
Downloaded from: http://researchonline.lshtm.ac.uk/id/eprint/1888/

DOI: https://doi.org/10.1097/EDE.0b013e3181fdcd99

Usage Guidelines:

Please refer to usage guidelines at https://researchonline.lshtm.ac.uk/policies.html or alternatively contact researchonline@lshtm.ac.uk.

Available under license: http://creativecommons.org/licenses/by-nc-nd/2.5/
Online Supplemental Material for the article
”The impact of heat waves on mortality”

Antonio Gasparrini*† and Ben Armstrong

Department of Social and Environmental Health Research
London School of Hygiene and Tropical Medicine, UK

Contents

S1 Modelling choices 2
  S1.1 Covariates ................................................................. 2
  S1.2 Main effect of temperature ............................................. 2
  S1.3 Added effect during HWs ............................................. 3

S2 Sensitivity analysis 4

S3 Residual and correlation analysis 6

S4 R and Stata code 7
  S4.1 R code (first part) ...................................................... 8
  S4.2 Stata code ................................................................. 12
  S4.3 R code (second part) .................................................. 12

Bibliography 16

*Correspondance: Antonio Gasparrini, London School of Hygiene and Tropical Medicine, 15-17 Tavistock Place, London WC1H 9SH, UK
†E-mail: antonio.gasparrini@lshtm.ac.uk
S1 Modelling choices

The city-specific model was defined in the manuscript as:

$$\log[E(Y_i)] = \alpha + \sum_{j=1}^{P} g_j(x_{ij}) + m(t_i) + w(t_i)$$  \hspace{1cm} (S1.1)

The following sections provide some further justifications about the choices on the functions to describe the effects of covariates $g_j(x_j)$, the main $m(t)$ and the added $w(t)$ effects of temperature.

S1.1 Covariates

As explained in the text, the covariates $x_j$ included in the model in (S1.1) are day of the week, dew point temperature, long time trend and seasonality. Their inclusion and specification is decided independently from statistical significance and actual confounding effect in the city-specific estimates, following the rationale of the NMMAPS analysis (Dominici et al., 2005, 2003).

Day of the week is specified as 6 indicator variables, while dew point temperature is characterized through a natural cubic spline with 3 df, 2 knots at equally-spaced percentiles. The effect of seasonality is modelled through a natural cubic spline with 4 df (3 equally-spaced knots), in order to describe the variation within the summer period considered here (June-September). This effect is supposed to remain constant across different years, following the assumptions of other analyses published earlier (Analitis et al., 2008; Baccini et al., 2008; Michelozzi et al., 2009). These studies used an indicator variable for month in order to model the seasonal effect. We use a similar number of df (1 per month), but describing the effect through a smooth function. Long time trend is included as a natural cubic spine with 3 df (2 equally-spaced knots), to capture the residual temporal variability.

S1.2 Main effect of temperature

The main effect of temperature $m(t)$ is specified by a cross-basis, a specific set of functions which can describe simultaneously the relationship both in the space of the predictor (temperature) and in the lags (Armstrong, 2006; Gasparrini et al., 2010). This choice allows a strong control of potentially non-linear and lagged effect, also accounting for short-time harvesting (if present), and is motivated by the need to accurately control for the effect of daily temperature occurrences. Given the strong correlation between the parameters used to describe the main and added effect, a weak control for the former might produce biased estimates for the latter, due to residual confounding effect.

The cross-basis functions can be described as tensor-products between the basis functions used to define the relationship in each dimension. Specifically, we use here a cubic spline with 6 df (without natural constraints, 3 knots at equally spaced values) to specify the dependency along the dimension of temperature, and a natural cubic spline with 5 df (3 knots at equally spaced values in the log scale, plus intercept) for the distributed lag effects, with 30 df overall. The maximum lag is fixed at 10, a period of time long enough to include delayed effects and short time harvesting.

We found that the fit of the model improves when relaxing the linearity constraints of the spline at
the boundaries of temperature distribution, using the same amount of df. This may be attributed to
a strong non-linear effect of heat at very high temperatures, which is better described by the spline
without natural constraints. The days showing high temperatures are likely to be defined as HW days:
an underestimation of the main effect in this range can therefore result in an overestimation of the
added effect.
We keep a natural cubic spline for the dimension of the lag in order to specify more knots with the
same df (for the natural cubic splines $df = k + 1$, while for a simple cubic spline $df = k + 3$, with $k$
number of knots). The knots are placed at equally-spaced values in the log scale (0.8, 1.9, 4.4 lags),
assuring enough flexibility in the first lags, where more variability is expected (Muggeo, 2008; Peng
and Dominici, 2009).

