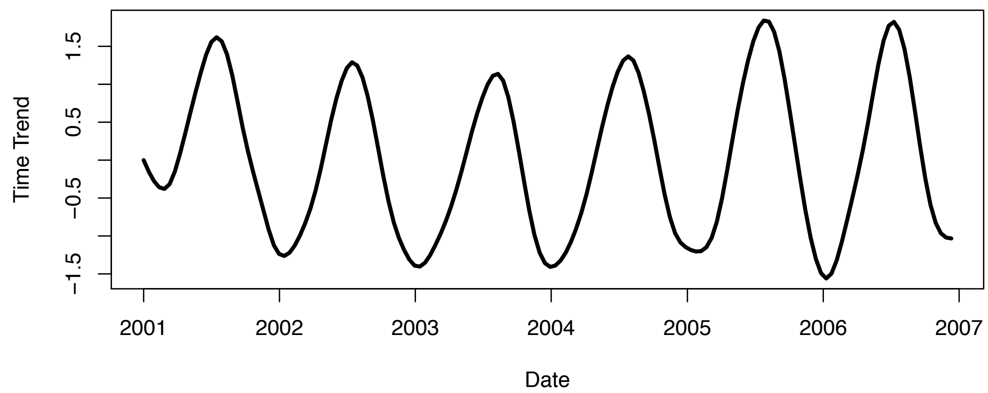
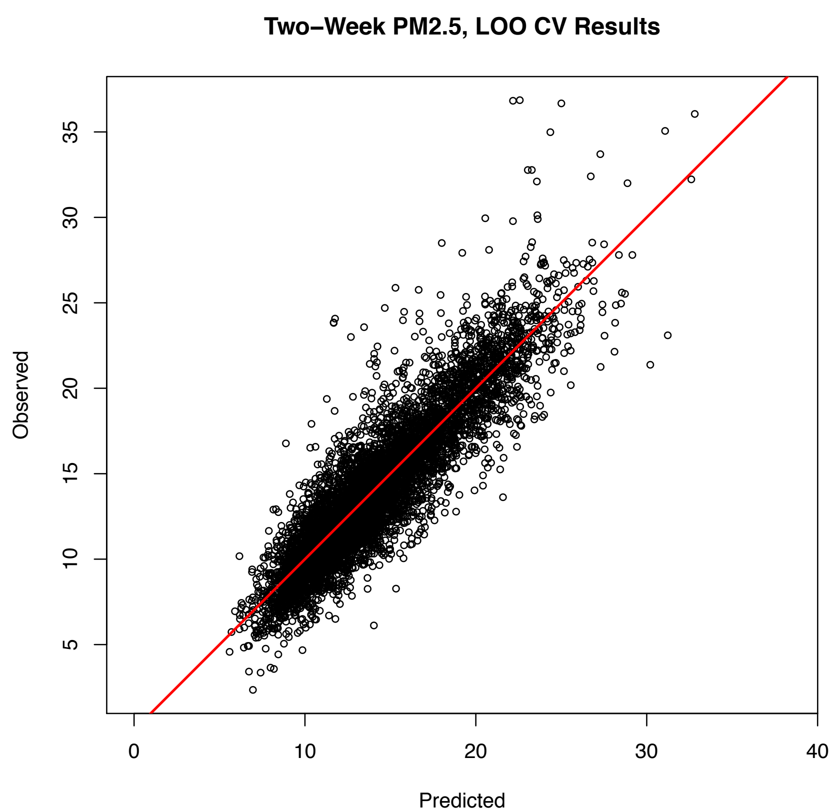
Supplementary Digital Content for "Measurement error correction for predicted spatiotemporal air pollution exposures."

