The R Commander and Plug-ins

Fox, John
McMaster University, Department of Sociology
1280 Main Street West
Hamilton, Ontario, M5P 1B1, Canada
E-mail: jfox@mcmaster.ca

The **Rcmdr** (R Commander) package is a platform-independent graphical user interface (GUI) for R (R Development Core Team 2011, Ihaka and Gentleman 1996), based on the **tcltk** package (Dalgaard 2001, Dalgaard 2002) (distributed with R) and on the **Tcl/Tk** GUI builder (Welch and Jones 2003). Although GUIs for statistical software have serious limitations, the R Commander provides new and casual users of R, such as students in basic-statistics classes, the ability to access some of R's capabilities though a point-and-click interface.

Introductory statistics classes are notoriously difficult to teach, particularly to students, such as those in the social sciences, who often have weak mathematical backgrounds. Compounding this difficulty with the task of learning how to use command-driven software such as R risks distracting students from the essential ideas that are at the heart of elementary statistics. Conversely, providing such students with an intuitive and familiar point-and-click interface that eliminates typing errors, prevents certain elementary statistical errors (such as trying to compute the mean of a categorical variable), and doesn't tax their memory for commands, allows them to concentrate on essential course material.

Designed originally with basic-statistics courses in mind, the R Commander has expanded beyond that purpose to encompass extensive facilities for fitting linear, generalized-linear, and some other statistical models. The R Commander is also extensible through "plug-in packages," and a number of such packages have been contributed to the Comprehensive R Archive Network (CRAN). Along with the basic use of the R Commander, this presentation demonstrates one such package, the **RcmdrPlugin.survival** package (Fox and Carvalho 2011), which provides a GUI for many of the facilities of the **survival** package for survival analysis (Therneau 2011, Therneau and Grambsch 2000) which in turn is part of the standard R distribution.

Introducing the R Commander

The **Rcmdr** package is loaded in the normal manner, via the R command `library("Rcmdr")`. Doing so brings up the R Commander GUI, as illustrated in Fig. 1. The R Commander window consists of several elements, all meant to be largely self-explanatory (and which are substantially customizable):

- A menu bar consisting of several top-level menus. The contents of these menus, along with some selected submenus, are shown in Fig. 2. Menu items that are inappropriate in the current context (for example, at the start of the session, menus that require an "active" data set) are grayed-out.

- A toolbar, with a button displaying the active data set, and which can be used to select among data sets currently in memory; a button to edit the active data set; another button to view the active data set; and a button to display the active statistical model, which can be used to select among multiple statistical models associated with the current data set.

- A script window, which accumulates R commands generated by the R Commander. The script window is a rudimentary programming editor: The user can edit commands, type new commands, and submit commands for execution using the *Submit* button.

- An output window, in which commands and the printed output that they produce are accumulated.
- A messages window, which displays error messages, warnings, and notes, such as the start-up note at the beginning of a session. The *Edit* menu and standard keyboard shortcuts may be used within the script, output, and messages windows.

To get a sense of normal work-flow in the R Commander, consider the following exercise from David Moore's *Basic Practice of Statistics* (Moore 2007), which I assign to students in my basic-statistics course for sociology majors:

> "The Survey of Study Habits and Attitudes (SSHA) is a psychological test that measures the motivation, attitude toward school, and study habits of students. Scores range from 0 to 200. A selective private college gives the SSHA to an SRS of both male and female first-year students...."

Moore goes on to give scores for 18 female and 20 male students, and to suggest that the reader carry out a two-sample difference-of-means t-test on the data. The data are also available on a CD distributed with Moore's text, in a plain-text file named ex19-34.dat. The contents of this file, with most lines elided (as indicated by the elipses, . . .) are shown in Fig. 3.

The first step is to read the data into R, using the R Commander menu sequence *Data* → *Import data* → *from text file, clipboard, or URL...*, which brings up the dialog box shown in Fig. 4. The R Commander is capable of reading data from a variety of file types and formats, but a particularly simple choice is a plain-text file with data values separated by white space, such as ex19-34.dat. I have retained all of the defaults in the dialog box, except I replaced the default name of the data set (`Dataset`) with the more descriptive `SSHA`. Clicking *OK*, which is the default button in the dialog, brings up a standard open-file dialog, allowing me to navigate to and select the data file. Once read, the data in the file become the active data set in the R Commander.

