Identification in `apc` package

12 April 2015

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

Contents

1   Introduction

The `apc` package works with a number of parametrisations. The estimation is done in terms of the canonical parameter defined in Kuang, Nielsen and Nielsen (2008). This parametrisation is invariant to the group of transformations characterising the identification problem.

  The plots of the fit do, however, allow the display of a two ad hoc identifications of the time effects. These time effects are not invariant, but may nonetheless be helpful in applications. The purpose with the vignette is to demonstrate that these parametrisations are equivalent. This is done by showing that they give the same fit.

  The two ad hoc identifications are as follows.

  The first, "sum.sum", displays the components that appear in the representation theorem in Nielsen (2014), which is a development of the representation in Kuang, Nielsen and Nielsen (2008).

  The second, "detrend" is possibly more useful in practice. This is the default option in `apc.plot.fit`. It displays detrended versions of the double sums, so that they start and end in zero. Thus, it displays the cumulated deviations from linearity arising from the double difference parameters in canonical parameter. Moreover, the three time effects are treated symmetrically so that the identification of one time trend has no bearing on the other time trends. This is based on the development in Nielsen (2014).

  The two ad hoc identifications have the feature that the number of non-zero elements is the same as the number of elements in the canonical parameter.

  We use the Belgian lung cancer data from Clayton and Schifflers (1987a) as an example.

2   Ad hoc identification of the time effects

The age-period-cohort model is given by

$$\mu_{ik} = \alpha_i + \beta_j + \gamma_k + \delta, \tag{1}$$

for age i, cohort k and period $j = i + k - 1$. The time effects α_i, β_j, γ_k are not fully identified.

  Canonical parameter. The `apc` package works around the canonical parameter from Kuang, Nielsen, Nielsen (2008)

$$\xi = (\mu_{UU}, \mu_{U+1,U}, \mu_{U,U+1}, \dots, \Delta^2\alpha_i, \dots, \dots, \Delta^2\beta_j, \dots, \dots, \Delta^2\gamma_k, \dots)'. \tag{2}$$

The design matrix is defined from the following representation taken from Nielsen (2014)

$$\mu_{ik} = \nu_0 + (i - U)\nu_a + (k - U)\nu_c + A_i + B_j + C_k \tag{3}$$

where $j = i + k - 1$ and $U = \text{integer}\{(L+3)/2\}$ while

$$A_i = 1_{(i<U)} \sum_{t=i+2}^{U+1} \sum_{s=t}^{U+1} \Delta^2 \alpha_s + 1_{(i>U+1)} \sum_{t=U+2}^{i} \sum_{s=U+2}^{t} \Delta^2 \alpha_s \tag{4}$$

$$B_j = 1_{(L \text{ odd } \& j=2U-2)} \Delta^2 \beta_{2U} + 1_{(j>2U)} \sum_{t=2U+1}^{j} \sum_{s=2U+1}^{t} \Delta^2 \beta_s \tag{5}$$

$$C_k = 1_{(k<U)} \sum_{t=k+2}^{U+1} \sum_{s=t}^{U+1} \Delta^2 \gamma_s + 1_{(k>U+1)} \sum_{t=U+2}^{k} \sum_{s=U+2}^{t} \Delta^2 \gamma_s \tag{6}$$

Six of these coefficients are zero. That is $A_U = A_{U+1} = C_U = C_{U+1} = 0$, if L is odd then $B_{2U+1} = B_{2U+2} = 0$, whereas if L is even then $B_{2U} = B_{2U+1} = 0$. We get the canonical parameter through the command

```
> # attach apc library
> library(apc)
> # get data from precoded function
> data <- data.Belgian.lung.cancer()
> # Estimate APC model
> model.family <- "poisson.dose.response"
> model.design <- "APC"
> fit <- apc.fit.model(data,model.family,model.design)
> c.c <- fit$coefficients.canonical
```

