eAsia Summer School. Time-series regression in Public Health

Practical 2. Distributed Lag Non-linear Model

In this practical, students will learn how to investigate lagged associations using the distributed lag non-linear model (DLNM) framework. The practical will guide students through the key steps for applying DLNMs to explore the bi-dimensional exposure-lag-response relationship between an outcome variable and a specific environmental exposure. Students will analyze real world data using R functions and packages to estimate and visualize lagged nonlinear associations and interpret the results.

First, install and load the necessary libraries and functions to do this practical.

```
# Remove all objects in current session
rm(list = ls())

# Install packages
install.packages("tsModel", "splines", "dlnm", "Epi")

# Load packages
library(tsModel); library(splines); library(dlnm); library(Epi)

# Load useful functions
source("qAIC.R")
```

In this practical, you will analyze a time-series dataset for the city of Valencia, Spain, between 2001-2007. The dataset contains many variables, but for this exercise, we will focus on daily mortality counts and mean temperature (ºC).

```
# Load data
data <- read.csv("valencia.csv")
names(data)
head(data)
str(data)

# Format variable
data$date <- as.Date(data$date)
data$hol <- as.factor(data$hol)

# Generate time variables
data$time <- seq(nrow(data))
data$dow  <- as.factor(weekdays(data$date))
```

1. Descriptive analysis

First, we make daily time-series plots for death counts and temperature, respectively.

```
par(mex=0.8,mfrow=c(2,1))
  # All-cause mortality
  plot(data$date, data$all, xaxt="n", type="l", col= "red", ylim=c(0,50),
       ylab="Num. of deaths", xlab="Date")
  axis(1, at =data$date[c(1,diff(data$year))==1], labels = 2000+1:7)
```

```
  # Mean temperature
  plot(data$date, data$tmean, xaxt="n", type="l", col= "blue", ylim=c(0,40),
       ylab="Temperature (ºC)", xlab="Date")
  axis(1, at =data$date[c(1,diff(data$year))==1], labels = 2000+1:7)
layout(1)
```

Then, make a scatterplot between death counts and temperature.

```
plot(data$tmean, data$all, ylim=c(0, 50),
     xlab="Temperature (ºC)", ylab="Num. of Deaths", pch=4, col="red")
```

Questions. What kind of association do you observe between daily mortality and mean temperature? Does it appear to be a positive or negative relationship? Can you tell if it is a linear or non-linear relationship?

Next, we examine the summary statistics for the main exposure temperature.

```
summary(data$tmean)
sd(data$tmean)
round(with(data,quantile(tmean, probs=c(.01, .025, .1, .5, .9, .975, .99))), 1)
```

2. Simple Time-series Regression Model

We fit a simple time series regression model to examine the linear relationship between daily mortality and mean temperature, with adjustments for time components in the mortality time series, the day of the week, and public holidays. We adjust for time components using a natural cubic spline with 6 degrees of freedom (df) per year to represent the intra-annual seasonality and inter-annual long-term trend.

```
# Linear model
  # number of years in the data
  numyears <- length(unique(data$year))
  # Natural cubic splines for seasonal and long-term trend with 6df per year
  nstime <- ns(data$time, df=6*numyears)

  modlin <- glm(all ~ tmean + nstime + dow + hol,
                family=quasipoisson(), data=data, na.action="na.exclude")
  summary(modlin)
  ci.exp(modlin, subset="tmean")
```

Questions. How do you interpret the parameter estimates for temperature? In this model, what is the assumption of the association between mortality and temperature? Can the model capture the lagged effect of temperature on mortality?

Next, we consider the moving averages of temperature to account for lagged effects. We first try a 7-day moving average. Then, we split this into two temperature moving averages. First is a 2-day moving average covering lag 0 to lag 1, while second is a 6-day moving average from lag 1 to lag 6.

2

```
# Moving average models
  # 7-day moving average of mean temperature
  data$tmean06 <- runMean(data$tmean, lags=0:6)

  modma3 <- glm(all ~ tmean06 + nstime + dow + hol,
                family=quasipoisson(), data=data, na.action="na.exclude")
  summary(modma3)
  ci.exp(modma3, subset="tmean06")

  # 2-day and 6-day moving averages of mean temperature
  data$tmean01 <- runMean(data$tmean, lags=0:1)
  data$tmean16 <- runMean(data$tmean, lags=1:6)

  modma3b <- glm(all ~ tmean01 + tmean16 + nstime + dow + hol,
                family=quasipoisson(), data=data, na.action="na.exclude")
  summary(modma3b)
  ci.exp(modma3b, subset="tmean01")
  ci.exp(modma3b, subset="tmean16")
```

Questions. In the first model, how do you interpret the parameter estimates for the moving average of temperature? What is the assumption of the lagged effects when we use the 7-day moving average? In the second model with two moving averages, how do you interpret the parameter estimates for temperature? Can you explain why the results are different?

