Homework 9

Advanced Methods for Data Analysis (36-402)

Due Friday, April 15, 2022 at 3:00pm EDT

Solutions – not to be posted online or shared, even after the end of the semester.

You should **always show all your work** and submit both a writeup and *R* code.

- Assignments must be submitted through Gradescope as a PDF. Follow the instructions here: `https://www.cmu.edu/teaching/gradescope/`
- Gradescope will ask you to mark which parts of your submission correspond to each homework problem. This is mandatory; if you do not, grading will be slowed down, and your assignment will be penalized.
- Make sure your work is legible in Gradescope. You may not receive credit for work the TAs cannot read. **Note:** If you submit a PDF with pages much larger than 8.5×11", they will be blurry and unreadable in Gradescope.
- For questions involving R code, we strongly recommend using R Markdown. The relevant code should be included with each question, rather than in an appendix. A template Rmd file is provided on Canvas.

In this homework, you'll work with `chicago.csv`, a dataset of measurements of air quality in Chicago, Illinois each day from January 1987 to the end of December 2000. It also includes the number of non-accidental deaths recorded in Chicago each day and the mean temperature each day.

The variables are:

time The date of observation, given as the number of days before or after December 31, 1993

death The number of non-accidental deaths on that date

pm10median The median density of PM_{10} pollution, i.e. particulate matter with diameter less than 10 micrometers (milligrams per cubic meter)

o3median The median concentration of ozone (parts per billion)

so2median The median concentration of sulfur dioxide (SO_2)

tmpd The mean temperature (Fahrenheit)

Note that you can convert `time` to a Date object using

```
as.Date(chicago$time, origin = as.Date("1993-12-31"))
```

which allows you to use it in plots and to determine when specific observations happened. Note also that some of the pollution variables have been shifted, so they contain negative values.

The pollution and temperature variables also have versions whose names start with `lag_`. These versions are 7-day averages of the original variables; for example, `lag_pm10median` on day 7 is the average of `pm10median` on days 1 through 7. **For the purposes of this assignment, we will ignore the lagged variables.**

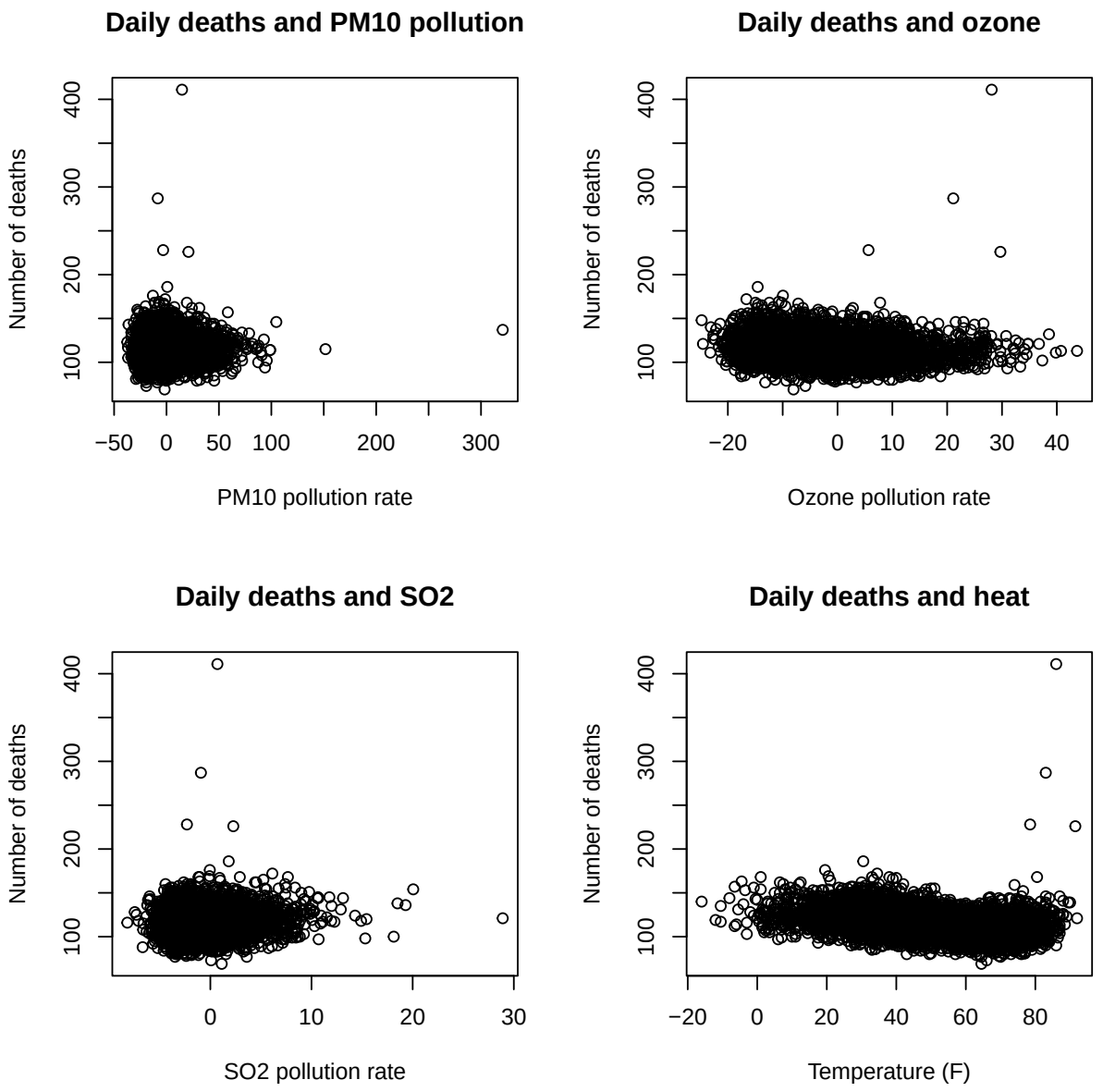
A few observations contain `NA` values. You can ignore these observations.

