Reproducing Nielsen (2022) using the apc package

3 July 2025

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 to use the `apc` package version 3.0.0 to reproduce some the result in Nielsen (2022): *Two-sample age-period-cohort models with an application to Swiss suicide rates.*. The `apc` package builds on the identification analysis and the forecast theory in Kuang, Nielsen and Nielsen (2008a,b), the development of deviance analysis for general data arrays in Nielsen (2014). The package is discussed in Nielsen (2015). Fannon and Nielsen (2019) provide a review.

The data originates from Riebler, Held, Rue and Bopp (2012). The data consists of counts of Swiss suicides and population by sex, in 5 year age groups from 15 to 79, by period from 1950 to 2007. It also includes time series such as a family index, the F-index. The purpose is to analyze if the age-period-cohort structure is common for the sexes and if the period effect can be replaced by a time series. This is modelled using a dose-response normal model with a two-sample age-period-cohort structure.

The data are available in the `apc` package. They can be called with the command

```
> # Call library and data. Give short names to data objects.
> library(apc)
> data <- data.Swiss.suicides()
> data.list.f<-data$data.list.f
> data.list.m<-data$data.list.m
> F_index<-data$v.F_index
```

Here `data.Swiss.suicides()` is a function that returns a list includes two `apc.data.list`s for females and males as well as five `vector`s of time series. Each `apc.data.list` is a list in itself including matrices for dose and response in `APm` format. This is a mixed scale format with $G = 5$ year age groups $15-19$, $20-24$, ..., $75-79$ and $H = 1$ year periods 1950, 1951, ..., 2007. To see the structure of the function use the code

```
> # Show some of the data
> objects(data)
```

```
[1] "data.list.f"       "data.list.m"       "v.F_index"
[4] "v.cumMarriage.f"   "v.cumMarriage.m"   "v.divorcePer1000"
[7] "v.marriagePer1000"
```

```
> (data.list.f)$response[1:5,1:5]
```

```
      1950 1951 1952 1953 1954
15-19   15    9    9   14    8
20-24   23   19   17   14   23
25-29   26   24   16   23   20
30-34   16   16   23   12   22
35-39   21   25   27   15   18
```

2 Figure 1: Crude rates per 100,000

The first figure shows crude suicide rates by age and period for women and men. The following command show plots separately for women and for men.

```
> # Various plots of data sums
> apc.plot.data.sums(data.list.f,"c",scale.rate=100000)
```

The figure in the paper has the crude rates for women and men in the same plot. First, a code for making two plots in one.

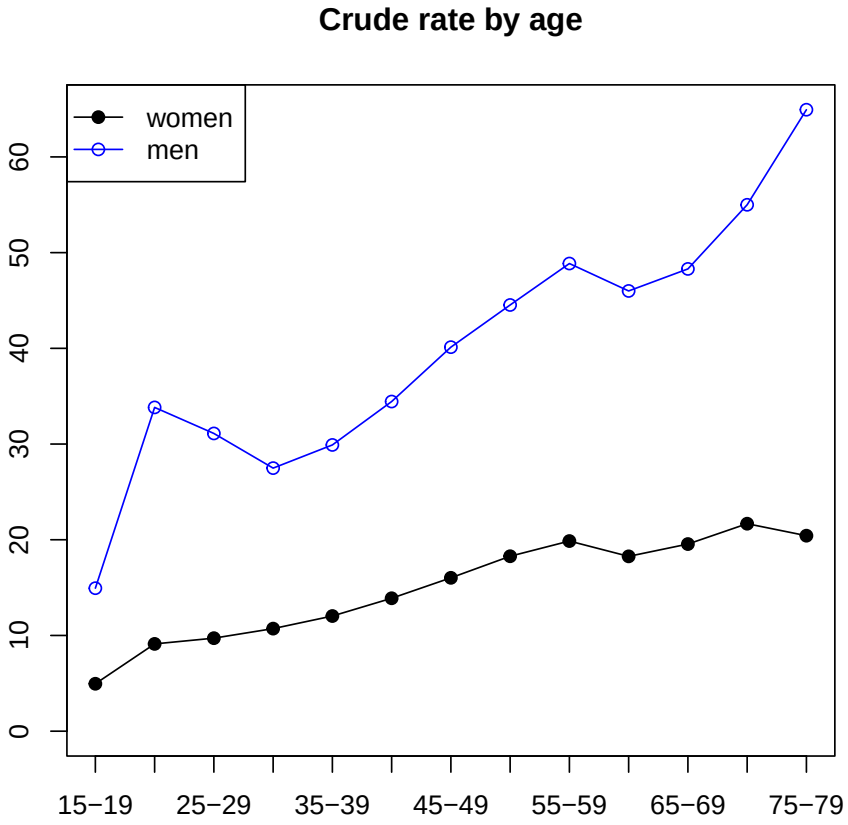
```
> # Function for plotting two variables in one plot
> plot.both<- function(y.f,y.m,main,position){
+ x<-1:length(y.f)
+ labels<-names(y.f)
+ xlim<-c(1,length(x))
+ ylim<-c(0,max(y.f,y.m))
+ plot(NULL,xaxt="n",xlab="",ylab="",xlim=xlim,ylim=ylim,main=main)
+ axis(side=1,at=x,labels)
+ legend(position,col=c("black","blue"),lty=c(1,1),pch=c(19,1),legend=c("women","men"
+  lines(x,y.f,col="black")
+ points(x,y.f,col="black",pch=19)
+  lines(x,y.m,col="blue" )
+ points(x,y.m,col="blue" ,pch=1 )
+ }
```

