Table S1: Interval forecast accuracy of mortality series for male and female using different univariate time series forecasting methods, as measured by mean interval score. For mortality, the interval scores were multiplied by 100 in order to keep two decimal places.

Evaluation Criteria	**Mortality ($\times$ 100)**		
Mean interval score	Male	Female	Total
ARIMA (2,1,1)*	2.67	1.48	2.23
est($\gamma = 0.2$)	5.36	3.02	5.86

*Our method uses ARIMA (2,1,1) to forecast univariate principal component scores.

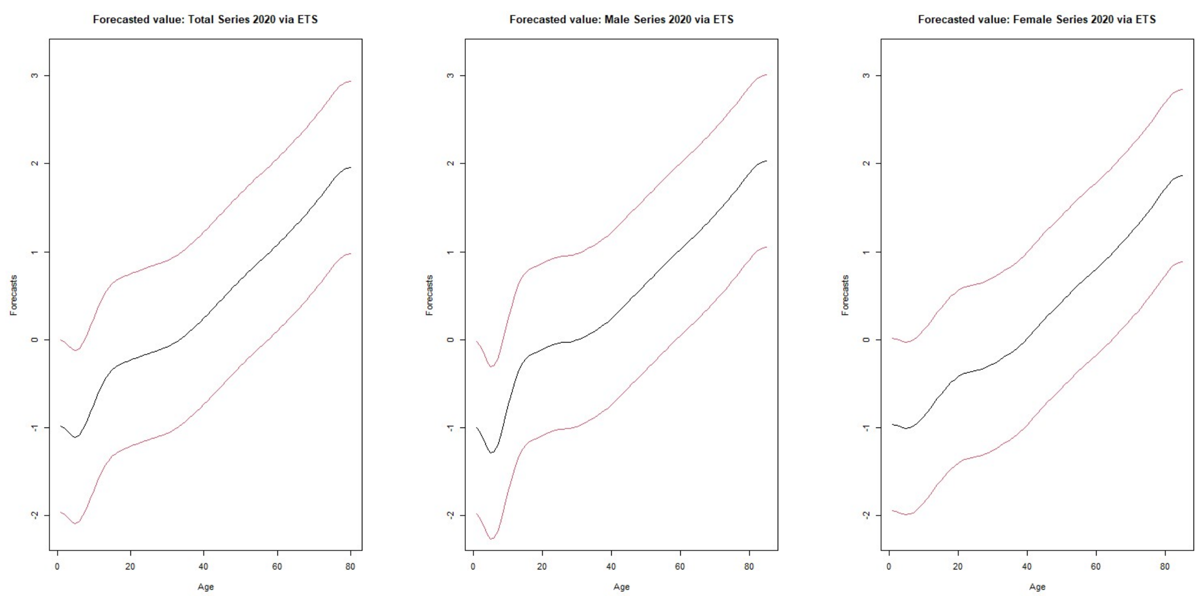

Figure S1. Interval (95% Prediction Interval) and point forecasted value of 2020 MR Canada: total (both sexes), male and female series using exponential smoothing technique.

