Introduction to analysis of aggregate data in the package apc

25 August 2020

Bent Nielsen   Department of Economics, University of Oxford
& Nuffield College
bent.nielsen@nuffield.ox.ac.uk
http://users.ox.ac.uk/~nuff0078

Contents

1 Introduction

The purpose of this vignette is give an introduction the age-period-cohort analysis of tables of aggregate data using R package `apc`.

The apc package uses the canonical parameters suggested by Kuang, Nielsen and Nielsen (2008a) and generalized by Nielsen (2014). These evolve around the second differences of age, period and cohort factors as well as an three parameters (level and two slopes) for a linear plane. The age, period and cohort factors themselves are not identifiable. They could be ad hoc identified by associating the levels and two slopes to the age, period and cohort factors in a particular way. This should be done with great care as such ad hoc identification easily masks which information is coming from the data and which information is coming from the choice of ad hoc identification scheme. An illustration is given below. A short description of the package can be found in Nielsen (2015).

2 Analysis of Belgian lung cancer data

The very first step is to call the package.

```
> library(apc)
```

The data

The data set is taken from table VIII of Clayton and Schifflers (1987a), which contains age-specific incidence rates (per 100,000 person-years observation) of lung cancer in Belgian females during the period 1955-1978. Numerators are also available. The original source was the WHO mortality database.

The package uses a special data format for aggregate data, where the data is kept in a matrix format. This data format also keeps track of the labels of the different time scales. This comes in handy when listing parameters, when plotting and when seeking to truncate the data. The package does not use the vectorized format in the traditional data.frame in R.

The Belgian data are already in the apc.data.list format. Other data can be organized using the apc.data.list function.

```
> data.list   <- data.Belgian.lung.cancer()
> objects(data.list)

 [1] "age1"        "coh1"        "data.format" "dose"        "label"
 [6] "n.decimal"   "per.max"     "per.zero"    "per1"        "response"
[11] "time.adjust" "unit"

> data.list

$response
      1955-1959 1960-1964 1965-1969 1970-1974
```

```
25-29            3            2            7            3
30-34           11           16           11           10
35-39           11           22           24           25
40-44           36           44           42           53
45-49           77           74           68           99
50-54          106          131           99          142
55-59          157          184          189          180
60-64          193          232          262          249
65-69          219          267          323          325
70-74          223          250          308          412
75-79          198          214          253          338
```

$dose
```
      1955-1959 1960-1964 1965-1969 1970-1974
25-29 15.789474 15.384615 14.000000 15.789474
30-34 16.666667 16.326531 15.277778 14.084507
35-39 14.102564 16.666667 16.326531 15.243902
40-44 13.483146 13.924051 16.600791 15.680473
45-49 15.909091 13.214286 13.793103 16.363636
50-54 16.060606 15.411765 12.941176 13.408876
55-59 15.154440 15.333333 14.905363 12.552301
60-64 13.075881 14.172266 14.555556 14.147727
65-69 10.667316 11.814159 12.971888 13.357994
70-74  8.498476  9.025271 10.108303 11.153221
75-79  5.915745  6.367153  6.880609  7.736324
```

$data.format
[1] "AP"

$age1
[1] 25

$per1
[1] 1955

$coh1
[1] 5

$unit
[1] 5

$per.zero
NULL

$per.max

```
NULL

$time.adjust
[1] 0

$label
NULL

$n.decimal
NULL
```

Plot the data

The data analysis can be initiated by plotting the data. The package has a number of plots. All plot formats can be called with a single command. Some of plots involve grouping of data. A warning is produced because the defaults settings lead to an unbalanced grouping of data.

```
> apc.plot.data.all(data.list)

[1] "apc.plot.data.within warning: maximal index not divisible by thin, so last group
[1] "apc.plot.data.within warning: maximal index not divisible by thin, so last group
[1] "apc.plot.data.within warning: maximal index not divisible by thin, so last group
[1] "apc.plot.data.within warning: maximal index not divisible by thin, so last group
[1] "apc.plot.data.within warning: maximal index not divisible by thin, so last group
[1] "apc.plot.data.within warning: maximal index not divisible by thin, so last group
```

Alternatively, the plots can be called individually. The first plot contains data sums.

```
> graphics.off()
> apc.plot.data.sums(data.list)
```