Then, we compute the crude rates and scale with 100000.

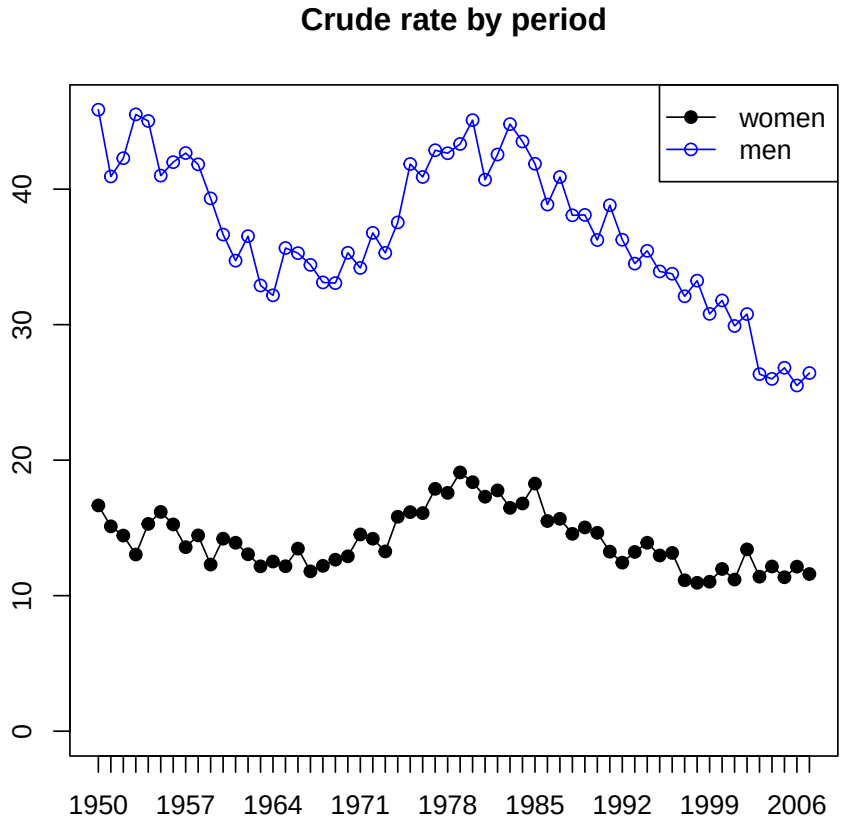
```
> # Compute crude rates
> crude.age.f<-100000*rowSums(data.list.f$response)/rowSums(data.list.f$dose)
> crude.age.m<-100000*rowSums(data.list.m$response)/rowSums(data.list.m$dose)
> crude.per.f<-100000*colSums(data.list.f$response)/colSums(data.list.f$dose)
> crude.per.m<-100000*colSums(data.list.m$response)/colSums(data.list.m$dose)
```