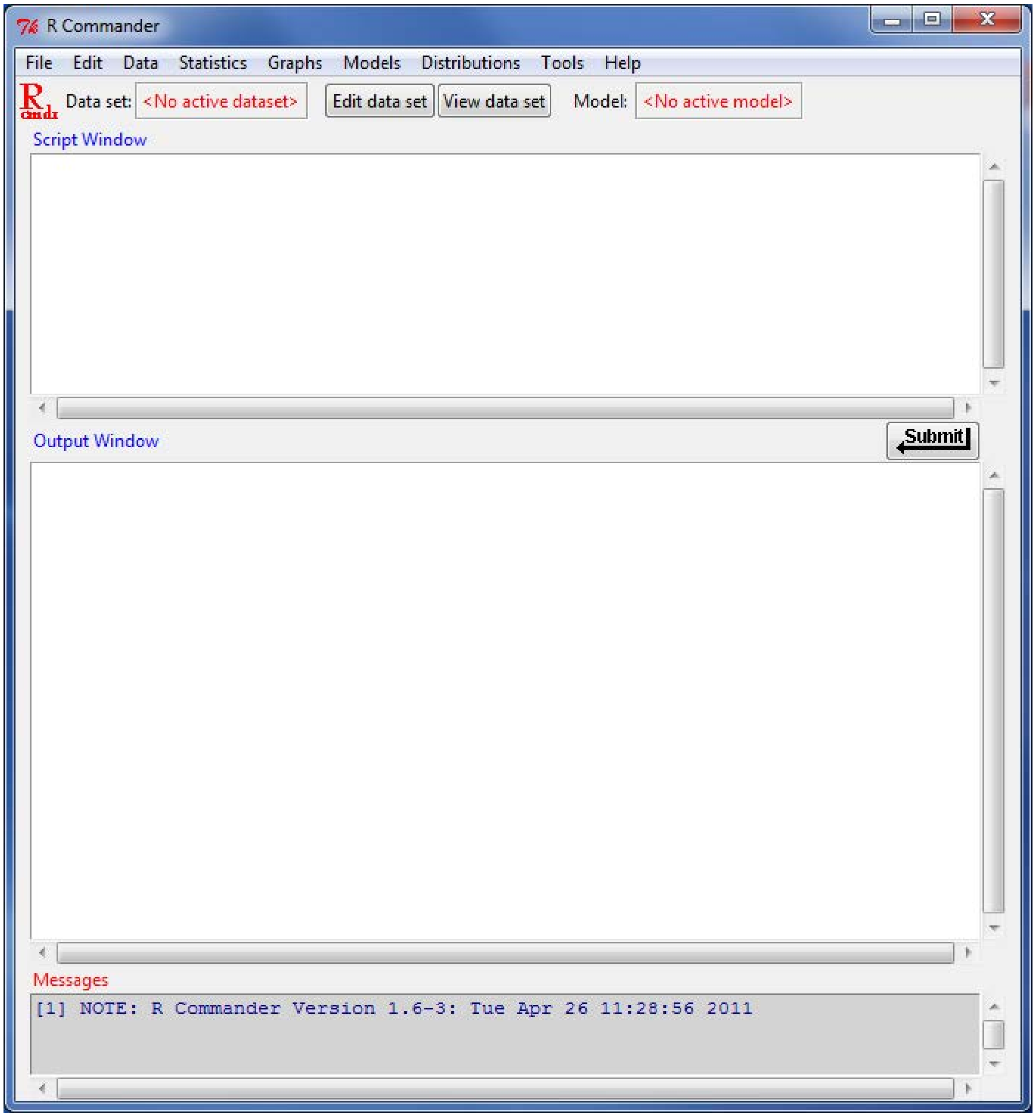
Then, to perform a difference of means t-test, I select the menu sequence *Statistics* → *Means* → *Independent samples t-test...*, which produces the dialog box shown in Fig. 5. Because there is just one two-level factor in the data set and just one numeric variable, these are preselected in the *Groups* and *Response Variable* list boxes. Moore suggests a two-sided test and prefers not to assume equal group variances, and so all the defaults in the dialog box are appropriate for this problem. Clicking *OK* updates the output window, which now contains the following accumulated contents:

```
> SSHA <- read.table("F:/PCDataSets/PC-Ascii/ex19-34.dat", header=TRUE,
+   sep="", na.strings="NA", dec=".", strip.white=TRUE)

> t.test(ssha~sex, alternative='two.sided', conf.level=.95, var.equal=FALSE,
+   data=SSHA)

Welch Two Sample t-test

data:  ssha by sex
t = 2.0561, df = 35.587, p-value = 0.04717
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  0.2616367 39.3494744
sample estimates:
```



```
[1] NOTE: R Commander Version 1.6-3: Tue Apr 26 11:28:56 2011
```

Figure 1: The main R Commander window at start-up.

