The Other Packages Gallery

David J. Scott

January 4, 2018

Contents

1 Introduction

This document represents a test of the functions in **xtable** which deal with other packages.

The first step is to load the package and set some options for this document.

```
library(xtable)
options(xtable.floating = FALSE)
options(xtable.timestamp = "")
options(width = 60)
set.seed(1234)
```

2 The packages spdep, splm, and sphet

Code for supporting these packages and most of the examples used in this section was originally provided by Martin Gubri (`martin.gubri@framasoft.org`).

2.1 The package spdep

First load the package and create some objects.

```
library(spdep)
```

```
## Loading required package:  sp
```

```
## Loading required package:  Matrix
```

```
## Loading required package:  spData
```

```
## To access larger datasets in this package, install
## the spDataLarge package with:
## `install.packages('spDataLarge')`
```

```
data("oldcol", package = "spdep")

data.in.sample <- COL.OLD[1:44,]
data.out.of.sample <- COL.OLD[45:49,]

listw.in.sample <- nb2listw(subset(COL.nb, !(1:49 %in% 45:49)))
listw.all.sample <- nb2listw(COL.nb)

COL.lag.eig <- lagsarlm(CRIME ~ INC + HOVAL, data = data.in.sample,
                        listw.in.sample)
class(COL.lag.eig)
```

```
## [1] "sarlm"
```

```
COL.errW.GM <- GMerrorsar(CRIME ~ INC + HOVAL, data = data.in.sample,
                          listw.in.sample, returnHcov = TRUE)
class(COL.errW.GM)
```

```
## [1] "gmsar"
```

```
COL.lag.stsls <- stsls(CRIME ~ INC + HOVAL, data = data.in.sample,
                       listw.in.sample)
class(COL.lag.stsls)
```

```
## [1] "stsls"
```

```
p1 <- predict(COL.lag.eig, newdata = data.out.of.sample,
              listw = listw.all.sample)
class(p1)
```

```
## [1] "sarlm.pred"
```

```
p2 <- predict(COL.lag.eig, newdata = data.out.of.sample,
              pred.type = "trend", type = "trend")
#type option for retrocompatibility with spdep 0.5-92
class(p2)
```

```
## [1] "sarlm.pred"
```

```
imp.exact <- impacts(COL.lag.eig, listw = listw.in.sample)
class(imp.exact)
```

```
## [1] "lagImpact"
```

```
imp.sim <- impacts(COL.lag.eig, listw = listw.in.sample, R = 200)
class(imp.sim)
```

```
## [1] "lagImpact"
```

sarlm objects

There is an `xtable` method for objects of this type.

```
xtable(COL.lag.eig)
```

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	49.3088	8.0720	6.11	0.0000
INC	-1.0475	0.3446	-3.04	0.0024
HOVAL	-0.2886	0.0993	-2.91	0.0036

The method for `xtable` actually uses the summary of the object, and an identical result is obtained when using the summary of the object, even if the summary contains more additional information.

```
xtable(summary(COL.lag.eig, correlation = TRUE))
```

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	49.3088	8.0720	6.11	0.0000
INC	-1.0475	0.3446	-3.04	0.0024
HOVAL	-0.2886	0.0993	-2.91	0.0036

This same pattern applies to the other objects from this group of packages.

Note that additional prettying of the resulting table is possible, as for any table produced using `xtable`. For example using the **booktabs** package we get:

```
print(xtable(COL.lag.eig), booktabs = TRUE)
```

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	49.3088	8.0720	6.11	0.0000
INC	-1.0475	0.3446	-3.04	0.0024
HOVAL	-0.2886	0.0993	-2.91	0.0036

gmsar objects

```
xtable(COL.errW.GM)
```

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	63.5692	5.3326	11.92	0.0000
INC	-1.0568	0.3701	-2.86	0.0043
HOVAL	-0.3283	0.1038	-3.16	0.0016

stsls objects

```
xtable(COL.lag.stsls)
```

	Estimate	Std. Error	t value	Pr(>|t|)
Rho	0.3085	0.2161	1.43	0.1534
(Intercept)	51.6851	12.5261	4.13	0.0000
INC	-1.0963	0.4192	-2.62	0.0089
HOVAL	-0.2901	0.1043	-2.78	0.0054

sarlm.pred objects

`xtable` has a method for predictions of `sarlm` models.

```
xtable(p1)
```

	fit	trend	signal
1045	32.12	22.39	9.74
1046	27.62	19.20	8.42
1047	26.51	18.53	7.99
1048	17.69	8.15	9.54
1049	19.15	9.41	9.74