And finally generate the plots

```
> # Plot of data sums by age
> plot.both(crude.age.f,crude.age.m,"Crude rate by age","topleft")
```




```
> # Plot of data sums by period
> plot.both(crude.per.f,crude.per.m,"Crude rate by period","topright")
```



3 Table 2: Analysis of variance, women

```
> # Analysis of variance, women
> apc.table.f<-apc.fit.table(data$data.list.f,"log.normal.rates")
> Table2<-apc.table.f[1:4,c(1,2,6,4,7)]
> # Add scale estimates
> sigma.f<-sqrt(c(apc.fit.model(data$data.list.f,"log.normal.rates","APC")$s2,
+ apc.fit.model(data$data.list.f,"log.normal.rates","AP" )$s2,
+ apc.fit.model(data$data.list.f,"log.normal.rates","AC" )$s2,
+ apc.fit.model(data$data.list.f,"log.normal.rates","PC" )$s2))
> sigma<-sigma.f
> Table2<-cbind(Table2,sigma)
> # Increase precision on p.values
> Table2[,5]<-pf(as.numeric(Table2[,3]),as.numeric(Table2[,4]),as.numeric(Table2[,2])
> print("Table 2: Analysis of variance, women")

[1] "Table 2: Analysis of variance, women"

> Table2
```

```
      -2logL df.residual F vs.APC df vs.APC     prob(>F)     sigma
APC -363.016         572      NaN       NaN          NaN 0.2183756
AP  -157.383         684    1.601       112 2.489398e-04 0.2288733
AC   -68.819         624    5.250        52 3.990311e-25 0.2541188
PC   -18.546         583   30.113        11 2.890638e-50 0.2718147
```

```
> # Normality test, women
> fit.apc.f        <- apc.fit.model(data$data.list.f,"log.normal.rates","APC")
> print("Normality test, women")

[1] "Normality test, women"

> apc.test.normal.residuals(fit.apc.f,remove.zeros=TRUE)[c(1,6),]

          cumulant test stat  df   p-value
obs              NA        NA 744        NA
skew+kurt        NA  4.498492   2 0.1054787
```

4 Table 3: Analysis of variance, men

The analysis of variance for men. The first command gives the basic table. The few lines add the estimated residual variances.

```
> # Analysis of variance, men
> apc.table.m<-apc.fit.table(data$data.list.m,"log.normal.rates")
> Table2<-apc.table.m[1:4,c(1,2,6,4,7)]
> # Add scale estimates
> sigma.m<-sqrt(c(apc.fit.model(data$data.list.m,"log.normal.rates","APC")$s2,
+ apc.fit.model(data$data.list.m,"log.normal.rates","AP" )$s2,
+ apc.fit.model(data$data.list.m,"log.normal.rates","AC" )$s2,
+ apc.fit.model(data$data.list.m,"log.normal.rates","PC" )$s2))
> sigma<-sigma.m
> Table2<-cbind(Table2,sigma)
> # Increase precision on p.values
> Table2[,5]<-pf(as.numeric(Table2[,3]),as.numeric(Table2[,4]),as.numeric(Table2[,2]),
> print("Table 2: Analysis of variance, men")

[1] "Table 2: Analysis of variance, men"

> Table2

      -2logL df.residual F vs.APC df vs.APC     prob(>F)     sigma
APC -841.704         572      NaN       NaN          NaN 0.1589809
AP  -605.475         684    1.879       112 1.072019e-06 0.1700386
AC  -434.270         624    7.883        52 1.560073e-41 0.1994294
PC  -210.521         583   68.104        11 4.532589e-97 0.2393236
```

```
> # Normality test, men
> fit.apc.m        <- apc.fit.model(data$data.list.m,"log.normal.rates","APC")
> print("Normality test, men")

[1] "Normality test, men"

> apc.test.normal.residuals(fit.apc.m,remove.zeros=TRUE)[c(1,6),]

          cumulant test stat  df  p-value
obs             NA        NA 744       NA
skew+kurt       NA 0.3505077   2 0.839244
```