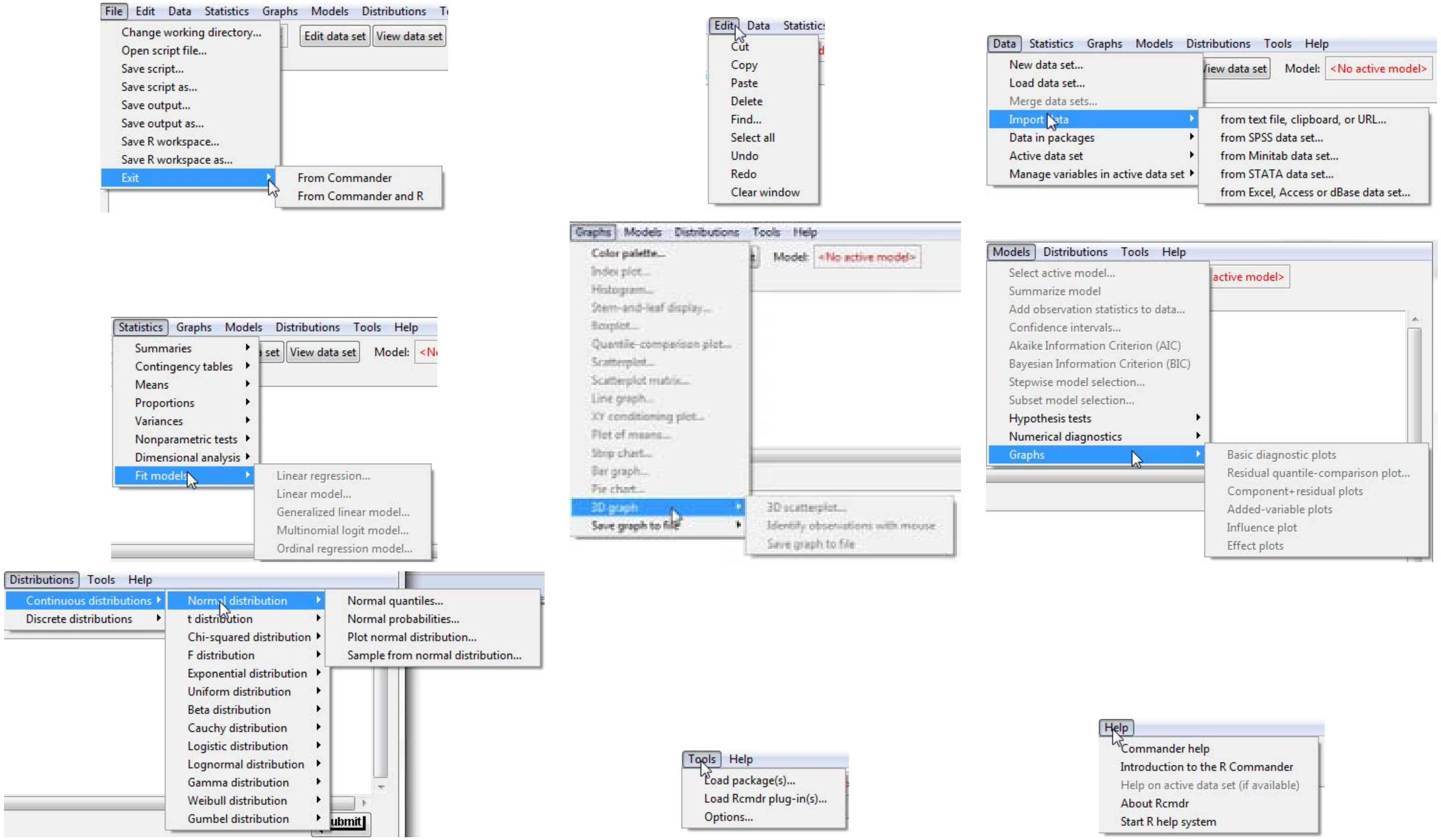
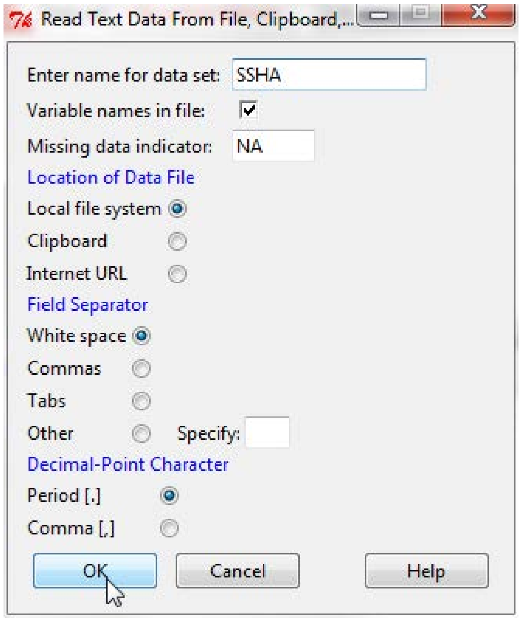


Figure 2: The R Commander top-level menus, with some sub-menus expanded.

```
sex ssha
F 154
F 109
F 137
F 115
. . .
M 115
M 187
M 104
```

Figure 3: The contents of the data file ex19-34.dat from Moore (2007), with most lines elided.