ss.dd or **sum.sum** parametrisation. We can also parametrise in terms of the coefficients A_i, B_j, C_k. These coefficients are ad hoc identified versions of the time effects α_i, β_j, γ_k. We get these parameters through the command `apc.identify`. Note, that this command is called indirectly by `apc.plot.fit`, so there is no need to use this command when working with plots only. The command is

```
> id <- apc.identify(fit)
> c.ssdd <- id$coefficients.ssdd
```

detrend parametrisation. In practise it is easier to see deviations from linearity through a different ad hoc identification where the time effects are constrained to start and end in zero; see Nielsen (2014b) for discussion of the interpretation. From A_i, B_j and C_k defined in (4)-(6) we form

$$A_i^{detrend} = A_i - A_1 - \frac{i-1}{I-1}(A_I - A_1), \tag{7}$$

$$B_{L+j}^{detrend} = B_{L+j} - B_{L+1} - \frac{j-1}{J-1}(B_{L+J} - B_{L+1}), \tag{8}$$

$$C_k^{detrend} = C_k - C_1 - \frac{k-1}{K-1}(C_K - C_1). \tag{9}$$

The constants and linear slopes are corrected correspondingly giving

$$\nu_0^{detrend} \quad = \quad \nu_0 + A_1 + B_{L+1} - \frac{L}{J+1}(B_{L+J} - B_{L+1}) + C_1, \tag{10}$$

$$\nu_a^{detrend} \quad = \quad \nu_a + \frac{A_I - A_1}{I-1} + \frac{B_{L+J} - B_{L+1}}{J-1} \tag{11}$$

$$\nu_c^{detrend} \quad = \quad \nu_c + \frac{B_{L+J} - B_{L+1}}{J-1} + \frac{C_K - C_1}{K-1} \tag{12}$$

We then get the representation

$$\mu_{ik} \quad = \quad \nu_0^{detrend} + (i-1)\nu_a^{detrend} + (k-1)\nu_c^{detrend} + A_i^{detrend} + B_{L+j}^{detrend} + C_k^{detrend} \tag{13}$$

These coefficients are also found using the identification command `apc.identify`, so that

```
> c.detrend <- id$coefficients.detrend
```

3 Ad hoc identification for 2-factor models

When one of the three time effects, α_i, β_j or γ_k is absent the linear trends can be attributed uniquely to the time effects, whereas the level is still unidentified. The `apc` package offers a parametrisation of that type.

demean parametrisation. The model design "AC" is a two factor model. The representation derives from the detrended representation (13). With $\Delta^2\beta_j = 0$ then $B_j = 0$ and hence $B_j^{detrend} = 0$ in that formula. Thus we can attribute the linear trends to the age and cohort time effect. This is done so that the first element of each time effect is zero. This identification choice is called the demeaned parameter.

$$\mu_{ik} \quad = \quad \nu_0^{demean} + A_i^{demean} + C_k^{demean} \tag{14}$$

where

$$A_i^{demean} \quad = \quad A_i^{detrend} + (i-1)\nu_a^{detrend} \qquad C_k^{demean} = C_k^{detrend} + (k-1)\nu_c^{detrend} \tag{15}$$

Note that $A_1^{demean} = C_1^{demean} = 0$. We get the single sum of difference parameters through the following command

```
> # fit AC model
> model.design <- "AC"
> fit.ac <- apc.fit.model(data,model.family,model.design)
> # identify to get sums of difference parameters
> id.ac <- apc.identify(fit.ac)
> c.demean <- id.ac$coefficients.demean
```

We get an exactly, invariant identified parameter by taking first differences of $A_i^{demean}, C_k^{demean}$. This could also have been used as canonical parameter for this two factor model. The level is as before.

```
> # get difference parameters
> c.dif <- id.ac$coefficients.dif
```

4   Check canonical parametrisation

The fit of the canonical parametrisation is discussed.