5 Table 4: Analysis of variance, both

The analysis of variance for both samples. First, it is done with OLS, that is imposing a common variance for the two samples. Second, it is done with GLS, that allows for different variance for the two samples. This requires that one specifies the relative variance. For this, the variances are computed from two one-sample analyses. Then the tables are put together.

```
> # Analysis of variance, both
> table.ols<-apc.fit.table.2s(data.list.f,data.list.m,"log.normal.rates","difference"
> s.f        <- sqrt(apc.fit.model(data.list.f,"log.normal.rates","APC")$s2)
> s.m        <- sqrt(apc.fit.model(data.list.m,"log.normal.rates","APC")$s2)
> table.gls        <- apc.fit.table.2s(data.list.f,data.list.m,"gls.log.normal.rates"
> Table4 <- cbind(table.ols[1:4,c(1,2,4,6,7,10)],table.gls[1:4,10])
> colnames(Table4)[c(1,6:7)] <- c("-2logL_OLS","sigma_OLS","sigma_GLS")
> Table4
```

Normality tests are computed for the two samples.

```
> # Normality tests, both
> fit.apc.OLS        <- apc.fit.model.2s(data.list.f,data.list.m,        "log.normal.
> fit.apc.GLS        <- apc.fit.model.2s(data.list.f,data.list.m,"gls.log.normal.rate
> print("Normality test, two-sample, OLS")
> apc.test.normal.residuals(fit.apc.OLS,remove.zeros=TRUE)[c(1,6),]
> print("Normality test, two-sample, GLS")
> apc.test.normal.residuals(fit.apc.GLS,remove.zeros=TRUE)[c(1,6),]
```

6 Figure 2: Detrended macro effects

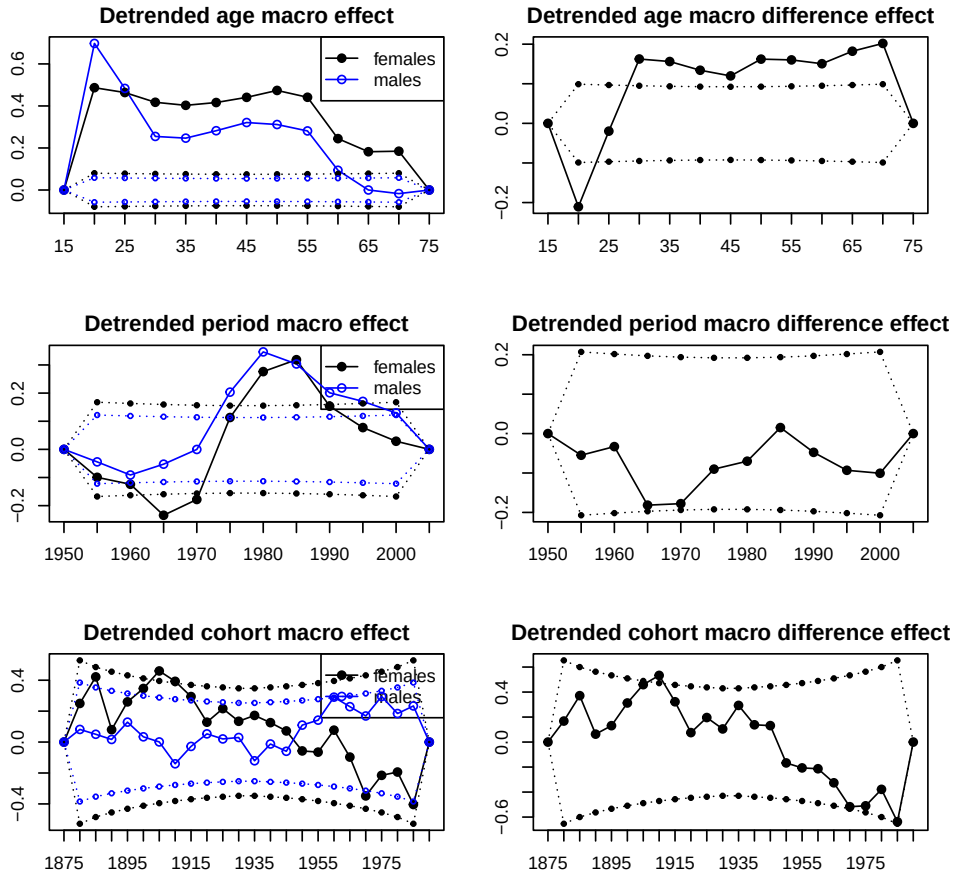
This figure is generated from the two one sample fits.