S1.3 Added effect during HWs

The different HW definitions used in the first analysis with the simple indicator variables follow from
choices already proposed in the literature (Anderson and Bell, 2009; Hajat et al., 2006). Regarding the
second analysis on the effect of consecutive HW days, we fixed the threshold to the 97$^{th}$ city-specific
percentile in order to obtain a suitable amount of HW days, and we pooled the results using a meta-
analytical technique based on the multivariate extension of the method of moment estimator of Der
Simonian and Laird (Jackson et al., 2010; White, 2009).
Given that many cities show only short HW periods, the maximum length is set to 10 days, coherently
with the time frame used to specify the cross-basis functions for the main effect. HW days beyond that
point will keep the value of 10. As explained in the manuscript, cities with maximum duration less
than 10 days may contribute only to a subset of parameters of the two functions, strata and quadratic
B-spline.
S2  Sensitivity analysis

The robustness of the results to the various choices adopted in our modelling approach was tested through a sensitivity analysis. The main results obtained by varying the parameter of the functions $g_j$, $s$ and $w$ in model (S1.1) are reported in the paper. Here we provide additional sensitivity analyses on the choices regarding the function $f$ of consecutive HW days, evaluating graphically the differences for Figure 1 in the main text.

In particular:

- **13 days**: extending the maximum HW consecutive days to 13.
- **only 10 days**: restricting the analysis to the subsample of cities showing HW periods of at least 10 days (49 cities).
- **98th**: using the 98th percentile as a cut-off to define consecutive HW days.
- **REML**: using restricted maximum likelihood as estimation procedure for multivariate meta-analysis.

The results are summarized in Figure S1.

The shape of the curve obtained by the original model in the main text does not seem to be strongly influenced by the changes listed above. Increasing the maximum number of consecutive HW days to 13 only slightly postpones the peak in risk. This result suggests that the risk is not confined to the first 10 HW days, but that additional effects can be associated to longer HW periods. Furthermore, this might be compensated by some harvesting effect at longer lags, as previously pointed out (Hertel et al., 2009; Kaiser et al., 2007; Le Tertre et al., 2006). The subsample of cities with maximum HW length of at least 10 days shows approximately the same relationship, indicating that the results are robust to city selection up to this point. Anyway, only a limited number of cities actually shows very long HW’s, and this selection precludes the generalizability of the results beyond this HW length. Applying a more stringent definition for consecutive HW days based on the 98th percentile reveals a similar effect, but starting earlier within the HW periods. The results are robust to the estimation method selected for the multivariate meta-analysis, as expected given the large sample of cities.

The R and Stata code of the main analysis is included in Section S4. The reader is free to perform further sensitivity checks changing the code directly.
Figure S1: Sensitivity analysis for the added effect (consecutive HW days)

(a) Extending to 13 consecutive HW days

(b) Subsample of cities with 10-days HW’s

(c) Choosing 98th percentile as threshold

(d) REML
In this Section we provide an analysis restricted to the city of Chicago, where two important HWs occurred in August 1988 and, particularly infamous, in July 1995. The results showed here are computed from the model where the added effect is specified with a continuous measure of consecutive HW days, defined using the 97\textsuperscript{th} percentile and 2 days of minimum duration. The correlation between mean temperature and HW terms is not very high, as in the rest of the NMMAPS cities. The coefficient $r$ is 0.39 using the simple HW indicator and 0.33 for consecutive HW days. Figure S2 illustrates the temperature distribution in HW and non-HW days. The plot shows a substantial overlap between the two distributions, due to the fact that HW days are defined not just in terms of temperature but also of duration, thus explaining the low correlation with the HW indicator.

Figure S2: Temperature distribution in HW and non-HW days

The analysis of standardized residuals suggests a good fit in general of the model, as illustrated in Figure S3. However, it is possible to detect 2 outliers, corresponding to 2 days in July 1995 (under predicted) and August 1988 (over predicted).

More specifically, as depicted in Figure S4, the model predicts the mortality quite well: in periods identified as HW days, the average observed-predicted number of deaths are 122.4-122.6 (12\textsuperscript{th}-18\textsuperscript{th} of August 1988) and 261.3-242.2 (13\textsuperscript{th}-16\textsuperscript{th} of July 1995).
Figure S3: Distribution, Q-Q plot and series of standardized residuals

Figure S4: Observed and predicted mortality during August 1988 and July 1995

S4 R and Stata code

R and Stata code to reproduce the main results of the analysis are included below. The first part of the R code (Section S4.1) performs the first-stage (city-level) model and store the results in a file readable from Stata, saved in the current directory. The Stata code (Section S4.2) then runs the multivariate meta-analysis and store the results in other Stata files. Finally, the second part of the R code (Section S4.3) imports the estimates back to R and produces the results for the first and second analysis reported in the paper.