**eTable 1.** Parameter estimates for spatiotemporal model of ambient $PM_{2.5}$.

| | Estimate | Standard Error |
|---|---|---|
| **Spatiotemporal Covariates** | | |
| Temperature (10° C) | 0.037 | 0.037 |
| Wind Speed (10 m/s) | –0.249 | 0.060 |
| Relative Humidity (10%) | –0.015 | 0.010 |
| **Long-term Mean** | | |
| Intercept | 1.744 | 1.117 |
| Percent Forest Cover (%) | –0.003 | 0.001 |
| Distance to Emissions Point Source (10 km) | –0.059 | 0.025 |
| Log of Range Parameter (km) | 4.179 | 0.506 |
| Log of Sill Parameter | –4.507 | 0.314 |
| **Time Trend Coefficients** | | |
| Intercept | 0.073 | 0.044 |
| Elevation (100m) | 0.013 | 0.006 |
| Percent Forest Cover (%) | 0.001 | <0.001 |
| Local Road Length (10 km) | –0.030 | 0.015 |
| Highway Length (10 km) | –0.056 | 0.026 |
| Log of Range Parameter (km) | 5.253 | 0.635 |
| Log of Sill Parameter | –5.371 | 0.534 |
| **Spatiotemporal Residual** | | |
| Log of Range Parameter (km) | 6.491 | 0.094 |
| Log of Sill Parameter | –3.130 | 0.075 |
| Log of Nugget Parameter | –4.782 | 0.030 |




**eFigure 1.** $PM_{2.5}$ time trend estimated from monitoring data. Trend is shown on the standardized scale.




**eFigure 2.** Observations and cross-validated predictions of two-week average $PM_{2.5}$ concentrations ($\mu g/m^3$) from 2002 through 2006. The 1-1 line is drawn for reference.

**eTable 2.** Estimated difference in birth weight (in grams) associated with 1 μg/m³ higher ambient average $PM_{2.5}$ during the specified trimester, among births to mothers in all counties. Bootstrap corrections are based upon 1,000 bootstrap samples.

| Cohort | Correction | Trimester | Estimate | Std. Err |
|---|---|---|---|---|
| Statewide | Non-Parametric | 1 | 0.43 | 0.55 |
| | | 2 | −0.65 | 0.56 |
| | | 3 | −0.51 | 0.58 |
| | Parameter | 1 | 0.28 | 0.68 |
| | | 2 | −0.63 | 0.67 |
| | | 3 | −0.65 | 0.61 |

**eTable 3.** Estimated difference in birth weight (in grams) associated with 1 μg/m³ higher ambient average $PM_{2.5}$ during the third trimester in the restricted cohort. The degrees of freedom (df) per year in the health model reflects different amount of temporal smoothing.

| df per year | Estimate | Std. Err | 95% CI |
|---|---|---|---|
| 1 | −0.70 | 0.39 | (−1.46, 0.05) |
| 2 | −0.75 | 0.69 | (−2.10, 0.59) |
| 3 | −1.98 | 0.77 | (−3.50, −0.46) |
| 4 | −2.36 | 0.78 | (−3.89, −0.83) |
| 5 | −2.31 | 0.79 | (−3.85, −0.77) |
| 6 | −2.44 | 0.80 | (−4.00, −0.88) |
| 7 | −2.38 | 0.80 | (−3.95, −0.80) |
| 8 | −2.33 | 0.80 | (−3.90, −0.75) |

**eAppendix 1.** Example R code for running non-parametric and parameter bootstrap.

```
# Example code for running the non-parametric bootstrap and the
# parameter bootstrap.

# First are three auxiliary functions,
# followed by a description of the R
# objects in the accompanying data file,
# followed then by code for running the
# bootstrap procedures.
# For each bootstrap (non-parametric and
# parameter), the code is in two parts.
# The first part deals with the exposure
# data, the second part with the health
# analysis.