1. Doctors and epidemiologists believe that air pollution is an important contributor to mortality; people who live in areas with higher levels of air pollution tend to have higher death rates. It is also known that extreme temperatures tend to lead to higher death rates, for instance among poor or elderly residents who do not have heating or air conditioning.

   Conduct an exploratory data analysis to examine the relationships between each type of air pollution and mortality, as well as heat and mortality. What associations do you see? Do they appear linear, or will you need a nonlinear model?

Solution: The simplest EDA is scatterplots:

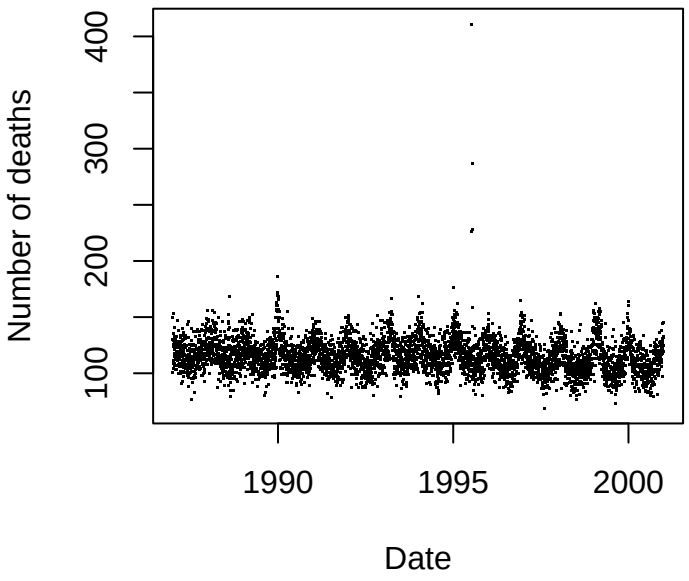
```
chicago <- read.csv("chicago.csv")
chicago$time <- as.Date(chicago$time, origin = as.Date("1993-12-31"))

# Limit the number of digits in R output
options(digits = 3)

par(mfrow = c(2, 2))

plot(death ~ pm10median, data = chicago,
     xlab = "PM10 pollution rate", ylab = "Number of deaths",
     main = "Daily deaths and PM10 pollution")
plot(death ~ o3median, data = chicago,
     xlab = "Ozone pollution rate", ylab = "Number of deaths",
     main = "Daily deaths and ozone")
plot(death ~ so2median, data = chicago,
     xlab = "SO2 pollution rate", ylab = "Number of deaths",
     main = "Daily deaths and SO2")
plot(death ~ tmpd, data = chicago,
     xlab = "Temperature (F)", ylab = "Number of deaths",
     main = "Daily deaths and heat")
```



It's hard to see strong relationships in the pollution plots; there may be slight trends, but the variance is quite high, making trends hard to see. We do see one sign of nonlinearity: daily deaths appear to decline as temperatures increase, until they suddenly increase when average daily temperatures are above 80 degrees Fahrenheit. This makes sense: when temperatures are high even at night, it is impossible for homes to cool down, and elderly people without air conditioning are at high risk of heatstroke.

