ONLINE RESOURCE 1

for

“Bayesian Population Forecasting: Extending the Lee-Carter Method”

Arkadiusz Wiśniowski*[1], Peter W.F. Smith[1],

Jakub Bijak[1], James Raymer[2,1], and Jonathan J. Forster[1]

[1]ESRC Centre for Population Change, University of Southampton

[2]Australian Demographic and Social Research Institute, Australian National University

A.1 Likelihood for the UK's migration data

The data on migration in the United Kingdom are obtained from the International Passenger Survey. They are disseminated as rounded to the nearest thousand. This renders the Poisson count model improper due to neglecting the variability resulting from rounding. Hence, for our purpose we utilize the following likelihood for the UK

*Contact email at A.Wisniowski@soton.ac.uk

emigration data:

$$
\begin{aligned}
Y_{x,t}^{E,k} \sim \Phi & \left(\frac{\log \left[\frac{1000 Y_{x,t}^{E,k} + 500}{R_{x,t}^{E,k}} \right] - \alpha_x^{E,k} - \beta_x^{E,k} \kappa_t^{E,k}}{\sigma^{E,k}} \right) \\
& -\Phi \left(\frac{\log \left[\frac{1000 Y_{x,t}^{E,k} - 500}{R_{x,t}^{E,k}} \right] - \alpha_x^{E,k} - \beta_x^{E,k} \kappa_t^{E,k}}{\sigma^{E,k}} \right) \mathbb{1}(Y_{x,t}^{E,k} > 0) \\
& -\Phi \left(\frac{\log \left[\frac{1000 Y_{x,t}^{E,k}}{R_{x,t}^{E,k}} \right] - \alpha_x^{E,k} - \beta_x^{E,k} \kappa_t^{E,k}}{\sigma^{E,k}} \right) \mathbb{1}(Y_{x,t}^{E,k} = 0), \qquad (1)
\end{aligned}
$$

and for immigration data:

$$
\begin{aligned}
Y_{x,t}^{I,k} \sim \Phi & \left(\frac{\log \left[1000 Y_{x,t}^{I,k} + 500 \right] - \alpha_x^{I,k} - \beta_x^{I,k} \kappa_t^{I,k}}{\sigma^{I,k}} \right) \\
& -\Phi \left(\frac{\log \left[1000 Y_{x,t}^{I,k} - 500 \right] - \alpha_x^{I,k} - \beta_x^{I,k} \kappa_t^{I,k}}{\sigma^{I,k}} \right) \mathbb{1}(Y_{x,t}^{I,k} > 0) \\
& -\Phi \left(\frac{\log \left[1000 Y_{x,t}^{I,k} \right] - \alpha_x^{I,k} - \beta_x^{I,k} \kappa_t^{I,k}}{\sigma^{I,k}} \right) \mathbb{1}(Y_{x,t}^{I,k} = 0), \qquad (2)
\end{aligned}
$$

where $\Phi(\cdot)$ denotes a cumulative distribution function of standardized normal distribution, $\mathbb{1}(f)$ is a indicator function taking one if f holds and zero otherwise and σ denotes standard deviation.

A.2 Smoothing of the migration data

We believe that the smoothing of the age profiles is necessary in the case of the UK data, due to the irregularities resulting from both rounding and the small samples in the underlying data (see Section 4.1).

Precision matrix for parameter β_x has been derived by using a formula for a conditional normal distribution, where the condition is that

$$
\beta_z = 1 - \sum_{i=1}^{z-1} \beta_i.
$$

In the case without smoothing, such as for mortality and fertility, the precision matrix Ψ_β from Eq. (12), of size $(z-1) \times (z-1)$, has form

$$\Psi_\beta = \begin{pmatrix} 2 & 1 & 1 & \dots & 1 \\ 1 & 2 & 1 & \dots & 1 \\ & & \vdots & & \\ 1 & 1 & 1 & \dots & 2 \end{pmatrix}.$$

In the case of migration, i.e. a model with smoothing, precision matrix Ψ_β from Eq. (15) of the same size as above, has form

$$\Psi_\beta = \begin{pmatrix} 2 & 0 & 1 & 1 & 1 & \dots & 1 & 2 \\ 0 & 3 & 0 & 1 & 1 & \dots & 1 & 2 \\ 1 & 0 & 3 & 0 & 1 & \dots & 1 & 2 \\ 1 & 1 & 0 & 3 & 0 & \dots & 1 & 2 \\ & & & & \dots & & & \\ 1 & 1 & \dots & 0 & 3 & 0 & 1 & 2 \\ 1 & 1 & \dots & 1 & 0 & 3 & 0 & 2 \\ 1 & 1 & \dots & 1 & 1 & 0 & 3 & 1 \\ 2 & 2 & \dots & 2 & 2 & 2 & 1 & 5 \end{pmatrix}.$$