############################
# create_timemx()
#
# Function for creating a list of time-space linkages
#
# INPUT:
#     gridIDs -- vector of grid IDs
#     dates -- optional data frame for supplying the conception, trimester,
#                  and birth dates
#     obsdates -- dates for which exposures are available. gestational
#                  dates will be matched to the closest date in this vector
#     concep.date -- conception dates
#     T2 -- dates of start of second trimester
#     T3 -- dates of start of third trimester
#     dob -- dates of birth
#
# OUTPUT: A list of length equal to the number of
#     unique grid IDs.  Each element in the list is
#     a set of three sparse matrices (one for each
#     trimester). The number of rows is equal to
#     the number of obsdates. The number of columns
#     depends upon how many births occurred in that
#     grid cell.  Columns of the matrix sum to 1,
#     with non-zero entries corresponding to the
#     interval of the corresponding trimester.
#     This is designed to allow for easily computing
#     averages in the bootstrap procedure by simple
#     matrix multiplication.
#
create_timemx <- function(gridIDs, dates=NULL, obsdates,
 concep.date=dates$concep.date, T2=dates$T2, T3=dates$T3, dob=dates$dob){
```

```
 T1.int <- findInterval(concep.date, obsdates)
 T2.int <- findInterval(T2, obsdates)
 T3.int <- findInterval(T3, obsdates)
 dob.int <- findInterval(dob, obsdates)

 newdates <- data.frame(T1.int=T1.int, T2.int=T2.int, T3.int=T3.int,
 dob.int=dob.int)
 ndates <- length(obsdates)

 grid.list = sort(unique(gridIDs))

 time.mx <- vector("list", length=length(grid.list))
 names(time.mx) <- grid.list

 for (i in 1:length(grid.list)) {
      grid.i = grid.list[i]

 ## Extract birth records in grid cell i
 use = which(gridIDs == grid.i)
 dat.i <- newdates[use,]

 ## Calculate exposure indicators
 time.mx[[i]]$T1 <-  Matrix(mapply(createExpInd,a = dat.i$T1.int, b =
 dat.i$T2.int, n= ndates), sparse=TRUE, dimnames=list(NULL, rownames(dat.i)))
 time.mx[[i]]$T2 <- Matrix(mapply (createExpInd,a = dat.i$T2.int, b =
 dat.i$T3.int, n= ndates), sparse=TRUE, dimnames=list(NULL, rownames(dat.i)))
 time.mx[[i]]$T3 <-  Matrix(mapply (createExpInd,a = dat.i$T3.int, b =
 dat.i$dob.int, n= ndates), sparse=TRUE, dimnames=list(NULL, rownames(dat.i)))
 }
 return(time.mx)
}

# Helper function for getting the indices
# corresponding to exposure. Creates
# an n-vector, with zeros everywhere
# except from indices a to b, in which
# there are equal values that sum to 1.
# This makes exposure assignment easily
# done through matrix multiplication.
createExpInd <- function(a, b, n) {
 x <- numeric(n)
 x[a:b] <- 1/(b-a + 1)
 return(x)
}


############################
```

```r
# make_exposure_assignment()
#
# Function for making exposure assignment
#
# INPUT:
#     obs -- matrix of exposure observations
#     time.list -- list output from create_timemx()
#
# OUTPUT:
#     expos -- matrix of average exposures
#           for each subject (row) and trimester (column)
#
make_exposure_assignment <- function(obs, time.list){

# Get location name list
locnamelist <- sapply(time.list, function(w) colnames(w$T1))
# Vector of location names, for assigning to rows of output matrix
locnames <- unlist(locnamelist, use.names=FALSE)
# Vector of location counts per grid, to allow for
# efficient index creation, all to avoid (slow!) name
# matching within the loop
indvec <- cumsum(c(1, sapply(locnamelist, length)))
expos <- matrix(data=0, nrow=length(locnames), ncol=3)
rownames(expos) <- locnames
colnames(expos) <- c("T1", "T2", "T3")

 for (i in 1:length(time.list) ){
      grid.i = names(time.list)[i]

      ## Extract 2-week time-series for grid cell i
      exp.i = obs[, colnames(obs) == grid.i]

      # ## Fill in exposure estimates
      T1 <- as.vector(exp.i %*% time.list[[i]]$T1)
      T2 <- as.vector(exp.i %*% time.list[[i]]$T2)
      T3 <- as.vector(exp.i %*% time.list[[i]]$T3)
      inds <- indvec[i]:(indvec[i+1]-1)
      expos[inds, ] <- cbind(T1, T2, T3)
 }
return(expos)
}
```

```r
##########################################################################

library(SpatioTemporal)
library(MASS)

#####################
#   Load the Data   #
#####################

load("GA_PM25_Birthweight.RData")