Notice the obvious outliers here with very high death rates. We should always investigate outliers to see if we can explain them, so let's plot deaths over time to see what could explain these:

```
plot(death ~ time, data = chicago, pch = ".",
     xlab = "Date", ylab = "Number of deaths")
```



The spike in 1995 corresponds to the 1995 Chicago heat wave. In mid-July, abnormally high temperatures caused 739 heat-related deaths.

2. We'd like to establish if there is evidence the pollution variables are associated with death rates, and if so, if there is evidence the relationships are nonlinear. Consider three models, each with death rate as the outcome variable:

   Model 0 No predictors, only an intercept

   Model 1 Linear model predicting death rates using the three pollution variables and temperature

   Model 2 Additive model version of Model 1, in which each term is a spline smoother with 4 effective degrees of freedom

   Fit these models to the data. Briefly conduct diagnostics to check their fits.

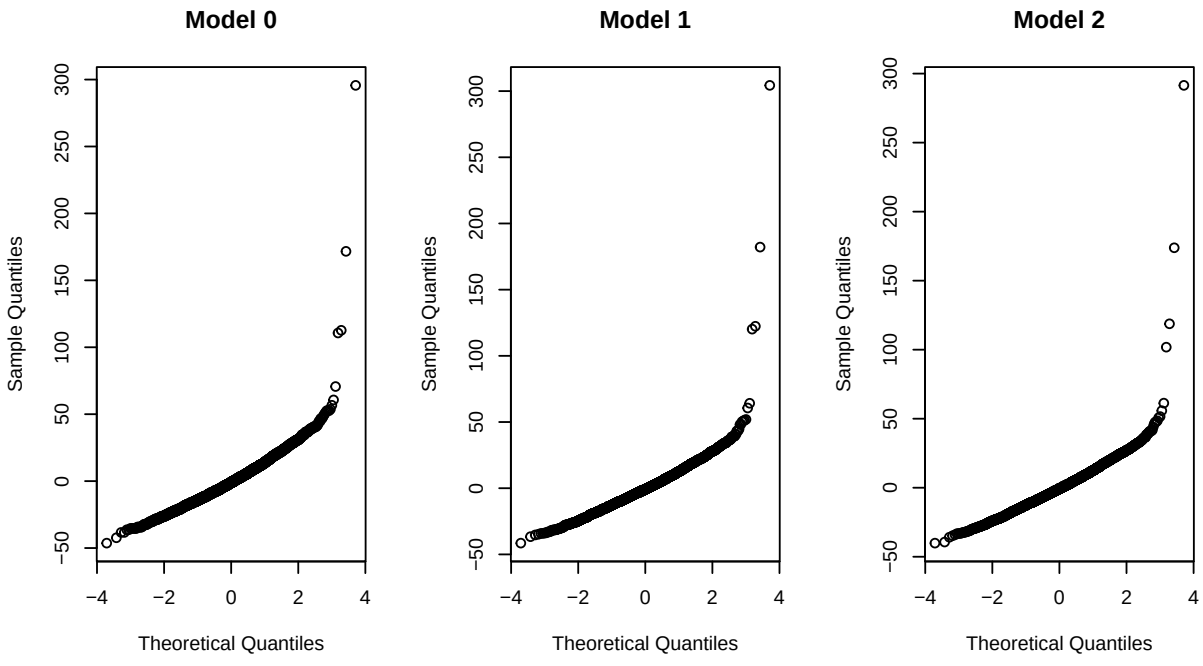
   Note: In order to compare the models, it will be necessary for them to be fit to the same data. Because a few observations of the variables are missing, Model 1 and Model 2 will omit a few rows of data. Be sure to fit Model 0 to only the rows of data with the variables present, so it's fit to the same data.