In the application of the Lee-Carter model to the migration data we found that there are differences in smoothing between males and females but not between emigration and immigration. Hence, in our specification of the prior for the smoothing parameter τ_β we assumed that it is sex-specific and follows a heavy-tailed Gamma distribution:

$$\tau_\beta^k \sim \Gamma(0.00001, 0.00001). \tag{3}$$

A.3 OpenBUGS code for the fertility model

```
model{
#Priors
z <- 32
for (i in 1:z){                                           #z denotes age groups
```

```
   a[i] ~ dnorm(0,0.01)                                #parameter alpha
}

t.b[1]~dgamma(0.001,0.001)                             #parameter tau.beta
for (i in 1:31){
   for (j in 1:31){
      R[i,j]<-Psi[i,j]*t.b[1]                          #Psi read in as data
   }
}

for (i in 1:(z-1)){
   m.b[i]<-1/z}                                        #expectation of beta
b[1:31] ~ dmnorm(m.b[1:31],R[1:31,1:31])               #parameter beta
b[z] <- 1-sum(b[1:(z-1)])

m.p[1]<-0;m.p[2]<-0;                                   #hyperparameters for
t.p[1,1]<-1;t.p[2,2]<-1;t.p[2,1]<-0;t.p[1,2]<-0 #time series model
p[1:2] ~ dmnorm(m.p[1:2],t.p[1:2,1:2])                 #for kappa and gamma
p[3:4] ~ dmnorm(m.p[1:2],t.p[1:2,1:2])

sig[1]~dunif(0,100)                                    #prior for standard deviation
sig[2]~dunif(0,100)
sig[3]~dunif(0,100)
tau[1]<-1/(sig[1]*sig[1])                              #computing precision
tau[2]<-1/(sig[2]*sig[2])
tau[3]<-1/(sig[3]*sig[3])



#Model
#Time component kappa
k[1]<- 0

for (t in 2:35){
   m.k[t] <-p[1]+p[2]*k[t-1]
   k[t] ~ dnorm(m.k[t], tau[2])
}

#Cohort component gamma
g[1]<- 0

for (t in 2:z+75){
   m.g[t] <- p[3]+p[4]*g[t-1]
   g[t] ~ dnorm(m.g[t], tau[3])
   }

#Poisson-log-normal Likelihood
for (t in 1:35){
   for (i in 1:z){
      bir[i,t] ~ dpois(mb[i,t])                        #bir - births, mb - birth rate
```

```
      mb[i,t] <- ex[i,t]*lfr[i,t]               #ex - exposure
      lfr[i,t] ~ dlnorm(m.lfr[i,t],tau[1])      #lfr - log fertility rate
      m.lfr[i,t]<-a[i]+b[i]*k[t]+g[z+t-i]       #Lee-Carter model
   }
}


#Forecasting
#anchoring forecasts in the last observed year
m.k.f[1] <- p[1]+p[2]*k[35]
k.f[1] ~ dnorm(m.k.f[1], tau[2])

for (t in 2:15){
   m.k.f[t] <-p[1]+p[2]*k.f[t-1]
   k.f[t] ~ dnorm(m.kf[t], tau[2])
}

for (t in 1:15){
   for (i in 1:z){
      m.lfr.f[i,t]<-a[i]+b[i]*k.f[t]+g[z+35+t-i]
      lfr.f[i,t] ~ dlnorm(m.lfr.f[i,t],tau[1])
   }
}

}

#Initial values
list(
a=c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),
b=c(0.03125,0.03125,0.03125,0.03125,0.03125,0.03125,0.03125,
0.03125,0.03125,0.03125,0.03125,0.03125,0.03125,0.03125,
0.03125,0.03125,0.03125,0.03125,0.03125,0.03125,0.03125,
0.03125,0.03125,0.03125,0.03125,0.03125,0.03125,0.03125,
0.03125,0.03125,0.03125,NA),
t.b=c(1000),
k=c(NA,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),
sig=c(1,1,1)
)
```