203	0.0452578 *
Bwt:SexM	1.67626	0.83733	2.0019	0.0472246 *

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

>



Question?

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

현재 내용을 사이팅하기 위해서는,

Friedrich Leisch. Creating R packages: A tutorial. In Paula Brito, editor, Compstat 2008 - Proceedings in Computational Statistics. Physica Verlag, Heidelberg, Germany, 2008.

다운로드 장소

현재 R Development Core Team에서는 한국어와 관련된 모든 자료는 **Open Statistics 코리아** 그룹 (R Development Local Team - Korea)에서 관리하도록 하였으며, 여러분들의 많은 지식공유 및 기부를 부탁드립니다. 본 자료의 다운로드는 <http://www.openstatistics.net> 을 방문해주세요.



References

Must-Know
Things In GNU
R
for Sharing
Knowledge

Friedrich
Leisch, Chel
Hee Lee

Introduction

Analytic
Solution

Objective
Oriented
Programming

S3 System

Interfacing
Formula

Packaging

References

Leisch, F. (2009). Creating R Packages: A Tutorial.

R Development Core Team (2010a). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0.

R Development Core Team (2010b). *Writing R Extensions*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-11-9.