Read Text Data From File, Clipboard,...

Enter name for data set: SSHA

Variable names in file: ☑

Missing data indicator: NA

Location of Data File

Local file system ●

Clipboard ○

Internet URL ○

Field Separator

White space ●

Commas ○

Tabs ○

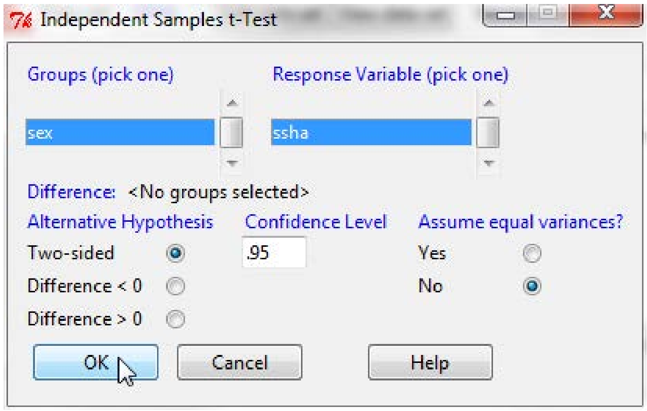
Other ○ Specify:

Decimal-Point Character

Period [.] ●

Comma [,] ○

Figure 4: R Commander dialog box for reading data from a text file.


Independent Samples t-Test

Groups (pick one)
sex

Response Variable (pick one)
ssha

Difference: <No groups selected>

Alternative Hypothesis
Two-sided ●
Difference < 0 ○
Difference > 0 ○

Confidence Level
.95

Assume equal variances?
Yes ○
No ●

Figure 5: R Commander dialog box for a two-sample t-test.

```
mean in group F mean in group M
       141.0556        121.2500
```

Fitting statistical models in the R Commander

As mentioned, the R Commander incorporates facilities for fitting a variety of statistical models: linear models, via the `lm` function in R; generalized linear models, via `glm`; multinomial logit models, via `multinom` in the **nnet** package (Venables and Ripley 2002); and ordinal regression models, via `polr` in the **MASS** package (Venables and Ripley 2002). As I illustrate below with the **RcmdrPlugin.survival** package, plug-in packages can expand the set of statistical models that the R Commander recognizes. The last statistical model fit to the current data set becomes the "active model," to which a variety of operations in the *Models* menu can be applied.

I illustrate using data from the 1994 wave of the Canadian Survey of Labour and Income Dynamics (the "SLID"), for the province of Ontario. This data set is included in the **car** package (Fox and Weisberg 2011), which is loaded when the **Rcmdr** starts up, and may be read into memory and made the active data set via the menu sequence *Data* → *Data in packages* → *Read data set from an attached package...*. To fit a linear model to the data, I select *Statistics* → *Fit models* → *Linear model...*, yielding the dialog box shown in Fig. 6. The buttons and variable-list box in the dialog assist in constructing an R model formula to represent the model, and it is also possible to type directly in the model-formula boxes in the dialog. Clicking the *OK* button produces the following output:

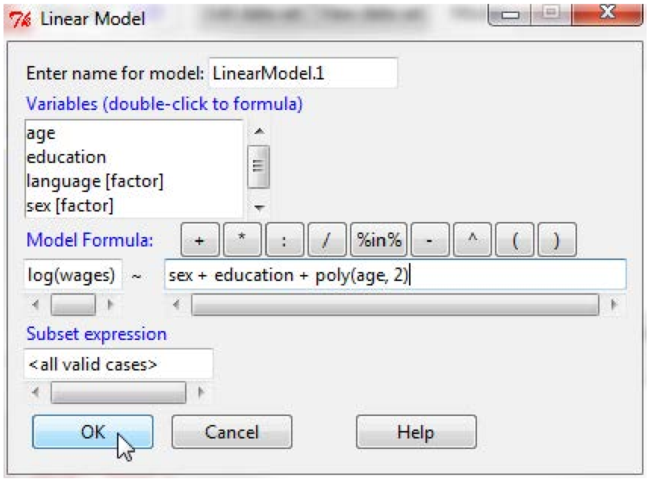
```
> LinearModel.1 <- lm(log(wages) ~ sex + education + poly(age, 2), data=SLID)

> summary(LinearModel.1)

Call:
lm(formula = log(wages) ~ sex + education + poly(age, 2), data = SLID)

Residuals:
     Min       1Q   Median       3Q      Max
-2.03897 -0.24103  0.02166  0.25546  1.78118

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    1.84230    0.02913  63.243   <2e-16 ***
```



Enter name for model: LinearModel.1

log(wages) ~ sex + education + poly(age, 2)

Subset expression
<all valid cases>

Figure 6: R Commander dialog box for specifying a linear model.