# Objects included:
# pm.stmodel -- the STmodel object containing the monitoring data
#                       and model specifications
# pm.stdata -- the STdata object containing the monitoring data
# grid.stdata -- an STdata object containing covariate information
#                        at grid locations. Used for making predictions.
# est.pm.stmodel -- an estimateSTmodel object containing parameter
#                             estimates for the exposure prediction model
# pm.aqsgrid.stmodel -- an STmodel object for simulating data
#                             at AQS and grid locations simultaneously.
# bwdata -- Synthetic birth weight data. These were generated using
#                 random dates, random locations, and random birth weights
#                 and thus differ from the observed data used in the
#                 manuscript analysis.

# Note: See the SpatioTemporal package help files for more
# information about structure of the STmodel and STdata objects.

#########################
#  Fit original models  #
#########################

fitT1 = lm (bwt~T1.exp, data = bwdata)
lmT1coef <- coef(fitT1)
lmT1sigma <- summary(fitT1)$sigma
fitT2 = lm (bwt~T2.exp, data = bwdata)
lmT2coef <- coef(fitT2)
lmT2sigma <- summary(fitT2)$sigma
fitT3 = lm (bwt~T3.exp, data = bwdata)
lmT3coef <- coef(fitT3)
lmT3sigma <- summary(fitT3)$sigma
```

```
##############################
#                            #
#  Non-Parametric Bootstrap  #
#                            #
##############################

# Extract spatial covariates
mon.covars <- pm.stdata$covars
# Spatiotemporal covariates
mon.st.covars <- pm.stdata$SpatioTemporal
# Monitor observations matrix
mon.obs <- createDataMatrix(pm.stdata)
# Monitor IDs
mon.list <- colnames(mon.obs)

#########################
#  NP Bootstrap, Part 1 #
#  Monitor Sampling    #
#########################

# Code shown here as a loop over B=2 iterations.
# These can also be run in parallel.
B <-2

for (i in 1:B){
 seed <- i
 set.seed(seed)

 mons.i <- sample(mon.list, replace=T)
 mon.obs.i <- mon.obs[, mons.i]
 colnames(mon.obs.i) <- 1:ncol(mon.obs.i)  # Give arbitrary monitor IDs
 mon.covars.i <- mon.covars[match(mons.i, mon.covars$ID), ]
 mon.covars.i $x[duplicated(mons.i)]  <-
 jitter(mon.covars.i$x[duplicated(mons.i)], amount=0.3) # Jitter by 300m in
 each direction
 mon.covars.i $y[duplicated(mons.i)]  <- jitter(mon.covars.i
 $y[duplicated(mons.i)], amount=0.3)
 mon.covars.i $ID <- 1:length(mons.i)  # Give arbitrary monitor IDs
 mon.st.covars.i <- mon.st.covars[, mons.i, ]
 dimnames(mon.st.covars.i)[[2]] <- 1:length(mons.i) # Give arbitrary IDs

 pm.stdata.i <- createSTdata(obs= mon.obs.i, covars= mon.covars.i,
 SpatioTemporal = mon.st.covars.i)
 pm.stdata.i$trend <- pm.stmodel$trend # Add the original trend
 pm.stdata.i$trend.fnc <- pm.stmodel$trend.fnc # Add the original trend

 # Model Setup
 LUR <- list (~ forest_2001 + dist_emiss_2002, ~elevation+forest_2001 +
```

```
 local_length + highway_length)
 cov.beta <- list(covf = "exp", nugget = F)
 cov.nu <- list (covf = "exp", nugget = T)
 locations <- list (coords =c("x", "y"), long.lat = c("long", "lat"))
 pm.stmodel.i <-createSTmodel(pm.stdata.i, LUR=LUR, ST=c("tmp", "wspd", "rh"),