R Code:

```
## A Functional Time Series Model for Forecasting Canadian Age Specific Mortality
#  CanMat:  a dataframe object with 17 rows and 30 columns
#           containing the log mortality values of 5 years age group
#           and years 1991 through 2019

# provide directory of the RData
getwd()
setwd("C:/Lenevo/Lenevo_laptop/RData")

# Installing and loading some required packages
install.packages("fda")
install.packages("ftsa")
install.packages("demography")
install.packages("forecast")
install.packages("rainbow")
install.packages("fields")
install.packages("ROMIplot")
library(fda)
library(ftsa)
library(demography)
library(forecast)
library(rainbow)
library(ROMIplot)
library(fields)


# Loading Canada Mortality Rate (Total series)
can.mort.log<-read.csv("Log_mortality_rate_Canada.csv",header=F)
```

```
medage<-read.csv("MedAge.csv",header=T)

myear<-read.csv("Mortality_Year.csv",header=T)

rownames(can.mort.log)<-medage$MedAge

colnames(can.mort.log)<-myear$Year

mat=as.matrix(can.mort.log)


# Ploting raw data for Total series

windows()

matplot(mat,type="l")

CanMat = mat

dim(CanMat)

CanLogMat_t = as.matrix(CanMat)


# Figure 1 Plot of total mortality rates in Canada for three selected years


  dimnames(CanLogMat_t)[[2]] <- paste('Year', 1991:2019, sep='')


  Fig1.data = cbind(CanLogMat_t[, c('Year1991', 'Year2001', 'Year2011')])

  nr=ncol(Fig1.data)

  CanTime = c(7,12,17,22,27,32,37,42,47,52,57,62,67,72,77,82,87)

  CanRng = c(7,87)

  quartz()

  matplot(CanTime, Fig1.data,

        type='l',lwd=2,xlab='Age',ylab='Log Death Rate',col=1:3,

        cex.lab=1.5,cex.axis=1.5)

  legend("topleft", colnames(Fig1data), col=seq_len(nr), cex=0.8,

       lty=seq_len(nr), lwd=2)

  title("Canada:Total Death Rate")
```

```
#  Smoothing log mortality observations (total series)---------------------#

# Obtaining smoothed functions for total series
  nbasis_t = 80
  norder_t = 6
  CanBasis_t = create.bspline.basis(CanRng, nbasis_t, norder_t)
  D2fdPar_t = fdPar(CanBasis_t, lambda=1e-7)
  CanLogMortfd = smooth.basis(CanTime, CanLogMat_t, D2fdPar_t)$fd
  tt=CanLogMortfd$coefs #smoothed values
  rownames(tt)<-seq(1:80)
  colnames(tt)<-myear$Year
  mort_fts.t<- rainbow::fts(x = 1:80, y = (tt),xname="Age",yname="Total Mortality Rates")#Main line
  colnames(mort_fts.t$y) <- seq_along(colnames(mort_fts.t$y))

# Canada Mortality Rate (Male)---------------------------------------------#
  can.mortm.log<-read.csv("Log_mortality_male_Canada.csv",header=F)
  mat_m=as.matrix(can.mortm.log)
  CanMatm = mat_m
  dim(CanMatm)
  CanLogMatm = as.matrix(CanMatm)

# Smoothing the log mortality observations for male series
  nbasis_m = 85
  norder_m = 6
  CanBasis_m = create.bspline.basis(CanRng, nbasis_m, norder_m)
  D2fdPar_m = fdPar(CanBasis_m, lambda=1e-7)
  CanLogMortmfd = smooth.basis(CanTime, CanLogMatm, D2fdPar_m)$fd
  mm=CanLogMortmfd$coefs #smoothed values
  mort_fts.m<- rainbow::fts(x = 1:85, y = (mm),xname="Age",yname="Male Mortality Rates")#Main line
```

```
  colnames(mort_fts.m$y) <- seq_along(colnames(mort_fts.m$y))


# Canada Mortality Rate (Female)------------------------------------------#
  can.mortf.log<-read.csv("Log_mortality_female_Canada.csv",header=F)
  matf=as.matrix(can.mortf.log)
  CanMatf = matf
  dim(CanMatf)
  CanLogMatf = as.matrix(CanMatf)

# Smoothing the log mortality observations for female series

  nbasis_f = 85
  norder_f = 6
  CanBasis_f = create.bspline.basis(CanRng, nbasis_f, norder_f)
  D2fdPar_f = fdPar(CanBasis_f, lambda=1e-7)
  CanLogMortffd = smooth.basis(CanTime, CanLogMatf, D2fdPar_f)$fd
  ff=CanLogMortffd$coefs #smoothed values
  mort_fts.f<- rainbow::fts(x = 1:85, y = (ff),xname="Age",yname="Female Mortality Rates")#Main line
  colnames(mort_fts.f$y) <- seq_along(colnames(mort_fts.f$y))


# Figure 2 Plot of smoothed functional log mortality series (a) to (c)

  windows()
  par(mfrow=c(1,3),mar=c(8,8,4,2))
  plot(mort_fts.m,col=rainbow(40),xlab="Age",
     ylab="Log Mortality Rates (per 1000 popluation)",
     main="Smoothed Mortality:Canada(Male)")
```

```
plot(mort_fts.f,col=rainbow(40),xlab="Age",
    ylab="Log Mortality Rates (per 1000 popluation)",
    main="Smoothed Mortality:Canada(Female)")
plot(mort_fts.t,col=rainbow(40),xlab="Age",
    ylab="Log Mortality Rates (per 1000 popluation)",
    main="Smoothed Mortality:Canada(Total)")

# Figure 3 Mean function for smoothed funcitonal mortality series

windows()
par(mfrow=c(1,1),mar=c(8,8,4,2))
plot(mean.fd(CanLogMortfd),main="Mean function",xlab="Age",ylim=c(-1,2.2),col="black")
lines(mean.fd(CanLogMortmfd),col="red",lty=2)
lines(mean.fd(CanLogMortffd),col="green",lty=3)
title("Smoothed Mean Function of Log Mortality Rate Canada: Total(black), Male(Red) and
Female(Green)")

# Figure 4 Ten year (2020-2029) forecasted mortality rate of Canada for Total, Male and Female sereis
windows()
par(mfrow=c(1,3),mar=c(8,8,4,2))
plot(forecast(ftsm(mort_fts.t,order=2),h=10)) # Total Series
legend("topleft",c("2020","2029"),col=c("red","blue"),lty=1)
title("Forecsted for 2020-2029:Total (both Sex)")

plot(forecast(ftsm(mort_fts.m,order=2),h=10)) # Male Series
legend("topleft",c("2020","2029"),col=c("red","blue"),lty=1)
title("Forecsted for 2020-2029:Male")

plot(forecast(ftsm(mort_fts.f,order=2),h=10)) # Female Series
```

```
legend("topleft",c("2020","2029"),col=c("red","blue"),lty=1)
title("Forecsted for 2020-2029:Female")
```