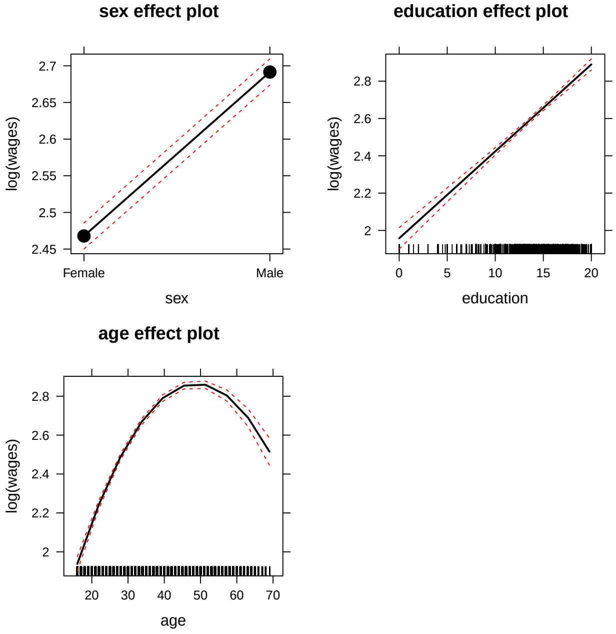


Figure 7: Effect plots for the regression model fit to the SLID data.

```
sex[T.Male]      0.22370    0.01252  17.873   <2e-16 ***
education        0.04659    0.00211  22.081   <2e-16 ***
poly(age, 2)1    3.04448    1.35655   2.244   0.0249 *
poly(age, 2)2 -24.59280    1.13745 -21.621   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3963 on 4009 degrees of freedom
  (3411 observations deleted due to missingness)
Multiple R-squared: 0.3815,Adjusted R-squared: 0.3809
F-statistic: 618.3 on 4 and 4009 DF,  p-value: < 2.2e-16
```

`LinearModel.1` is now the active model, and so, for example, I can draw “effect plots” for the model (Fox 2003) via *Models* → *Graphs* → *Effect plots*, producing Fig. 7.

Using R Commander plug-ins: The RcmdrPlugin.survival package

R Commander plug-in packages can either be loaded from the R Commander *Tools* menu or directly in a fresh R session by the library command. In the first case, the R Commander GUI is restarted, while in the second, the **Rcmdr** package is loaded automatically along with the plug-in. When the **RcmdrPlugin.survival** plug-in starts up, for example, it adds the menus and menu items shown in Fig. 8 to the standard R Commander interface.

To illustrate some of the capabilities added to the R Commander by the **RcmdrPlugin.survival** package, I will use a data set on criminal recidivism distributed with this package, called Rossi. The data, which originate with Rossi et al. (1980), and which are employed by Allison (1995) to illustrate survival analysis, include the following variables, among others:

week of first arrest after release or censoring; all censored observations are censored at 52 weeks.

arrest Event indicator, 1 if arrested, 0 if not arrested.

fin Financial aid provided upon release, a randomized treatment; no or yes.

age in years at time of release.

race Coded as black or other.

wexp Full-time work experience before incarceration, no or yes.

mar Marital status at time of release, married or not married.

paro Released on parole? no or yes.

prio Number of convictions prior to current incarceration.

educ Level of education: coded 2 = 6th grade or less; 3 = 7th to 9th grade; 4 = 10th to 11th grade; 5 = 12th grade; 6 = some college.

Following Allison (1995), and employing the menu sequence *Statistics* → *Fit models* → *Cox regression model...*, which produces the dialog in Fig. 9, I specify a Cox regression of week to rearrest, with arrest as the event indicator, on the other variables, producing the following output:

```
> CoxModel.2 <- coxph(Surv(week,arrest) ~ age + educ + fin + mar + paro +
+   prio + race + wexp, method="efron", data=Rossi)

> summary(CoxModel.2)
Call:
coxph(formula = Surv(week, arrest) ~ age + educ + fin + mar +
    paro + prio + race + wexp, data = Rossi, method = "efron")

  n= 432, number of events= 114

                        coef exp(coef) se(coef)      z Pr(>|z|)
age                 -0.05768   0.94395  0.02187 -2.638  0.00835 **
educ                -0.18578   0.83046  0.13153 -1.412  0.15782
fin[T.yes]          -0.35963   0.69794  0.19180 -1.875  0.06079 .
mar[T.not married]   0.42496   1.52953  0.38209  1.112  0.26605
paro[T.yes]         -0.08991   0.91401  0.19568 -0.459  0.64589
```