First, we plot the macro effects. At first, the effects from both samples are plotted together. There is one plot for each of age, period and cohort. Next, the cross-sample difference is plotted. Again, there are separate plots for each of age, period and cohort. If the option which.plot is omitted or set to 0 all six plots are made in one go. Otherwise, individual plots can be picked by choosing a value in the range 1-6 for which.plot.

Model			Test vs APC				
diff.	$-2\log L_{GLS}$	df	F	df	p_F	$\hat{\sigma}_{OLS}$	$\hat{\sigma}_{GLS}$
APC	-1129.99	1144				0.191	0.159
AP	-836.72	1256	2.19	112	0.0000	0.201	0.167
AC	-1052.62	1196	1.16	52	0.2092	0.192	0.160
PC	-1021.68	1155	7.74	11	0.0000	0.197	0.164
OLS: $\chi^2_{normality}(2) = 15.82$ $[p = 0.0004]$							
GLS: $\chi^2_{normality}(2) = 2.13$ $[p = 0.3440]$							

Table 1: Two-sample analysis of variance. Submodels refer to restrictions on cross-sample differenced predictor, while the common predictor is an unrestricted APC model. For GLS, data for women are scaled to have same dispersion as men.
The structure of the table is close to that of Table **??**. Column 7 shows the scale estimated by OLS, so that the scale is common across samples. Column 8 shows the scale estimated by GLS, allowing different scales for the samples.

```
> # Plot 1-sample fits jointly. Macro effects.
> par(mfrow=c(3,2),oma=c(0,0,2,0),mar=c(4,2,2,2)+0.1)
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="macro",which.plot=1)
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="macro",which.plot=4)
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="macro",which.plot=2)
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="macro",which.plot=5)
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="macro",which.plot=3)
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="macro",which.plot=6)
```



Here the macro effects are identified by setting first and last value to zero. This preserves the degrees of freedom from the underlying double differences and ensures that there are no cross-constraints between plots. This also supports the interpretation of the time effects as showing non-linear effects only. Thus deviations from zero and the shape are interpretable. In line with this, the error bands are plus/minus to standard errors around zero.

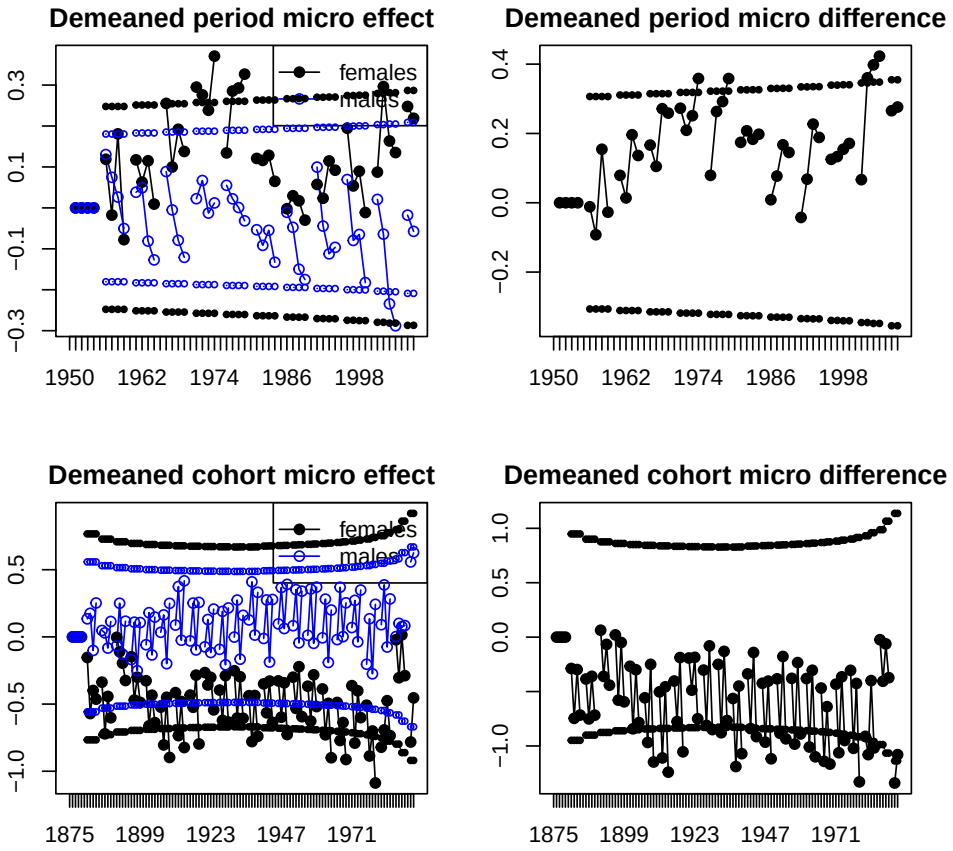
An alternative identification scheme is to set the two last age values and the first two period and cohort values to zero. This shows the underlying double sums of double difference. This is achieved by setting type to macro.ssdd.