Additional information on the specific analytical steps are provided as comments within the code. The reader should pay attention to run the code in the order explained above.
S4.1 R code (first part)

```r
require(dlnm);require(Epi);require(tsModel)
require(NMMAPSlite);require(metafor);require(foreign)

# FUNCTION TO CREATE AN HEAT WAVE INDICATOR FOR A TEMPERATURE SERIES
# BASED ON THE THRESHOLD AND THE DURATION, BY GROUPS
fun.hw.thr <- function(x,thr,dur,group=NULL) {
  as.numeric(apply(Lag(x>=thr,0:(dur-1),group=group),
                     1,sum,na.rm=T)>(dur-1))
}

# INITIALIZE THE DATASET
initDB()
cities <- listCities()

# CREATE THE MATRICES TO STORE THE RESULTS
# DESCRIPTIVE STATS
descr.tmean <- matrix(NA,length(cities),7, dimnames=list(cities,
                          names(summary(c(1:10,NA)))))
hw.N <- matrix(NA,length(cities),6, dimnames=list(cities,
                      paste("hw",rep(c(2,4),each=3),rep(c(97,98,99),2),sep=".")))
hw.cons <- matrix(NA,length(cities),4, dimnames=list(cities,
                             c("N","Max",">3",">7")))
# REGRESSION MODELS
main.eff <- added.eff <- matrix(NA,length(cities),12,
                                dimnames=list(cities,paste("hw",rep(c(2,4),each=6),rep(c(97,98,99),
                                                   each=2),c("est","sd"),sep=".")))
strata.eff <- matrix(NA,length(cities),5,dimnames=list(cities,1:5))
strata.vcov <- vector("list",length(cities)) ; names(strata.vcov) <- cities
quad.eff <- strata.eff
quad.vcov <- strata.vcov
# MEAN SUMMER TEMPERATURE
meantemp <- 0

#########################################################################
# START THE LOOP FOR CITIES
time <- proc.time()
```

---

S4.1 R code (first part)
for(i in seq(length(cities))) {

  # LOAD AND PREPARE DATASET
  datatot <- readCity(cities[i], collapseAge = T)
  datatot$tmean <- (datatot$tmpd-32)*5/9
  datatot$time <- 1:nrow(datatot)
  datatot$year <- as.numeric(substr(datatot$date,1,4))
  datatot$month <- as.numeric(substr(datatot$date,6,7))
  datatot$doy <- sequence(tapply(datatot$year,datatot$year,length))
  datatot$dp01 <- filter(datatot$dptp,c(1,1)/2,side=1)
  percentiles <- quantile(datatot$tmean,c(75,97:99)/100,na.rm=T)
  data <- datatot[datatot$month%in%6:9,]

  # SAVE DESCRIPTIVE STATISTICS FOR TEMPERATURE
  descr.tmean[i,7] <- sum(is.na(data$tmean))
  meantemp[i] <- mean(data$tmean,na.rm=T)

  # CREATE THE CROSSBASIS FOR THE MAIN TEMPERATURE-MORTALITY RELATIONSHIP
  # CENTERED ON 75TH PERCENTILE, REFERENCE VALUE FOR PREDICTED EFFECTS
  range <- round(range(data$tmean,na.rm=T),0)
  ktemp <- range[1] + (range[2]-range[1])/4*1:3
  basis <- crossbasis(data$tmean,group=data$year,vartype="bs",vardegree=3,
                      varknots=ktemp,lagdf=5,maxlag=10,cenvalue=percentiles[1])

  # FIRST ANALYSIS: INDICATOR FOR DIFFERENT HW DEFINITIONS
  hw.def <- cbind(rep(percentiles[2:4],2),rep(c(2,4),c(3,3)))

  # HW DEFINITIONS
  hw <- fun.hw.thr(data$tmean,hw.def[k,1],hw.def[k,2],data$year)
  hw.N[i,k] <- sum(hw)

  # RUN THE MODEL FOR EACH DEFINITION
  for(k in 1:nrow(hw.def)) {

    # CREATE HEATWAVE INDICATOR FOR THE SPECIFIC HW DEFINITION
    hw <- fun.hw.thr(data$tmean,hw.def[k,1],hw.def[k,2],data$year)
    hw.N[i,k] <- sum(hw)