3. Distributed lag model

We now use a more flexible method called the distributed lag model to accommodate the lagged effects of temperature. We first include all the lags into the same model. We do this using the cross-basis function in the ***dlnm*** package.

```
# Distributed lag model (unconstrained)
  maxlag <- 21 # set number of lags, excluding current day lag 0
  cbtmean.unc <- crossbasis(data$tmean, lag=maxlag, argvar=list(fun="lin"),
                            arglag=list(fun="integer"))
  summary(cbtmean.unc)
  dim(cbtmean.unc)

  mod.unc <- update(modma3, .~. -tmean06+cbtmean.unc)
  predtmean.unc <- crosspred(cbtmean.unc, mod.unc, by=1, cen=18)
  # when "lin" is used, the default reference value is 0
  # here we arbitrarily selected the median 18C as the reference

  # for 1C increase above reference, var=19
  plot(predtmean.unc, var=19, type="p", ci="bars", pch=16, cex=1, col="red",
       las=1, xlab="Lag Day", ylab="RR", xlim=c(0,21), ylim=c(0.98,1.02),
       main="Unconstrained Distributed Lag Model")
```

Questions. Can you understand the summary and dimension of the cross-basis object created above? Can you describe the pattern in the plot? What is the reason for the observed pattern?

3

We can try constraining the shape of the lags using a smoothing spline. We do this by changing the argument for the lag-response association within the cross-basis function. The codes below show the lags were constrained using a natural cubic spline with 5 df (equivalent to 3 equally spaced knots). The temperature exposure variable is assumed linear.

We can extract the effect estimate provided as a relative risk (RR) at a specific temperature and lag. In the example below, we do this for temperature 19°C and lag 0.

```
# Distributed lag model (constrained using natural cubic spline)
  cbtmean.spl <- crossbasis(data$tmean, lag=maxlag,
                            argvar=list(fun="lin"),
                            arglag=list(fun="ns", df=5)) # 3 knots
  mod.spl <- update(modma3, .~. -tmean06+cbtmean.spl)
  predtmean.spl <- crosspred(cbtmean.spl, mod.spl, by=1, cen=18)
  # for 1C increase above reference, var=19
  plot(predtmean.spl, var=19,
       type="p", ci="bar", pch=16, cex=1,
       col="red", las=1, xlab="Lag Day", ylab="RR",
       xlim=c(0,21), ylim=c(0.99,1.01),
       main="Lags Constrained using Moving Average")

# Obtain RR for 1C increase above reference=18C at lag 0
  predtmean.spl$matRRfit["19","lag0"]
  predtmean.spl$matRRlow["19","lag0"]; predtmean.spl$matRRhigh["19","lag0"]
```

Questions. Can you describe the pattern of the lag-response association after applying a smoothing spline to constrain the lags? How do you interpret the RR and 95% confidence interval obtained above? What do you think about the linear assumption for the exposure-response association between temperature and mortality?

4. Distributed lag non-linear model

We can extend the above distributed lag model to accommodate non-linear exposure-response associations. Below we show an example to specify the cross-basis for temperature. Instead of linear association, now the temperature variable is smoothed using a quadratic B-spline with 2 equally spaced knots. The 3 knots for the lags are now specified in logarithmic scale. We re-center the prediction at minimum mortality temperature (MMT) where mortality risk is the lowest. We can plot the 3-D exposure-lag-response risk surface. We can also plot the lag-response curve at specific temperatures. Below we selected 7°C and 28°C corresponding to the 2.5[th] and 97.5[th] percentile temperature for cold and heat effect, respectively. Finally, we can obtain the overall cumulative exposure-response curve which accumulates delayed effects across all 21 lags.

```
# Distributed lag non-linear model
  # Number of lag days
  maxlag <- 21
  # number of years in the data
  numyears <- length(unique(data$year))
  # Natural cubic splines for seasonal and long-term trend with 6 df per year
  nstime <- ns(data$time, df=6*numyears)

  # Crossbasis for temperature
```

```
    varknot <- equalknots(data$tmean, nk=2) # 2 equally spaced knots
    lknots <- logknots(maxlag, 3) # 3 knots at log scale

    cbtemp <- crossbasis(data$tmean, lag=maxlag,
                        argvar=list(fun="bs", degree=2, knots=varknot),
                        arglag=list(fun="ns", knots=lknots))
    summary(cbtemp)

    # quasi-Poisson regression model - main model
    mod <- glm(all ~ cbtemp + nstime + dow + hol, data=data,
               family=quasipoisson(), na.action="na.exclude")
    pdtemp <- crosspred(cbtemp, mod, by=1)

    # Get prediction centered at the MMT (re-center)
    mmt <- pdtemp$predvar[which.min(pdtemp$allRRfit)]
    pdtempcen <- crosspred(cbtemp, mod, cen=mmt, by=1)

    # Plot exposure-lag-response risk surface
      plot3d <- plot(pdtempcen, shade=0.05, theta=235, phi=30, ltheta=-135,
                     col="antiquewhite", xlab="Temperature", zlab="RR",
                     main="Exposure-lag-response risk surface")
      # lag-exposure curve at temperature 28C and 7C
      lines(trans3d(x=28, y=0:maxlag, z=pdtempcen$matRRfit["28",],
                    pmat=plot3d), lwd=2.5, col="red")
      lines(trans3d(x=7, y=0:maxlag, z=pdtempcen$matRRfit["7",],
                    pmat=plot3d), lwd=2.5, col="blue")

    # Plot lag-response curve at specific temperature
    par(mfrow=c(1,2))
      plot(pdtempcen, var=28, type="l", ci="area",
           col="red", lwd=2, las=1, xlab="Lag", ylab="RR",
           xlim=c(0,maxlag), ylim=c(0.95,1.2), main="Lag-response at 28ºC")

      plot(pdtempcen, var=7, type="l", ci="area",
           col="blue", lwd=2, las=1, xlab="Lag", ylab="RR",
           xlim=c(0,maxlag), ylim=c(0.85,1.2), main="Lag-response at 7ºC")

    # Plot overall cumulative exposure-response curve
      plot(pdtempcen, "overall", lwd=2, col=1, ci.arg=list(col="antiquewhite"),
           xlab="Temperature (ºC)", ylab="RR",
           main="Overall cumulative exposure-response")
      abline(v=c(7,mmt,28), lty=(c(2,1,2)), lwd=c(1.2,1,1.2))
      text(x=c(7,mmt,28)+1.3, y=c(rep(2.7,3)), labels=c("2.5th","MMT","97.5th"))
```