A sparsity plot shows where data are thin. In this case, the plots are blank with default settings. We therefore change sparsity.limits.

```
> graphics.off()
> apc.plot.data.sparsity(data.list)
> apc.plot.data.sparsity(data.list,sparsity.limits=c(5,10))
```

The next plots visualize data using different pairs of the three time scales. These plots are done for mortality ratios. All plots appear to have approximately parallel lines. This indicates that interpretation should be done carefully.

```
> graphics.off()
> apc.plot.data.within.all.six(data.list,"m")

[1] "apc.plot.data.within warning: maximal index not divisible by thin, so last group
[1] "apc.plot.data.within warning: maximal index not divisible by thin, so last group
```

Get a deviance table

A deviance table is constructed. For this, we need to formulated the distribution of the model. The table show that the sub-models "AC" and "Ad" cannot be rejected relative to the unrestricted "APC" model

```
> apc.fit.table(data.list,"poisson.dose.response")
```

	deviance	df.residual	prob(>chi_sq)	LR vs.APC	df vs.APC	prob(>chi_sq)
APC	20.225	18	0.320	NaN	NaN	NaN
AP	25.558	30	0.697	5.333	12	0.946
AC	21.454	20	0.371	1.229	2	0.541
PC	99.228	27	0.000	79.004	9	0.000
Ad	26.584	32	0.737	6.359	14	0.957
Pd	253.562	39	0.000	233.337	21	0.000
Cd	100.712	29	0.000	80.487	11	0.000
A	85.577	33	0.000	65.352	15	0.000
P	6390.146	40	0.000	6369.921	22	0.000
C	1217.030	30	0.000	1196.805	12	0.000
t	254.518	41	0.000	234.293	23	0.000
tA	308.135	42	0.000	287.910	24	0.000
tP	6390.708	42	0.000	6370.483	24	0.000
tC	1612.070	42	0.000	1591.845	24	0.000
1	6499.777	43	0.000	6479.552	25	0.000

	aic
APC	341.397
AP	322.730
AC	338.625
PC	402.400
Ad	319.756
Pd	532.733
Cd	399.884
A	376.749
P	6667.318
C	1514.202
t	529.690
tA	581.307
tP	6663.879
tC	1885.241
1	6770.948

Estimate selected models

We consider the "APC" and "Ad" model. We also consider also the sub-model "A", which is not supported by the tests in the deviance table. We get the three fits

```
> fit.apc <- apc.fit.model(data.list,"poisson.dose.response","APC")
> fit.ad  <- apc.fit.model(data.list,"poisson.dose.response","Ad")
> fit.a   <- apc.fit.model(data.list,"poisson.dose.response","A")
```

The coefficients for canonical parameters are found through

```
> fit.apc$coefficients.canonical
```

	Estimate	Std. Error	z value	Pr(>|z|)
level	1.957545765	0.06587835	29.71455339	4.980891e-194
age slope	0.504384208	0.07522008	6.70544587	2.007923e-11
cohort slope	0.120878598	0.06799397	1.77778399	7.543934e-02
DD_age_35	-0.497116645	0.42748123	-1.16289700	2.448713e-01
DD_age_40	0.253907149	0.28839948	0.88040084	3.786422e-01
DD_age_45	-0.155115204	0.20515233	-0.75609770	4.495906e-01
DD_age_50	-0.205504949	0.15042595	-1.36615360	1.718908e-01
DD_age_55	-0.043344184	0.11872510	-0.36508022	7.150515e-01
DD_age_60	-0.092603145	0.09711954	-0.95349653	3.403386e-01
DD_age_65	0.023605714	0.08354890	0.28253770	7.775312e-01
DD_age_70	-0.046471233	0.07644644	-0.60789267	5.432587e-01
DD_age_75	-0.077331927	0.07619553	-1.01491426	3.101467e-01
DD_period_1965	-0.065187056	0.06656344	-0.97932225	3.274208e-01
DD_period_1970	0.064058271	0.06211963	1.03120825	3.024432e-01
DD_cohort_1890	0.089055820	0.12918107	0.68938756	4.905794e-01
DD_cohort_1895	0.022794960	0.09524872	0.23932039	8.108572e-01
DD_cohort_1900	-0.009887705	0.07806299	-0.12666316	8.992070e-01
DD_cohort_1905	-0.087602585	0.07716974	-1.13519348	2.562943e-01
DD_cohort_1910	0.070174649	0.08627345	0.81339796	4.159899e-01
DD_cohort_1915	0.005654086	0.10239118	0.05522044	9.559628e-01
DD_cohort_1920	0.015051099	0.12850964	0.11712039	9.067647e-01
DD_cohort_1925	-0.093529779	0.15857808	-0.58980270	5.553229e-01
DD_cohort_1930	0.191506367	0.20187869	0.94862102	3.428134e-01
DD_cohort_1935	-0.214529781	0.28443477	-0.75423192	4.507100e-01
DD_cohort_1940	0.160455202	0.43666159	0.36745894	7.132767e-01
DD_cohort_1945	-0.609263039	0.81479117	-0.74775361	4.546088e-01

```
> fit.ad$coefficients.canonical
```