7 Figure 3: Detrended micro effects

Next, we plot micro effects. The micro effects only exist when the data have mixed frequency as here. With $G = 5$ age groups $H = 1$ period groups only period and cohort micro effects exists.

```
> # Plot 1-sample fits jointly. Micro effects.
> par(mfrow=c(2,2),oma=c(0,0,2,0),mar=c(4,2,2,2)+0.1)
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="micro",which.plot=2)
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="micro",which.plot=5)
```

```
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="micro",which.plot=3)
> apc.plot.fit.2s(fit.apc.f,fit.apc.m,type="micro",which.plot=6)
```



8 Figure 4: F index

We will seek to replace the period effect with the F index. Only the non-linear part of the period effect is identified. Therefore only the non-linear part of the F index is relevant. Thus, we will plot a detrended version of the F index.

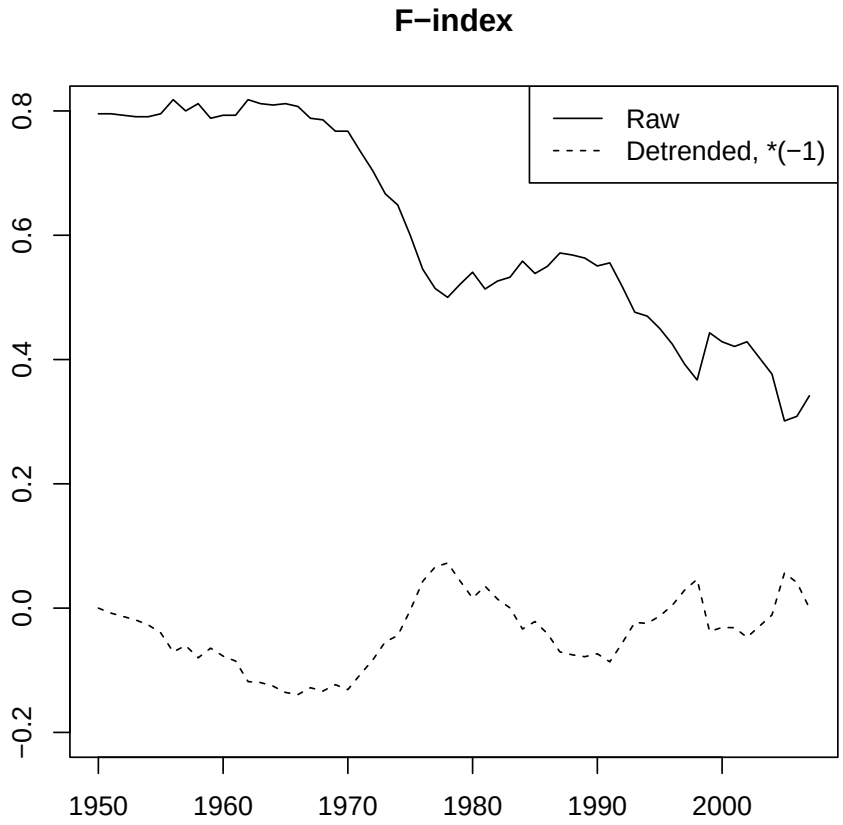
We start by detrending

```
> # Detrending F_index
> x.F         <- 1950:2007
> F_index.detrend         <- F_index-F_index[1] - (0:57)/57*(F_index[58]-F_index[1])
```