In the background the generalized linear model package is used. `apc` supplies a design matrix that is fed into `glm.fit`. This delivers the following standard output including coefficients and linear predictors.

```
> # Coefficients
> fit$coeefficients

 [1]  1.957545765  0.504384208  0.120878598 -0.497116645  0.253907149
 [6] -0.155115204 -0.205504949 -0.043344184 -0.092603145  0.023605714
[11] -0.046471233 -0.077331927 -0.065187056  0.064058271  0.089055820
[16]  0.022794960 -0.009887705 -0.087602585  0.070174649  0.005654086
[21]  0.015051099 -0.093529779  0.191506367 -0.214529781  0.160455202
[26] -0.609263039


> # Arrange linear predictors as matrix in original format
> # Create matrix of original dimension
> m.fit <- fit$response
> m.fit[fit$index.data] <-fit$linear.predictors
> m.fit

       1955-1959 1960-1964 1965-1969 1970-1974
25-29   1.378189  1.377244  1.403232  1.098612
30-34   2.344254  2.563195  2.456646  2.559679
35-39   2.783587  2.998695  3.152449  3.107739
40-44   3.505436  3.679197  3.837896  4.019294
45-49   4.297587  4.238527  4.357794  4.575612
50-54   4.733915  4.813556  4.700184  4.876142
55-59   5.129520  5.191959  5.219343  5.172929
60-64   5.255470  5.474294  5.486497  5.577828
65-69   5.339078  5.589386  5.755991  5.834897
70-74   5.375296  5.560837  5.757173  5.992722
75-79   5.288267  5.398145  5.535912  5.800197
```

The coefficients are equal to the canonical parameters. We get the summary of these and check that the coefficients are the same as for the generalized linear model estimation.

```
> # Canonical paramters
> c.c

                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

> # Check canonical coefficients are the same as the standard coefficients
> sum(abs(c.c[,1]-fit$coefficients))

[1] 0
```

We check that the canonical parameters give the same fit as the standard generalized linear model fit. We get the design matrix and multiply with the canonical parameter

```
> # get design matrix
> m.design <- apc.get.design(fit)$design
> # create matrix of original dimension
> m.fit.canonical.no.dose <- fit$response
> m.fit.canonical.no.dose[fit$index.data] <- m.design %*% c.c[,1]
> if(is.null(data$dose)==TRUE)
+  m.fit.canonical <- m.fit.canonical.no.dose
> if(is.null(data$dose)==FALSE)
+  m.fit.canonical <- m.fit.canonical.no.dose + log(data$dose)
> m.fit.canonical

      1955-1959 1960-1964 1965-1969 1970-1974
25-29  1.378189  1.377244  1.403232  1.098612
30-34  2.344254  2.563195  2.456646  2.559679
```

```
35-39   2.783587   2.998695   3.152449   3.107739
40-44   3.505436   3.679197   3.837896   4.019294
45-49   4.297587   4.238527   4.357794   4.575612
50-54   4.733915   4.813556   4.700184   4.876142
55-59   5.129520   5.191959   5.219343   5.172929
60-64   5.255470   5.474294   5.486497   5.577828
65-69   5.339078   5.589386   5.755991   5.834897
70-74   5.375296   5.560837   5.757173   5.992722
75-79   5.288267   5.398145   5.535912   5.800197

> # Check canonical coefficients give same fit as standard fit
> sum(abs(m.fit-m.fit.canonical),na.rm=TRUE)

[1] 0
```

5 Check identification of double sums of double differences

We check the ad hoc identification in terms of the double sums of double differences (4)-(6).