 cov.beta=cov.beta, cov.nu=cov.nu, locations=locations)

 # Estimate model parameters
 ## This may take a long time! ##
 x.init<-cbind (c(3, -4, 3, -5, 3, -3, -5), rep (0, 7))
 est.pm.stmodel.i <- estimate (pm.stmodel.i, x.init, type="p")

 # Make predictions
 ## This may take a long time! ##
 grid.pred.temp.i <- predict(object= pm.stmodel.i , x= est.pm.stmodel.i,
 STdata= grid.stdata, pred.var=FALSE, Nmax=2000)
 EX.i <- grid.pred.temp.i$EX

 # Save the results, if desired
 stdatafilename <- paste0("nonparametric_bootstrap_stdata_seed", seed,
 ".RData")
 save(mons.i, EX.i, est.pm.stmodel.i, file=stdatafilename)
}

#########################
#  NP Bootstrap, Part 2 #
#   Health Analysis     #
#########################

obsdates <- seq(as.Date('2001/1/1'), as.Date('2006/12/11'), by="14 days")
# Create list of time-space linkage information, to facilitate
# simpler computation of trimester exposures
tm <- create_timemx(gridIDs= bwdata$grid, dates= bwdata, obsdates= obsdates)

# Matrix for estimates
coefMX <- matrix(NA, nrow=3, ncol=B, dimnames=list(c("T1", "T2", "T3"), 1:B))

# Loop over simulated exposures
for (i in 1:B){

 # Load the ST data for this bootstrap run
 stdata.boot.filename  <- paste0("nonparametric_bootstrap_stdata_seed", i,
 ".RData")
 load(stdata.boot.filename, verbose=TRUE)

 obsPred.i <- exp(EX.i)
 expPred <- make_exposure_assignment(obs= obsPred.i, time.list=tm)
 bwdata.i <- bwdata
```

```
 bwdata.i$T1.exp <- expPred[match(rownames(bwdata), rownames(expPred)), "T1"]
 bwdata.i$T2.exp <- expPred[match(rownames(bwdata), rownames(expPred)), "T2"]
 bwdata.i$T3.exp <- expPred[match(rownames(bwdata), rownames(expPred)), "T3"]

 subj.i <- sample(nrow(bwdata.i), replace=TRUE)
 bwdata.i <- bwdata.i[subj.i, ]

 # Fit Models
 fitT1.i = lm ( bwt~T1.exp, data = bwdata.i )
 coefs <- c(coef(fitT1.i)[2])
 fitT2.i = lm ( bwt~T2.exp, data = bwdata.i )
 coefs <- c(coefs, coef(fitT2.i)[2])
 fitT3.i = lm ( bwt~T3.exp, data = bwdata.i )
 coefs <- c(coefs, coef(fitT3.i)[2])

 coefMX[, i] <- coefs
}

# Bootstrap corrected estimate for Trimester 1 is:
2* lmT1coef["T1.exp"] - mean(coefMX["T1",])


##############################
#                            #
#    Parameter Bootstrap     #
#                            #
##############################

###########################
# Parameter Boot., Part 1 #
# Exposure Simulation     #
###########################

# Code shown here as a loop over B=2 iterations.
# These can also be run in parallel.
B <-2

for (i in 1:B){
 seed <- i
 set.seed(seed)

 # Simulate the data
 pm.sim <- simulate(pm.aqsgrid.stmodel, nsim=1, x=coef(est.pm.stmodel)$par,
 nugget.unobs=exp(coef(est.pm.stmodel)$par[length(coef(est.pm.stmodel)$par)]))
 # Put into space-time matrix form
 obs <- exp(createDataMatrix(obs=pm.sim$obs[[1]]$obs,
 date=pm.sim$obs[[1]]$date, ID=pm.sim$obs[[1]]$ID))
```

```
 # Create new STmodel object, with new 'monitor' data
 new.pm.stmodel <- pm.stmodel
 new.pm.stmodel$obs <- pm.sim$obs[[1]][pm.sim$obs[[1]]$ID %in%
 pm.stmodel$locations$ID,]

 # Sample new exposure parameters
 #
 Sigma <- solve(-est.pm.stmodel$res.best$hessian.all)
 parest <- mvrnorm(n=1, mu=est.pm.stmodel$res.best$par.all$par, Sigma=Sigma)