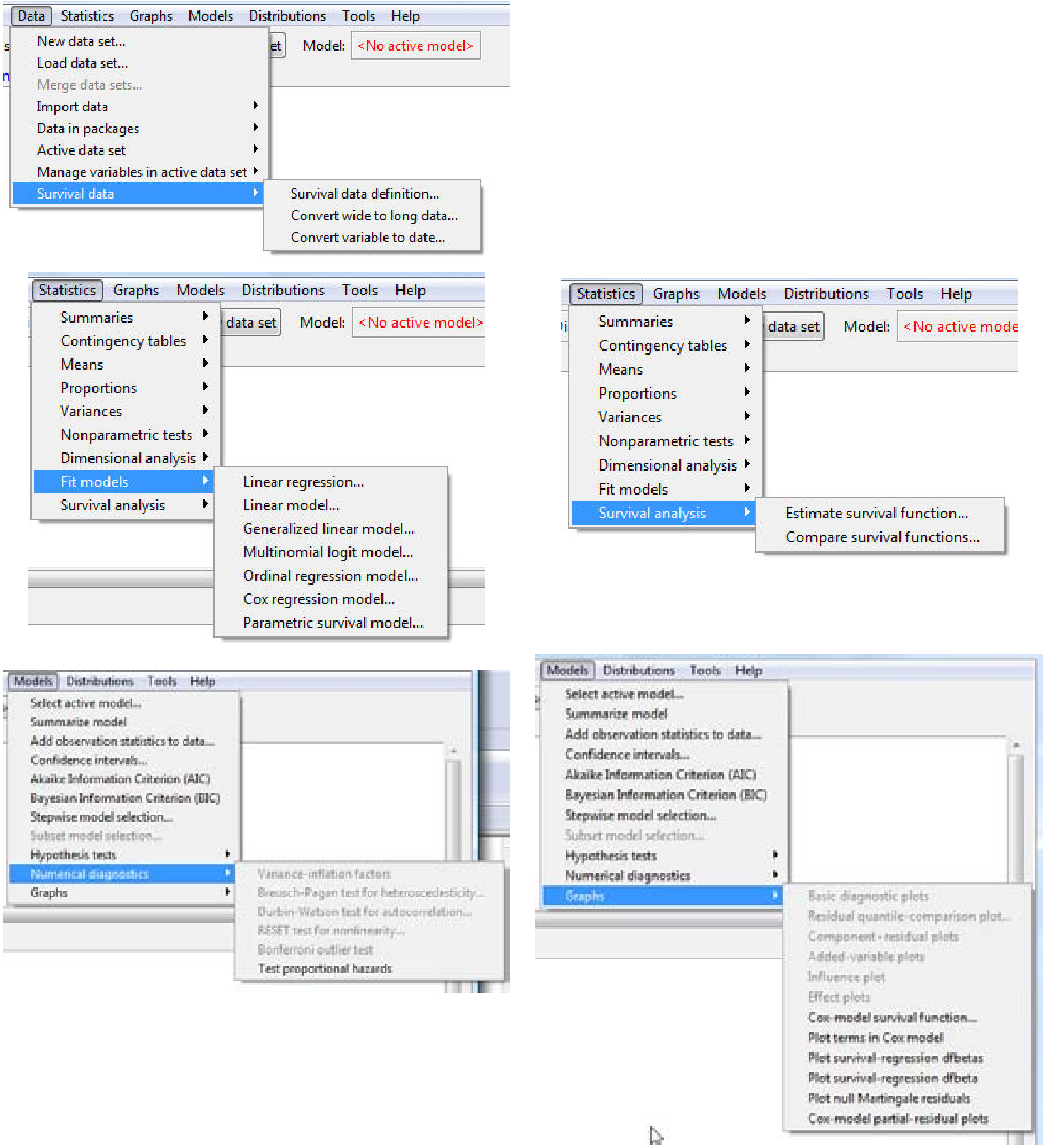


Figure 8: Menus and menu items added to the standard R Commander menus by the **RcmdrPlugin.survival** package.

```
prio                 0.08469   1.08838  0.02919  2.902  0.00371 **
race[T.other]       -0.34554   0.70784  0.30907 -1.118  0.26356
wexp[T.yes]         -0.11439   0.89191  0.21311 -0.537  0.59145
---
Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1

                   exp(coef) exp(-coef) lower .95 upper .95
age                   0.9440     1.0594    0.9044    0.9853
educ                  0.8305     1.2042    0.6417    1.0747
fin[T.yes]            0.6979     1.4328    0.4792    1.0164
mar[T.not married]    1.5295     0.6538    0.7233    3.2344
paro[T.yes]           0.9140     1.0941    0.6229    1.3413
prio                  1.0884     0.9188    1.0279    1.1525
race[T.other]         0.7078     1.4128    0.3862    1.2972
wexp[T.yes]           0.8919     1.1212    0.5874    1.3543

Concordance= 0.656  (se = 0.027 )
Rsquare= 0.079   (max possible= 0.956 )
Likelihood ratio test= 35.35  on 8 df,   p=2.31e-05
Wald test            = 33.74  on 8 df,   p=4.529e-05
Score (logrank) test = 35.1  on 8 df,   p=2.568e-05
```