We first get the cooefficients through

```
> c.ssdd
```

	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
SS_DD_age_25	-1.493388676	0.48248663	-3.09519183	1.966856e-03
SS_DD_age_30	-0.846214843	0.34857045	-2.42767231	1.519607e-02
SS_DD_age_35	-0.696157655	0.26337966	-2.64317170	8.213337e-03
SS_DD_age_40	-0.292193318	0.18606144	-1.57041306	1.163190e-01
SS_DD_age_45	-0.043344184	0.11872510	-0.36508022	7.150515e-01
SS_DD_age_50	0.000000000	NA	NA	NA
SS_DD_age_55	0.000000000	NA	NA	NA
SS_DD_age_60	-0.092603145	0.09711954	-0.95349653	3.403386e-01
SS_DD_age_65	-0.161600575	0.15520771	-1.04118911	2.977878e-01
SS_DD_age_70	-0.277069239	0.21680571	-1.27796101	2.012632e-01
SS_DD_age_75	-0.469869830	0.28164144	-1.66832631	9.525097e-02
SS_DD_period_1955	0.000000000	NA	NA	NA
SS_DD_period_1960	0.000000000	NA	NA	NA
SS_DD_period_1965	-0.065187056	0.06656344	-0.97932225	3.274208e-01
SS_DD_period_1970	-0.066315840	0.10404630	-0.63736857	5.238848e-01
SS_DD_cohort_1880	0.105445533	0.30899731	0.34125066	7.329149e-01
SS_DD_cohort_1885	0.020910393	0.24866705	0.08408992	9.329849e-01

```
SS_DD_cohort_1890   0.025431073 0.19073810   0.13332980  8.939326e-01
SS_DD_cohort_1895   0.052746714 0.13709278   0.38475194  7.004212e-01
SS_DD_cohort_1900   0.070174649 0.08627345   0.81339796  4.159899e-01
SS_DD_cohort_1905   0.000000000         NA           NA            NA
SS_DD_cohort_1910   0.000000000         NA           NA            NA
SS_DD_cohort_1915   0.005654086 0.10239118   0.05522044  9.559628e-01
SS_DD_cohort_1920   0.026359272 0.15579102   0.16919635  8.656422e-01
SS_DD_cohort_1925  -0.046465322 0.21409492  -0.21703140  8.281839e-01
SS_DD_cohort_1930   0.072216451 0.28513167   0.25327404  8.000565e-01
SS_DD_cohort_1935  -0.023631557 0.37047481  -0.06378721  9.491397e-01
SS_DD_cohort_1940   0.040975638 0.47410340   0.08642764  9.311265e-01
SS_DD_cohort_1945  -0.503680205 0.78473175  -0.64185016  5.209705e-01
```

Compare the output for `c.ssdd` with that for `c.c` above. `c.ssdd` has more elements as some zero elements are squeezed in according to the indicator functions in (3). The coefficients and standard errors neighbouring the zero elements can be recognised directly from `c.ssdd`. The number of non-zero elements is the same.

The prediction based on these parameters is found as follows. An age-cohort indexation is needed for the data. This is generated by the command `apc.get.index`. It is automatically linked as an object for fit. We can therefore generate the linear prediction as follows and compare with the previous predictions. Note, this first check requires that model design is APC.

```
> age <- fit$index.trap[,1]
> coh <- fit$index.trap[,2]
> # From this we get the period. Need to correct for lowest period value.
> per.zero <- fit$per.zero
> per <- age+coh-1-per.zero
> U <- fit$U
> # Then we can compute the prediction as a vector
> if(model.design=="APC")
+ {
+  prediction <- c.ssdd[1,1] +
+    + c.ssdd[2,1]*(age-U) +
+    + c.ssdd[3,1]*(coh-U) +
+    + c.ssdd[id$index.age.max[age],1] +
+    + c.ssdd[id$index.per.max[per],1] +
+    + c.ssdd[id$index.coh.max[coh],1]
+  # Then we embed it into a matrix
+  m.fit.ssdd <- fit$response
+  m.fit.ssdd[fit$index.data] <- prediction
+  # Add dose
+  m.fit.ssdd  <- m.fit.ssdd + log(data$dose)
+  # Check fit is correct
+  sum(abs(m.fit.canonical-m.fit.ssdd),na.rm=TRUE)
+ }
```