Figure 5 Interval (95% CI) and point forecasted values of 2020 Mortality rate Canada

```
windows()
par(mfrow=c(1,3),mar=c(8,8,4,2))
canmort.t=forecast(ftsm(mort_fts.t,order=2),h=1)
#Plot the lower and upper bounds
plot(canmort.t,ylim=c(-1.5,2))
lines(canmort.t$lower,col=2);lines(canmort.t$upper,col=2)
title("Forecsted value:Total 2020")

canmort.m=forecast(ftsm(mort_fts.m,order=2),h=1)
#Plot the lower and upper bounds
plot(canmort.m,ylim=c(-1.5,2))
lines(canmort.m$lower,col=2);lines(canmort.m$upper,col=2)
title("Forecsted value:Male 2020")

canmort.f=forecast(ftsm(mort_fts.f,order=2),h=1)
#Plot the lower and upper bounds
plot(canmort.f,ylim=c(-1.5,2))
lines(canmort.f$lower,col=2);lines(canmort.f$upper,col=2)
title("Forecsted value:Female 2020")
```

Figure 6, 7 and 8 Principal Component Regression Output (Total, Male and Female Series)

```
plot(forecast(ftsm(mort_fts.t,order=2),h=1),"components") # total series

plot(forecast(ftsm(mort_fts.m,order=2),h=1),"components") # Male series
```

```
  plot(forecast(ftsm(mort_fts.f,order=2),h=1),"components") # Female series


# Figure 9 image plot of age effect measuring for total series
  windows()
  par(mfrow=c(1,2),mar=c(8,8,4,2))
  image.plot(as.matrix(medage),as.matrix(myear),CanLogMat_t,
         xlab="age",ylab="age",cex.lab=1.5,cex.axis=1.5)
  ROMI.plot(Dx = NULL, Nx = NULL, mx = NULL, smooth = TRUE)




#----------------------------End-----------------------------------#
```