Because `CoxModel.2` is now the active model in the R Commander, it is possible to manipulate via the *Models* menu. For example, *Models* → *Graphs* → *Cox model survival function...* leads to the dialog box shown in Fig. 10. I’ve filled in typical values of the covariates, with the exception of `fin` (provision of financial aid), which I’ve set alternatively to `yes` and `no`. The resulting graph is displayed in Fig. 11.

REFERENCES

P D. Allison, 1995. *Survival Analysis Using the SAS System: A Practical Guide.* Cary, NC: SAS Institute.

P. Dalgaard, 2001. A primer on the R-Tcl/Tk package. *R News* 1(3): 27–31.

P. Dalgaard, 2002. Changes to the R-Tcl/Tk package. *R News* 2(3): 25–27.

J. Fox, 2003. Effect displays in R for generalised linear models. *Journal of Statistical Software*, 8(15): 1–27.

J. Fox and M. Sá Carvalho, 2011. The RcmdrPlugin.survival package: Extending the R Commander interfact to survival analysis. *Journal of Statistical Software*, submission.

J. Fox and S. Weisberg, 2011. *An R Companion to Applied Regression, Second Edition.* Thousand Oaks CA: Sage.

R. Ihaka and R. Gentleman, 1996. R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5: 299–314.

D. S. Moore, 2007. *The Basic Practice of Statistics, Fourth Edition.* New York: Freeman.

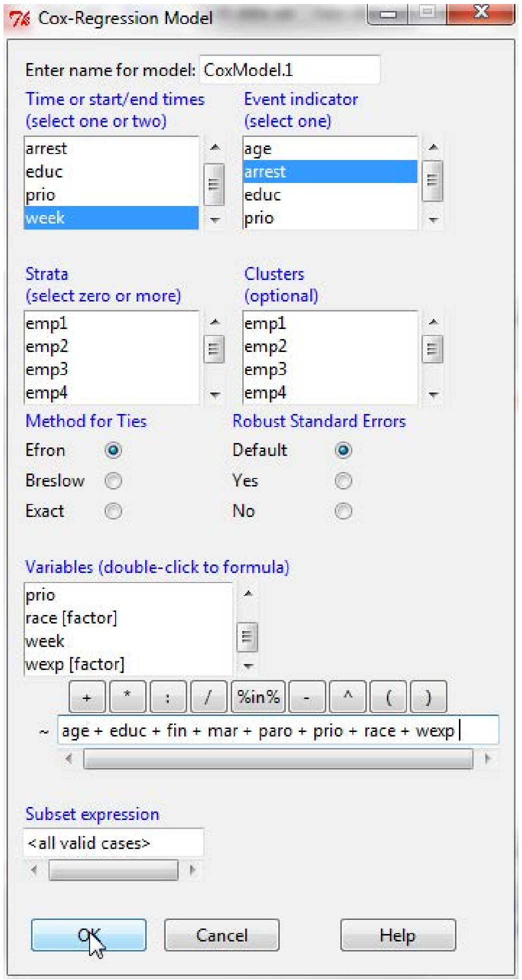
R Development Core Team, 2011. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

P. H. Rossi, R. A. Berk, and K. J. Lenihan, 1980. *Money, Work, and Crime: Some Experimental Results.* New York: Academic Press.

T. Therneau, 2011. survival: Survival analysis including penalized likelihood. R package version 2.36-8. URL: http://CRAN.R-project.org/package=survival.

T. M. Therneau and P. M. Grambsch, 2000. *Modeling Survival Data: Extending the Cox Model.* New York: Springer.

W. N. Venables and B. D. Ripley, 2002. *Modern Applied Statistics with S, Fourth Editin.* New York:



Cox-Regression Model

Enter name for model: CoxModel.1

Time or start/end times (select one or two)
- arrest
- educ
- prio
- week (selected)