	Estimate	Std. Error	z value	Pr(>|z|)
level	1.98218357	0.04901861	40.4373684	0.000000e+00
age slope	0.48079205	0.06025798	7.9788936	1.476508e-15
cohort slope	0.08871334	0.01158119	7.6601263	1.857504e-14
DD_age_35	-0.61853253	0.40273616	-1.5358257	1.245811e-01
DD_age_40	0.26923140	0.27445998	0.9809496	3.266176e-01
DD_age_45	-0.18763742	0.19544414	-0.9600565	3.370268e-01
DD_age_50	-0.16690515	0.14277769	-1.1689862	2.424092e-01

```
DD_age_55     -0.05010697 0.11368129 -0.4407671 6.593816e-01
DD_age_60     -0.08993029 0.09378346 -0.9589141 3.376020e-01
DD_age_65      0.02072825 0.08101869  0.2558452 7.980703e-01
DD_age_70     -0.04428379 0.07371656 -0.6007305 5.480195e-01
DD_age_75     -0.08566734 0.07233153 -1.1843706 2.362664e-01
```

Residual analysis

We get a number of plots to illustrate the fit. We plot estimators, probability transforms of responses given fit, residuals, fitted values, linear predictors, and data.

   In the probability transform plot: Black circle are used for central part of distribution. Triangles are used in tails, green/blue/red as responses are further in tail No sign of mis-specification for "APC" and "Ad": there are many black circles and only few coloured triangles. In comparison the model "A" yields more extreme observations. That model is not supported by the data. To get numerical values see apc.plot.fit.pt

```
> graphics.off()
> apc.plot.fit.all(fit.apc)

WARNING apc.plot.fit: sdv large for plot 5 - possibly not plotted

> apc.plot.fit.all(fit.ad)

WARNING apc.plot.fit: sdv large for plot 5 - possibly not plotted

> apc.plot.fit.all(fit.a)

WARNING apc.plot.fit: sdv large for plot 5 - possibly not plotted
```

Plot estimated coefficients for sub models

We consider the "APC", "Ad" and "A" models and plot the estimated coefficients The first row of plots show double differences of parameters The second row of plots shows level and slope determining a linear plane The third row shows double sums of double differences, all identified to be zero at the begining and at the end. Thus the plots in third row must be interpreted jointly with those in the second row. The interpretation of the third row plots is that they show deviations from linear trends. The third row plots are not invariant to changes to data array.

   For the "APC" and "Ad" the estimated coefficients are similar. For the "A" model the coefficients are different, reflecting the misspecification.

```
> graphics.off()
> apc.plot.fit(fit.apc)

WARNING apc.plot.fit: sdv large for plot 5 - possibly not plotted

> apc.plot.fit(fit.ad)

WARNING apc.plot.fit: sdv large for plot 5 - possibly not plotted

> apc.plot.fit(fit.a)

WARNING apc.plot.fit: sdv large for plot 5 - possibly not plotted
```