The above code will fail if `model.design` is not APC. We then need to introduce various checks as in the following snippet, that recycles `age`, `per`, `coh`, `per.zero`, `U` defined above.

```
> # We need two further variables
> slopes <- fit$slopes
> difdif <- fit$difdif
> # Compute the prediction as a vector
> prediction <- c.ssdd[1,1]
> # Add the age double differences and age slope
> if(difdif[1]) # TRUE if age double differences
+  prediction <- prediction + c.ssdd[id$index.age.max[age],1]
> if(difdif[2]) # TRUE if period double differences
+  prediction <- prediction + c.ssdd[id$index.per.max[per],1]
> if(difdif[3]) # TRUE if cohort double differences
+  prediction <- prediction + c.ssdd[id$index.coh.max[coh],1]
> if(slopes[1]) # TRUE if age linear trend
+ { prediction <- prediction + c.ssdd[2,1]*(age-U)
+   if(slopes[3])
+   prediction <- prediction + c.ssdd[3,1]*(coh-U)
+ }
> if(slopes[1]==FALSE)
+ { if(slopes[2])
+    prediction <- prediction + c.ssdd[2,1]*(per-1)
+   if(slopes[3])
+    prediction <- prediction + c.ssdd[2,1]*(coh-U)
+ }
> # Then we embed it into a matrix
> m.fit.ssdd <- fit$response
> m.fit.ssdd[fit$index.data] <- prediction
> # Add dose
> if(is.null(data$dose)==FALSE)
+  m.fit.ssdd  <- m.fit.ssdd + log(data$dose)
> # Check fit is correct
> sum(abs(m.fit.canonical-m.fit.ssdd),na.rm=TRUE)

[1] 4.440892e-15
```

We can plot these coefficients using `apc.plot.fit` and the reconcile with the prediction as follows.

```
> apc.plot.fit(fit,type="sum.sum")

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

> m.fit.canonical.no.dose
```

```
         1955-1959   1960-1964   1965-1969    1970-1974
25-29 -1.3811545  -1.3561239  -1.2358252  -1.66073121
30-34 -0.4691568  -0.2295965  -0.2697529  -0.08539592
35-39  0.1372306   0.1852846   0.3596579   0.38355968
40-44  0.9039953   1.0455791   1.0284460   1.26687764
45-49  1.5306960   1.6572287   1.7336254   1.78055061
50-54  1.9575458   2.0784244   2.1397700   2.28022499
55-59  2.4112260   2.4619300   2.5176215   2.64302542
60-64  2.6847006   2.8230071   2.8085240   2.92827379
65-69  2.9718931   3.1200873   3.1932068   3.24278197
70-74  3.2354094   3.3608086   3.4438158   3.58099357
75-79  3.5106495   3.5469930   3.6072052   3.75427065
```

The anchoring point is for $age = cohort = U = 6$ and $period = L + 1$, or in real coordinates $age = "50 - 54"$ and $period = "1955 - 1959"$ corresponding to cohort 1905. The predictor is 1.9574. This is the sum of the entries in panels (d)-(i) of the plot, which are all zero apart from the constant.

6 Check identification of detrended double sums of double differences

The double differences capture variation in the time trends over and above linear trends. Thus, when considering double sums of double differences only variation over and above linear trends should be interpreted. To emphasize this variation, Nielsen (2014) suggests to detrend the double sums so that they start and end in zero as outlined in (**??**). Some interpretations of those detrended double sums are also offered in that paper.

The prediction based on these parameters is found as follows.

```
> # We use age, coh, per, per.zero, id, slopes, difdif defined above.
> # Compute the prediction as a vector
> prediction <- c.detrend[1,1]
> # Add the age double differences and age slope
> if(difdif[1]) # TRUE if age double differences
+  prediction <- prediction + c.detrend[id$index.age.max[age],1]
> if(difdif[2]) # TRUE if period double differences
+  prediction <- prediction + c.detrend[id$index.per.max[per],1]
> if(difdif[3]) # TRUE if cohort double differences
+  prediction <- prediction + c.detrend[id$index.coh.max[coh],1]
> if(slopes[1]) # TRUE if age linear trend
+ { prediction <- prediction + c.detrend[2,1]*(age-1)
+   if(slopes[3])
+   prediction <- prediction + c.detrend[3,1]*(coh-1)
+ }
> if(slopes[1]==FALSE)
+ { if(slopes[2])
```

```
+     prediction <- prediction + c.detrend[2,1]*(per-1)
+   if(slopes[3])
+     prediction <- prediction + c.detrend[2,1]*(coh-1)
+ }
> # Then we embed it into a matrix
> m.fit.detrend <- fit$response
> m.fit.detrend[fit$index.data] <- prediction
> # Add dose
> if(is.null(data$dose)==FALSE)
+  m.fit.detrend  <- m.fit.detrend + log(data$dose)
> # Check fit is correct
> sum(abs(m.fit.canonical-m.fit.detrend),na.rm=TRUE)


[1] 9.103829e-15
```