This method transforms the `sarlm.pred` objects into data frames, allowing any number of attributes vectors which may vary according to predictor types.

```
xtable(p2)
```

	fit
1045	22.39
1046	19.20
1047	18.53
1048	8.15
1049	9.41

lagImpact objects

The `xtable` method returns the values of direct, indirect and total impacts for all the variables in the model. The class `lagImpact` has two different sets of attributes according to if simulations are used. But the `xtable` method always returns the three components of the non-simulation case.

```
xtable(imp.exact)
```

	Direct	Indirect	Total
INC	-1.08	-0.54	-1.62
HOVAL	-0.30	-0.15	-0.45

```
xtable(imp.sim)
```

	Direct	Indirect	Total
INC	-1.08	-0.54	-1.62
HOVAL	-0.30	-0.15	-0.45

spautolm objects

The need for an `xtable` method for `spautolm` was expressed by Guido Schulz (`schulzgu@student.hu-berlin.de`), who also provided an example of an object of this type. The required code was implemented by David Scott (`d.scott@auckland.ac.nz`).

First create an object of the required type.

```
library(spdep)
example(NY_data)
spautolmOBJECT <- spautolm(Z ~ PEXPOSURE + PCTAGE65P,data = nydata,
                           listw = listw_NY, family = "SAR",
                           method = "eigen", verbose = TRUE)
summary(spautolmOBJECT, Nagelkerke = TRUE)
```

```
class(spautolmOBJECT)
```

```
## [1] "spautolm"
```

```
xtable(spautolmOBJECT,
       display = c("s",rep("f", 3), "e"), digits = 4)
```

	Estimate	Std. Error	z value	Pr(>|z|)
(Intercept)	-0.9128	0.1258	-7.2541	4.0457e-13
PEXPOSURE	0.0915	0.0453	2.0204	4.3346e-02
PCTAGE65P	3.8294	0.6313	6.0655	1.3151e-09

2.2 The package splm

First load the package and create some objects.

```
library(splm)
data("Produc", package = "plm")
data("usaww",  package = "splm")
fm <- log(gsp) ~ log(pcap) + log(pc) + log(emp) + unemp
respatlag <- spml(fm, data = Produc, listw = mat2listw(usaww),
                  model="random", spatial.error="none", lag=TRUE)
class(respatlag)
```

```
## [1] "splm"
```

```
GM <- spgm(log(gsp) ~ log(pcap) + log(pc) + log(emp) + unemp, data = Produc,
           listw = usaww, moments = "fullweights", spatial.error = TRUE)
class(GM)
```

```
## [1] "splm"
```

```
imp.spml <- impacts(respatlag, listw = mat2listw(usaww, style = "W"), time = 17)
class(imp.spml)
```

```
## [1] "lagImpact"
```

splm objects

```
xtable(respatlag)
```

	Estimate	Std. Error	t-value	Pr(>|t|)
(Intercept)	1.6581	0.1507	11.00	0.0000
log(pcap)	0.0129	0.0249	0.52	0.6037
log(pc)	0.2256	0.0216	10.43	0.0000
log(emp)	0.6708	0.0264	25.39	0.0000
unemp	-0.0058	0.0009	-6.50	0.0000

```
xtable(GM)
```

	Estimate	Std. Error	t-value	Pr(>|t|)
log(pcap)	0.0043	0.0253	0.17	0.8652
log(pc)	0.2145	0.0233	9.22	0.0000
log(emp)	0.7831	0.0280	27.99	0.0000
unemp	-0.0026	0.0011	-2.43	0.0152

The `xtable` method works the same on impacts of `splm` models.

```
xtable(imp.spml)
```

	Direct	Indirect	Total
log(pcap)	0.01	0.00	0.02
log(pc)	0.23	0.04	0.27
log(emp)	0.68	0.12	0.80
unemp	-0.01	-0.00	-0.01