Recursive analysis

Cut the first period group and redo analysis

```
> data.list.subset.1 <- apc.data.list.subset(data.list,0,0,1,0,0,0)

WARNING apc.data.list.subset: cuts in arguments are:
[1] 0 0 1 0 0 0
have been modified to:
[1] 0 0 1 0 1 0
WARNING apc.data.list.subset: coordinates changed to "AC" & data.format changed to "t

> apc.fit.table(data.list.subset.1,"poisson.dose.response")
```

	deviance	df.residual	prob(>chi_sq)	LR vs.APC	df vs.APC	prob(>chi_sq)
APC	12.096	9	0.208	NaN	NaN	NaN
AP	20.876	20	0.404	8.780	11	0.642
AC	13.341	10	0.205	1.244	1	0.265
PC	45.500	18	0.000	33.404	9	0.000
Ad	21.847	21	0.408	9.750	12	0.638
Pd	200.976	29	0.000	188.879	20	0.000
Cd	47.171	19	0.000	35.075	10	0.000
A	45.822	22	0.002	33.725	13	0.001
P	5036.080	30	0.000	5023.983	21	0.000
C	516.272	20	0.000	504.175	11	0.000
t	201.864	30	0.000	189.767	21	0.000
tA	223.070	31	0.000	210.973	22	0.000
tP	5036.598	31	0.000	5024.502	22	0.000
tC	806.013	31	0.000	793.916	22	0.000
1	5081.113	32	0.000	5069.017	23	0.000

	aic
APC	264.065
AP	250.844
AC	263.309
PC	279.468
Ad	249.815
Pd	412.944
Cd	279.139
A	271.790
P	5246.048
C	746.240
t	411.832
tA	431.038
tP	5244.567
tC	1013.981
1	5287.081

Effect of ad hoc identification

At first a subset is chosen where youngest age and cohort groups are truncated. This way sparsity is eliminated and ad hoc identification effects are dominated by estimation uncertainty. Then consider Plot 1: parameters estimated from data without first age groups Plot 2: parameters estimated from all data Note that estimates for double difference very similar. Estimates for linear slopes are changed because the indices used for parametrising these are changed Estimates for detrended double sums of age and cohort double differences are changed, because they rely on a particular ad hoc identifications that have changed. Nonetheless these plots are useful to evaulate variation in time trends over and above linear trends.

```
> graphics.off()
> data.list     <- data.Belgian.lung.cancer()
> data.list.subset <- apc.data.list.subset(data.list,2,0,0,0,0,0)

WARNING apc.data.list.subset: cuts in arguments are:
[1] 2 0 0 0 0 0
have been modified to:
[1] 2 0 0 0 0 2
WARNING apc.data.list.subset: coordinates changed to "AC" & data.format changed to "t

> fit.apc      <- apc.fit.model(data.list,"poisson.dose.response","APC")
> fit.apc.subset  <- apc.fit.model(data.list.subset,"poisson.dose.response","APC")
> apc.plot.fit(fit.apc.subset,
+              main.outer="1. Belgian lung cancer: cut first two age groups")

WARNING apc.plot.fit: sdv large for plot 5 - possibly not plotted

> apc.plot.fit(fit.apc,main.outer="2. Belgian lung cancer data: all data")

WARNING apc.plot.fit: sdv large for plot 5 - possibly not plotted
```

3   References

Clayton, D. and Schifflers, E. (1987a) Models for temperoral variation in cancer rates. I: age-period and age-cohort models. *Statistics in Medicine* 6, 449-467.

Kuang, D., Nielsen, B. and Nielsen, J.P. (2008a) Identification of the age-period-cohort model and the extended chain ladder model. *Biometrika* 95, 979-986. *Download*: Earlier version: http://www.nuffield.ox.ac.uk/economics/papers/2007/w5/ KuangNielsenNielsen07.pdf.

Kuang, D., Nielsen, B. and Nielsen, J.P. (2008b) Forecasting with the age-period-cohort model and the extended chain-ladder model. *Biometrika* 95, 987-991. *Download*: Earlier version: http://www.nuffield.ox.ac.uk/ economics/papers/2008/w9/KuangNielsenNielsen_Forecast.pdf.

Nielsen, B. (2014) Deviance analysis of age-period-cohort models. *Download*: http://www.nuffield.ox.ac.uk/economics/papers/2014/apc_deviance.pdf.

Nielsen, B. (2015) apc: An R package for age-period-cohort analysis. *R Journal* 7, 52-64. *Download*: https://journal.r-project.org/archive/2015-2/nielsen.pdf.