We can plot these coefficients using `apc.plot.fit` and the reconcile with the prediction as follows.

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


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


> m.fit.canonical.no.dose


        1955-1959   1960-1964   1965-1969    1970-1974
25-29 -1.3811545 -1.3561239 -1.2358252 -1.66073121
30-34 -0.4691568 -0.2295965 -0.2697529 -0.08539592
35-39  0.1372306  0.1852846  0.3596579  0.38355968
40-44  0.9039953  1.0455791  1.0284460  1.26687764
45-49  1.5306960  1.6572287  1.7336254  1.78055061
50-54  1.9575458  2.0784244  2.1397700  2.28022499
55-59  2.4112260  2.4619300  2.5176215  2.64302542
60-64  2.6847006  2.8230071  2.8085240  2.92827379
65-69  2.9718931  3.1200873  3.1932068  3.24278197
70-74  3.2354094  3.3608086  3.4438158  3.58099357
75-79  3.5106495  3.5469930  3.6072052  3.75427065
```

We consider the top left point of the prediction matrix with real coordinates $age = "25 - 29"$ and $period = "1970 - 74"$ corresponding to cohort 1945. In mathematical coordinates, that is $age = 1$, $cohort = coh.max$, $period = per.max$. Thus all double difference sums in panels (g)-(i) and age slope in panel (d) is zero. The level in panel (e) is -2.34. The cohort slope value in (f) is $0.68 = (14 - 1) * 0.052$. In total this is -1.66.

7 Check identification for model design "AC"

We now check the identification for the two factor model. To check the parametrisation we first recalculate the fit using the canonical parameter. Then the alternative parametrisations are checked

```
> if(model.design=="AC")
+ {
+   ###################################
+   # Get fit of canonical parameters
+   # get the canonical parameters
+   c.c.ac <- fit.ac$coefficients.canonical
+   # Get design matrix
+   m.design.ac <- apc.get.design(fit.ac)$design
+   # Create matrix of original dimension
+   m.fit.canonical.ac <- fit.ac$response
+   m.fit.canonical.ac[fit.ac$index.data] <- m.design.ac %*% c.c.ac[,1]
+   ####################################
+   # Get fit of sum of difference parameters
+   prediction <- c.demean[1,1] +
+     + c.demean[id.ac$index.age.sub[age],1] +
+     + c.demean[id.ac$index.coh.sub[coh],1]
+   # Create matrix of original dimension
+   m.fit.demean.ac <- fit.ac$response
+   m.fit.demean.ac[fit.ac$index.data] <- prediction
+ }
```

8 Summary of all checks

```
> # ALL CHECKS
> sum(abs(c.c[,1]-fit$coefficients))

[1] 0

> sum(abs(m.fit-m.fit.canonical),na.rm=TRUE)

[1] 0

> sum(abs(m.fit.canonical-m.fit.ssdd),na.rm=TRUE)

[1] 4.440892e-15

> sum(abs(m.fit.canonical-m.fit.detrend),na.rm=TRUE)

[1] 9.103829e-15

> sum(abs(m.fit.canonical.ac-m.fit.demean.ac),na.rm=TRUE)

[1] 2.803313e-14
```

References

Clayton, D., 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. (2008) 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.

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