2.3 The package sphet

First load the package and create some objects.

```
library(sphet)
```

```
##
## Attaching package:  'sphet'
```

```
## The following object is masked from 'package:splm':
##
##     listw2dgCMatrix
```

```
data("columbus", package = "spdep")
listw <- nb2listw(col.gal.nb)
data("coldis", package = "sphet")
res.stsls <- stslshac(CRIME ~ HOVAL + INC, data = columbus, listw = listw,
                      distance = coldis, type = 'Triangular')
class(res.stsls)
```

```
## [1] "sphet" "stsls"
```

```
res.gstsls <- gstslshet(CRIME ~ HOVAL + INC, data = columbus, listw = listw)
class(res.gstsls)
```

```
## [1] "sphet"  "gstsls"
```

```
imp.gstsls <- impacts(res.gstsls, listw = listw)
class(imp.gstsls)
```

```
## [1] "lagImpact"
```

sphet objects

```
xtable(res.stsls)
```

	Estimate	SHAC St.Er.	t-value	Pr(>|t|)
Wy	0.4546	0.1651	2.75	0.0059
(Intercept)	44.1164	8.0757	5.46	0.0000
HOVAL	-0.2695	0.1787	-1.51	0.1315
INC	-1.0077	0.5119	-1.97	0.0490

```
xtable(res.gstsls)
```

	Estimate	Std. Error	t-value	Pr(>|t|)
(Intercept)	44.1241	7.5067	5.88	0.0000
HOVAL	-0.2756	0.1769	-1.56	0.1193
INC	-0.9875	0.4602	-2.15	0.0319
lambda	0.4529	0.1433	3.16	0.0016
rho	0.0648	0.3051	0.21	0.8317

`sphet` also provides a method for computing impacts.

```
xtable(imp.gstsls)
```

	Direct	Indirect	Total
HOVAL	-0.29	-0.21	-0.50
INC	-1.05	-0.76	-1.80

3 The zoo package

```
library(zoo)

##
## Attaching package:  'zoo'

## The following objects are masked from 'package:base':
##
##    as.Date, as.Date.numeric

xDate <- as.Date("2003-02-01") + c(1, 3, 7, 9, 14) - 1
as.ts(xDate)
```

Time Series: Start = 1 End = 5 Frequency = 1 [1] 12084 12086 12090 12092 12097

```
x <- zoo(rnorm(5), xDate)
xtable(x)
```

	Value
12084	0.54
12085	
12086	0.53
12087	
12088	
12089	
12090	-0.71
12091	
12092	0.06
12093	
12094	
12095	
12096	
12097	-0.91

```
tempTs <- ts(cumsum(1 + round(rnorm(100), 0)),
             start = c(1954, 7), frequency = 12)
tempTable <- xtable(tempTs, digits = 0)
tempTable
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1954							0	-1	1	1	3	4
1955	7	8	11	12	14	15	16	17	18	20	21	20
1956	22	22	24	25	26	28	28	29	29	31	32	32
1957	33	33	33	34	33	36	37	36	36	37	36	37
1958	38	40	43	45	47	48	50	51	54	55	57	58
1959	60	61	62	63	67	67	69	70	72	71	71	72
1960	72	73	75	76	76	77	80	83	84	87	89	91
1961	91	90	92	92	92	93	92	94	96	97	96	96
1962	96	98	100	101	102	103	105	107	110	110		

```
tempZoo <- as.zoo(tempTs)
xtable(tempZoo, digits = 0)
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1954							0	-1	1	1	3	4
1955	7	8	11	12	14	15	16	17	18	20	21	20
1956	22	22	24	25	26	28	28	29	29	31	32	32
1957	33	33	33	34	33	36	37	36	36	37	36	37
1958	38	40	43	45	47	48	50	51	54	55	57	58
1959	60	61	62	63	67	67	69	70	72	71	71	72
1960	72	73	75	76	76	77	80	83	84	87	89	91
1961	91	90	92	92	92	93	92	94	96	97	96	96
1962	96	98	100	101	102	103	105	107	110	110		

4 The survival package

```
library(survival)
test1 <- list(time=c(4,3,1,1,2,2,3),
              status=c(1,1,1,0,1,1,0),
              x=c(0,2,1,1,1,0,0),
              sex=c(0,0,0,0,1,1,1))
coxFit <- coxph(Surv(time, status) ~ x + strata(sex), test1)
xtable(coxFit)
```

	coef	exp(coef)	se(coef)	z	p
x	0.80	2.23	0.82	0.98	0.33