Questions. Do you agree with the number of knots and their placement for the variable and the lag dimension? Why do we need to re-center the estimation of the association? Can you describe the pattern in the 3-D plot? Describe the shape of the heat and cold lags; do the effects appear immediate or delayed? Describe the shape of the overall cumulative exposure-response curve? Is it linear or non-linear?

We can extract the estimated effects of heat and cold as below.

```
# Extract the effect estimates
  # heat risk
    with(pdtempcen, cbind(allRRfit, allRRlow, allRRhigh))["28",]
```

```
# cold risk
  with(pdtempcen, cbind(allRRfit, allRRlow, allRRhigh))["7",]
```

Questions. Can you interpret the above estimates for heat and cold? What is the baseline/reference of these effect estimates?

Some researchers may explore a different definition for heat and cold. For example, instead of MMT, different baseline is used for heat and cold at the 90th and 10th percentile, respectively. We can do this by re-centering the prediction as below:

```
# Heat (99th vs 90th percentile) based on the second model
  # re-center at 90th percentile
  pdtempcen3 <- crosspred(cbtemp2, mod2, cen=26, by=1)
  with(pdtempcen3, cbind(allRRfit, allRRlow, allRRhigh))["29",]
# Cold (1st vs 10th percentile)
  # re-center at 10th percentile
  pdtempcen4 <- crosspred(cbtemp2, mod2, cen=10, by=1)
  with(pdtempcen4, cbind(allRRfit, allRRlow, allRRhigh))["6",]
```

Questions. Compare the new estimates for heat and cold to the ones obtained previously. Why are they different?

5. Sensitivity analysis

Below, we extract some model fit statistics which are useful for model comparison and sensitivity analysis.

```
# Model fit
  # qAIC
    qAIC(mod, type="dev")
  # Sum of absolute PACF of residuals
    res.mod <- residuals(mod, type="response")
    pacf.mod <- pacf(res.mod, na.action=na.omit)
    sum(abs(pacf.mod$acf))
```

Finally, we check the sensitivity of our main model to the number of knots with a different placement. We increase the number of knots for temperature variable to three, placed at the 10th, 75th and 90th percentiles which are not equally distanced.

```
# Sensitivity to number of knots with a different placement
    pct <- quantile(data$tmean, prob=c(.10, .75, .90), na.rm=TRUE)
    varknot <- pct[c(1,2,3)] # knots at 10th, 75th, 90th percentile

    cbtemp2 <- crossbasis(data$tmean, lag=maxlag,
                    argvar=list(fun="bs", degree=2, knots=varknot),
                    arglag=list(fun="ns", knots=lknots) )
    summary(cbtemp2)

    mod2 <- glm(all ~ cbtemp2 + nstime + dow + hol, data=data,
```

```
           family=quasipoisson(), na.action="na.exclude")
pdtemp2 <- crosspred(cbtemp2, mod2, by=1)

  # qAIC
    qAIC(mod2, type="dev")
  # Sum of absolute PACF of residuals
    res.mod2 <- residuals(mod2, type="response")
    pacf.mod2 <- pacf(res.mod2, na.action=na.omit)
    sum(abs(pacf.mod2$acf))

  # Get prediction centered at the MMT
    mmt2 <- pdtemp2$predvar[which.min(pdtemp2$allRRfit)]
    pdtempcen2 <- crosspred(cbtemp2, mod2, cen=mmt2, by=1)

# Compare main model to alternative model
plot(pdtempcen, "overall", lwd=3, col=1, ci.arg=list(col="antiquewhite"),
     xlab="Temperature (ºC)", ylab="RR",
     main="Overall cumulative exposure-response")
lines(pdtempcen2, "overall", lwd=2.5, lty=2, col=4, ci="area",
      ci.arg=list(density=20, col=4, angle=75))
```

Questions. Compare the two curves. Can you identify the portion of curves that differ between the two models? Why?