Event indicator (select one)
- age
- arrest (selected)
- educ
- prio

Strata (select zero or more)
- emp1
- emp2
- emp3
- emp4

Clusters (optional)
- emp1
- emp2
- emp3
- emp4

Method for Ties
- Efron ●
- Breslow ○
- Exact ○

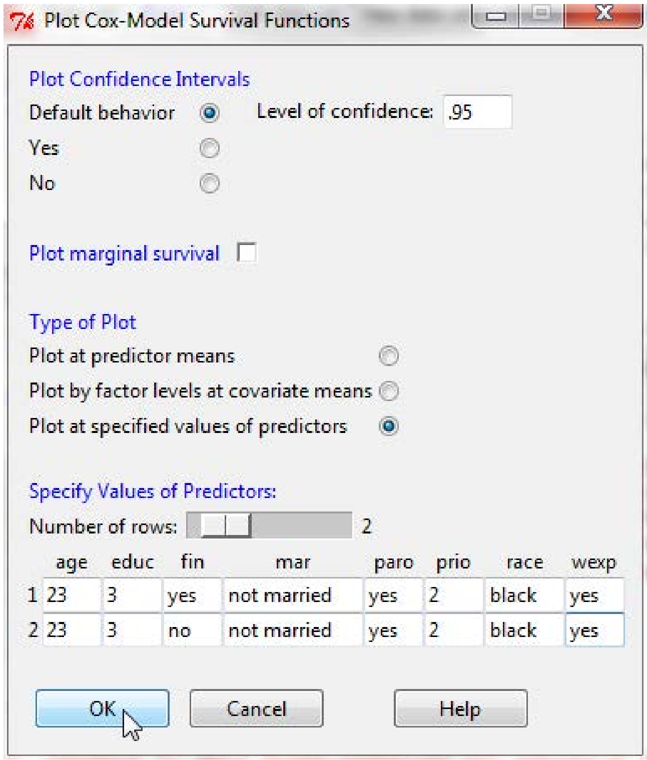
Robust Standard Errors
- Default ●
- Yes ○
- No ○

Variables (double-click to formula)
- prio
- race [factor]
- week
- wexp [factor]

~ age + educ + fin + mar + paro + prio + race + wexp

Subset expression: <all valid cases>

Figure 9: R Commander dialog box for specifying a Cox regression model.



Plot Cox-Model Survival Functions

Plot Confidence Intervals
- Default behavior ●
- Yes ○
- No ○

Level of confidence: .95

Plot marginal survival □

Type of Plot
- Plot at predictor means ○
- Plot by factor levels at covariate means ○
- Plot at specified values of predictors ●

Specify Values of Predictors:

Number of rows: 2

	age	educ	fin	mar	paro	prio	race	wexp
1	23	3	yes	not married	yes	2	black	yes
2	23	3	no	not married	yes	2	black	yes

Figure 10: Dialog box for graphing estimated Cox-regression survival functions.

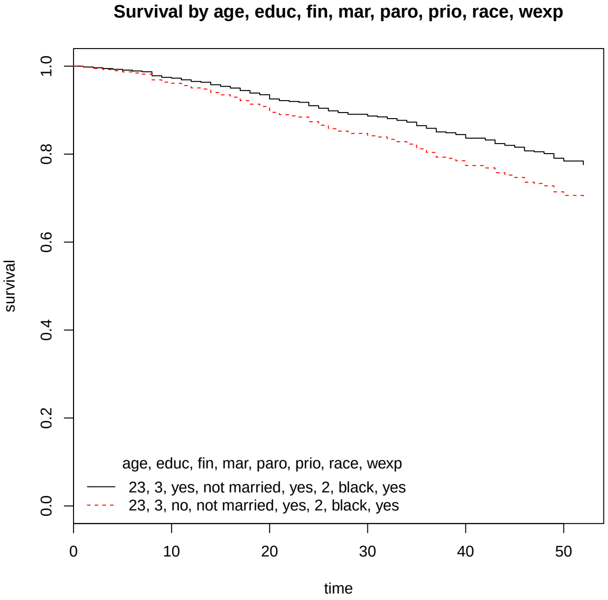


Figure 11: Estimated survival functions for those receiving and not receiving financial aid, evaluated at typical values of the other covariates.

Springer.

B. Welsch and K. Jones, 2003. *Practical Programming in Tcl and Tk, Fourth Edition.* Upper Saddle River, NJ: Prentice Hall PTR.

ABSTRACT

The **Rcmdr** *(*R Commander*) package is a platform-independent graphical user interface (GUI) for* R*, based on the* **tcltk** *package and on the* **Tcl/Tk** *GUI builder. Although GUIs for statistical software have serious limitations, the* R Commander *provides new and casual users of* R*, including students in basic-statistics courses, the ability to access some of* R*'s capabilities though a point-and-click interface. The* R Commander *has expanded beyond its initial purpose of supporting basic-statistics courses to encompass extensive facilities for fitting linear, generalized-linear, and some other statistical models. The* R Commander *is also extensible through "plug-in packages," and a number of such packages have been contributed to the Comprehensive R Archive Network (CRAN). Along with the basic use of the* R Commander*, this presentation demonstrates one such package, the* **RcmdrPlugin.survival** *package, which provides a GUI for many of the facilities of the* **survival** *package for survival analysis.*