    # RUN THE MODEL
S4.1 R code (first part) R and Stata code

```r
model.first <- glm(death ~ hw + basis + dow + ns(year,3) +
    ns(doy,df=4) + ns(dp01,df=3), family=quasipoisson(), data)

# SAVE MAIN EFFECT
if(sum(hw)>0) {
    tmedian <- median(data$tmean[hw==1],na.rm=T)
    pred <- crosspred(basis,model.first,
        at=c((range[1]+1):(range[2]-1),tmedian))
    main.eff[i,c(k*2-1,k*2)] <- cbind(pred$allfit,
        pred$allse)[as.character(tmedian),]
} else main.eff[i,c(k*2-1,k*2)] <- c(NA,NA)

# SAVE ADDED EFFECT
added.eff[i,c(k*2-1,k*2)] <- ci.lin(model.first)["hw",1:2]
}
```

Explanation:

The code starts by fitting a generalized linear model (GLM) to the data using the `glm` function in R. The formula `death ~ hw + basis + dow + ns(year,3) + ns(doy,df=4) + ns(dp01,df=3)` specifies that the death count is modeled as a function of heatwave (hw), a basis (basis), day of the week (dow), natural splines of year with 3 degrees of freedom (ns(year,3)), natural splines of day of year with 4 degrees of freedom (ns(doy,df=4)), and natural splines of dp01 with 3 degrees of freedom (ns(dp01,df=3)). The family parameter is set to `quasipoisson`, indicating that a quasi-Poisson distribution is used.

Next, the code checks if there are any heatwave days in the data. If so, it calculates the median of the temperature on days with heatwaves, denoted by `tmedian`. Using this median value, it then calls the `crosspred` function, which performs cross-validation for predicting the main effect of the heatwave variable. The result is stored in the `main.eff` matrix, with the row indices corresponding to the current iteration and the columns indexing the heatwave variable and its added effect.

If there are no heatwave days, the corresponding entries in the `main.eff` matrix are set to NA, indicating that no prediction can be made.

The code then calculates the added effect by calling the `ci.lin` function to compute confidence intervals for the linear effect of the heatwave variable, and stores these in the `added.eff` matrix.

The second analysis involves creating a heatwave indicator and consecutive term using the 97th percentile. The code first computes the heatwave indicator and consecutive term using the `fun.hw.thr` function. It then creates a heatwave consecutive days variable up to 10 days and saves statistics for consecutive heatwave days. The code also creates strata of consecutive heatwave days and runs a model with these strata, specifying additional terms such as `dow`, natural splines of year, and natural splines of day of year.

Finally, the code saves the related coefficients and variance-covariance matrix, including those that are missing.
strata.eff[i,!index1%in%index2] <- ci.lin(model.strata)[index,1]
strata.vcov[[i]] <- matrix(NA,length(index1),length(index1))
strata.vcov[[i]][!index1%in%index2,!index1%in%index2] <-
vcov(model.strata)[index,index]

# CREATE THE SPLINE OF CONSECUTIVE HW DAYS
quad <- bs(hw.lin,knots=c(2,5,8),Bound=c(0,10),degree=2)
# RUN THE MODEL
model.quad <- glm(death ~ basis + quad + dow + ns(dp01,df=3) +
  ns(year,3) + ns(doy,df=4),family=quasipoisson(), data)
# SAVE THE RELATED COEF AND VCOV (INCLUDING MISSING)
index1 <- grep("quad",names(coef(model.quad)))
index2 <- (1:length(coef(model.quad)))[is.na(coef(model.quad))]
index <- index1[!index1%in%index2]
quad.eff[i,!index1%in%index2] <- ci.lin(model.quad)[index,1]
quad.vcov[[i]] <- matrix(NA,length(index1),length(index1))
quad.vcov[[i]][!index1%in%index2,!index1%in%index2] <-
vcov(model.quad)[index,index]
}
proc.time()-time
# TAKES APPROXIMATELY 5-6 MIN IN A 2GHz LAPTOP

########################
# TO STATA
########################
index <- cbind(rep(1:5,5),rep(1:5,each=5))
names <- c(paste("b",1:5,sep="_"),
  paste("V",rep(1:5,5),rep(1:5,each=5),sep="_"))
temp1 <- temp2 <- matrix(0,length(cities),length(names))
for(i in 1:length(cities)) {
  temp1[i,] <- c(strata.eff[i,],strata.vcov[[i]][index])
  temp2[i,] <- c(quad.eff[i,],quad.vcov[[i]][index])
}
colnames(temp1) <- colnames(temp2) <- names
library(foreign)
write.dta(as.data.frame(temp1),"strata.dta")
write.dta(as.data.frame(temp2),"quad.dta")
S4.2 Stata code