We now plot the original F index along with the detrended version. The detrended version is multiplied by -1 to match macro period effect better.

```
> # Plot F_index
> plot(x.F,F_index,type="l",ylim=c(-0.2,0.8),xlab="",ylab="")
> lines(x.F,-F_index.detrend,lty=2)
```

```
> title(main="F-index")
> legend("topright",legend=c("Raw","Detrended, *(-1)"),lty=c(1,2))
```



9 Table 5: Analysis of variance with external time series

We do an analysis of variance where the period effect is replaced by an external time series. This is done by first computing four analyses of variance. We estimate by OLS and by GLS. Note that F values are the same with the two estimation methods.

```
> # Analysis of variance, period replaced by time series
> s.f        <- sqrt(apc.fit.model(data.list.f,"log.normal.rates","APC")$s2)
> s.m        <- sqrt(apc.fit.model(data.list.m,"log.normal.rates","APC")$s2)
> table.gls.ts.apc<-apc.fit.table.2s(data.list.f,data.list.m,"gls.log.normal.rates","
> table.ols.ts.apc<-apc.fit.table.2s(data.list.f,data.list.m,"log.normal.rates","diff
> table.gls.ts.atc<-apc.fit.table.2s(data.list.f,data.list.m,"gls.log.normal.rates","
> table.ols.ts.atc<-apc.fit.table.2s(data.list.f,data.list.m,"log.normal.rates","diff
```

We then combine the results.

Model			Test vs APC			Test vs AC+F				
diff.	$-2\log L_{GLS}$	df	F	df	p_F	F	df	p_F	$\hat{\sigma}_{OLS}$	$\hat{\sigma}_{GLS}$
APC	-1129.99	1144							0.191	0.159
AC+F	-1058.80	1195	51	1.084	0.321				0.191	0.159
AC	-1052.62	1196	52	1.158	0.209	1	4.906	0.027	0.192	0.160

Table 2: Two-sample analysis of variance. Submodels refer to restrictions on cross-sample differenced predictor, while the common predictor is an unrestricted APC model. The AC+F model has the cross-sample difference period effect replaced with F-index. The structure of the table is close to that of Table 1. Columns 4-6 report F-tests against the unrestricted APC model. Columns 7-9 report F-tests against the AC+F model.

```
> # Combine analysis of variance results
> Table5<-cbind(table.ols.ts.apc[1:3,c(1,2,4,6,7)],
+ rbind(c(NaN,NaN,NaN),
+ table.ols.ts.atc[1:2,c(4,6,7)]),
+ table.ols.ts.apc[1:3,10],
+ table.gls.ts.apc[1:3,10])
> colnames(Table5)[c(1,9:10)]<-c("-2logL_OLS","sigma_OLS","sigma_GLS")
> Table5
```

References

Fannon, Z. and Nielsen, B. (2019) Age-period-cohort models. Oxford Research Encyclopedia of Economics and Finance. Oxford University Press. Download: Article: https://www.tandfonline.com/doi/full/10.1080/01621459.2017.1366908. Early version circulated as Nuffield Discussion Paper 2018-W04 https://www.nuffield.ox.ac.uk/economics/Papers/2018/2018W04_age_period_cohort_models.pdf.

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.

Nielsen, B. (2022) Two-sample age-period-cohort models with an application to Swiss suicide rates. *Download*: Nuffield Discussion Paper 2022-W03: https://www.nuffield.ox.ac.uk/economics/Papers/2022/2022-W03apc_2sample.pdf.

Riebler, A. and Held, L. and Rue, H. and Bopp, M. (2012) Gender-specific differences and the impact of family integration on time trends in age-stratified Swiss suicide rates. *Journal of the Royal Statistical Society, Series A*, 175, 479-490.