 # Make predictions
 # Note this may take a long time!
 grid.pred.temp <- predict(object= new.pm.stmodel , x= parest, STdata=
 grid.stdata, pred.var=FALSE, Nmax=2000)
 EX <- grid.pred.temp$EX
 rm(grid.pred.temp)

 # Save the results, if desired
 stdatafilename <- paste0("parameter_bootstrap_stdata_seed", seed, ".RData")
 save(obs, EX, parest, file=stdatafilename)
}

##########################
# Parameter Boot., Part 2 #
#   Health Analysis       #
##########################

# Create list of time-space linkage information, to facilitate
# simpler computation of trimester exposures
obsdates <- seq(as.Date('2001/1/1'), as.Date('2006/12/11'), by="14 days")
tm <- create_timemx(gridIDs= bwdata$grid, dates= bwdata, obsdates= obsdates)

# Matrix for estimates
coefMX <- matrix(NA, nrow=3, ncol=B, dimnames=list(c("T1", "T2", "T3"), 1:B))

# Loop over simulated exposures
for (i in 1:B){
 set.seed(1e5 + i)

 # Load the simulated exposure observations and predictions
 stfilename <- paste0("parameter_bootstrap_stdata_seed", i, ".RData")
 load(stfilename, verbose=TRUE)

 ########################################
 # Simulate new Health Data, using bootstrapped
 # exposure data at subject locations

 # Make Exposure Assignments
```

```
 exp <- make_exposure_assignment(obs=obs, time.list=tm)
 lmMMT1.i <- cbind(1, T1.exp=exp[match(rownames(bwdata),rownames(exp)), "T1"])
 lmMMT2.i <- cbind(1, T2.exp=exp[match(rownames(bwdata),rownames(exp)), "T2"])
 lmMMT3.i <- cbind(1, T3.exp=exp[match(rownames(bwdata),rownames(exp)), "T3"])

 # Simulate the health data
 T1bwt_mean <- lmMMT1.i %*% lmT1coef
 T2bwt_mean <- lmMMT2.i %*% lmT2coef
 T3bwt_mean <- lmMMT3.i %*% lmT3coef
 eps <- matrix(rnorm(n=3*nrow(bwdata)), nrow=nrow(bwdata), ncol=3)
 T1bwt <- T1bwt_mean + lmT1sigma*eps[, 1]
 T2bwt <- T2bwt_mean + lmT2sigma*eps[, 2]
 T3bwt <- T3bwt_mean + lmT3sigma*eps[, 3]

 #########################################
 # Compute predicted exposures, using
 # predictions developed from simulated
 # monitor data.

 # Make Exposure Assignments
 obsPred <- exp(EX)
 expPred <- make_exposure_assignment(obs=obsPred, time.list=tm)

 # Replace model matrices with predictions
 lmMMT1.i <- cbind(1, T1.exp= expPred[match(rownames(bwdata),
 rownames(expPred)), "T1"])
 lmMMT2.i <- cbind(1, T2.exp= expPred[match(rownames(bwdata),
 rownames(expPred)), "T2"])
 lmMMT3.i <- cbind(1, T3.exp= expPred[match(rownames(bwdata),
 rownames(expPred)), "T3"])

 fitT1.i <- solve(crossprod(lmMMT1.i), crossprod(lmMMT1.i, T1bwt))
 coefs <- c(T1=fitT1.i[2])
 rm(fitT1.i, lmMMT1.i); gc()
 fitT2.i <- solve(crossprod(lmMMT2.i), crossprod(lmMMT2.i, T2bwt))
 coefs <- c(coefs, T2=fitT2.i[2])
 rm(fitT2.i, lmMMT2.i); gc()
 fitT3.i <- solve(crossprod(lmMMT3.i), crossprod(lmMMT3.i, T3bwt))
 coefs <- c(coefs, T3=fitT3.i[2])
 rm(fitT3.i, lmMMT3.i); gc()

 coefMX[, i] <- coefs
}

# Bootstrap corrected estimate for Trimester 1 is:
2* lmT1coef["T1.exp"] - mean(coefMX["T1",])
```