*cd "..."
set more off

* QUAD MM
use quad, clear
mvmeta b V, mm bscov
matrix b = e(b)
matrix V = e(V)
clear
svmat b
svmat V
save quad_mm, replace

* STRATA MM
use strata, clear
mvmeta b V, mm bscov
matrix b = e(b)
matrix V = e(V)
clear
svmat b
svmat V
save strata_mm, replace

S4.3 R code (second part)

#################
# FROM STATA (STATA CODE SHOULD HAVE BEEN RUN)
#################

quad.pool.est <- as.matrix(read.dta("quad_mm.dta")[1,1:5])
quad.pool.vcov <- as.matrix(read.dta("quad_mm.dta")[1:5,6:10])
strata.pool.est <- as.matrix(read.dta("strata_mm.dta")[1,1:5])
strata.pool.vcov <- as.matrix(read.dta("strata_mm.dta")[1:5,6:10])

###############################
# RESULTS: DESCRIPTIVE STATISTICS
###############################
# SUMMARY FOR TMEAN
summary(descr.tmean[,c("Mean","NA's")])

# TOTAL NUMBER OF HW DAYS UNDER DIFFERENT HW DEFINITIONS
summary(hw.N)

# CONSECUTIVE HW DAYS (WITH 97TH PERCENTILE)
# % OF CITIES WITH MAX LENGTH >7 AND >9
sum(hw.cons[,"Max"]>6)/nrow(hw.cons)*100
sum(hw.cons[,"Max"]>9)/nrow(hw.cons)*100

# % OF CONSECUTIVE HW DAYS ABOVE 3 AND 7
colSums(hw.cons[,c(">3",">7")])/sum(hw.cons[,"N"])*100

# RESULTS: FIRST ANALYSIS

label <- paste("hw",rep(c(2,4),each=3),rep(c(97,98,99),2),sep=".")
table1 <- matrix(NA,6,7,dimnames=list(label,
c("N comm","Est.main","95%CI.main","P-het.added","Est.added",
"95%CI.added","P-het.added")))

for(i in 1:6) {
  added.eff[added.eff[,2*i]==0,c(2*i-1,2*i)] <- NA
  main.eff[is.na(added.eff[,2*i]),c(2*i-1,2*i)] <- NA

  # RUN THE META-ANALYSIS
  pool.main <- rma.uni(yi=main.eff[,2*i-1],sei=main.eff[,2*i])
  pool.added <- rma.uni(yi=added.eff[,2*i-1],sei=added.eff[,2*i])

  table1[i,] <- c(sum(!is.na(added.eff[,2*i-1])),
               round(exp(pool.main$b)*100-100,1),
               paste(round(exp(pool.main$b-1.96*pool.main$se)*100-100,1),"to",
               round(exp(pool.main$b+1.96*pool.main$se)*100-100,1)),
               round(pool.main$QEp,3),
               round(exp(pool.added$b)*100-100,1),
               paste(round(exp(pool.added$b-1.96*pool.added$se)*100-100,1),"to")
}
round(exp(pool.added$b+1.96*pool.added$se)*100-100,1)),
  round(pool.added$QEp,3))
}

# TABLE 1 IN THE MANUSCRIPT
table1

# RESULTS: SECOND ANALYSIS

# CREATE THE BASIS VARIABLES FOR PREDICTION
x <- 0:100/10
x.quad <- bs(x,knots=c(2,5,8),degree=2,Bound=c(0,10))
x.strata <- mkbasis(0:20/2,type="strata",knots=c(1,2,4,6,8))$basis

# PLOT
quad.plot <- cbind(x.quad%*%t(quad.pool.est),
                   sqrt(diag(x.quad%*%quad.pool.vcov%*%t(x.quad))))
plot(x,exp(quad.plot[,1]),type="n",ylim=c(0.95,1.10),yaxt="n",
     xlab="Number of consecutive HW days",frame.plot=F)
axis(2,labels=-1:2*5,at=0.95+0:3*0.05)
polygon(c(x,rev(x)),c(exp(quad.plot[,1]+1.96*quad.plot[,2]),
                   rev(exp(quad.plot[,1]-1.96*quad.plot[,2]))),border=NA,col=grey(0.9))
abline(h=1)
lines(x,exp(quad.plot[,1]))

strata.plot <- cbind(x.strata%*%t(strata.pool.est),
                      sqrt(diag(x.strata%*%strata.pool.vcov%*%t(x.strata))))
lines(0:20/2,exp(strata.plot[,1]),type="S",lty=2)
References