Solution: We can fit the three models all using `mgcv`, just for consistency. Using `lm` for the linear models would also be fine.

```
suppressPackageStartupMessages(library(mgcv))

## Warning:  package 'mgcv' was built under R version 4.1.2

complete_rows <- !is.na(chicago$pm10median) & !is.na(chicago$o3median) &
  !is.na(chicago$so2median) & !is.na(chicago$tmpd)

model0 <- gam(death ~ 1, data = chicago[complete_rows,])
model1 <- gam(death ~ pm10median + o3median + so2median + tmpd,
              data = chicago)
model2 <- gam(death ~ s(pm10median, k = 4 + 1, fx = TRUE) +
                s(o3median, k = 4 + 1, fx = TRUE) +
                s(so2median, k = 4 + 1, fx = TRUE) +
                s(tmpd, k = 4 + 1, fx = TRUE),
              data = chicago)
```

Let's examine the errors of each model:

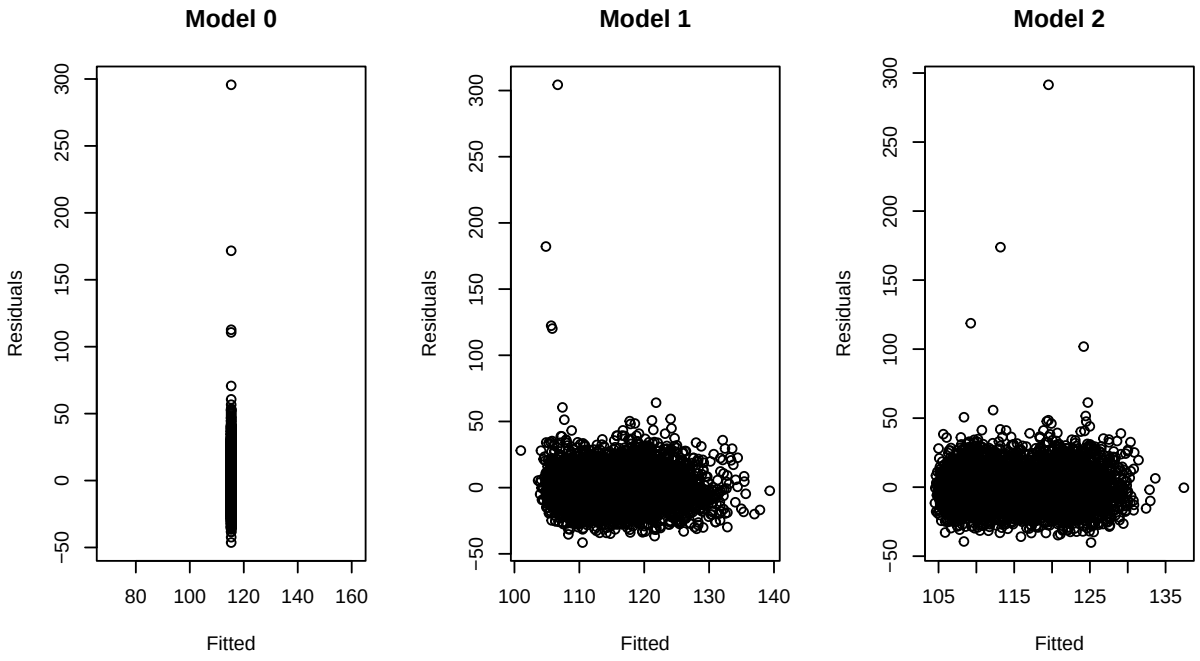
```
par(mfrow = c(1, 3))

qqnorm(residuals(model0), main = "Model 0")
qqnorm(residuals(model1), main = "Model 1")
qqnorm(residuals(model2), main = "Model 2")
```



These Q-Q plots may look alarming, but there are really only a few data points in the right tails that are problematic. Those are almost certainly the outlier days with very high death rates, as we saw in our EDA. Apart from those, normality doesn't appear to be a big problem.

Let's check for constant variance as well:

```
par(mfrow = c(1, 3))

plot(fitted(model0), residuals(model0), main = "Model 0",
     xlab = "Fitted", ylab = "Residuals")
plot(fitted(model1), residuals(model1), main = "Model 1",
     xlab = "Fitted", ylab = "Residuals")
plot(fitted(model2), residuals(model2), main = "Model 2",
     xlab = "Fitted", ylab = "Residuals")
```



As Model 0 is only an intercept, it only has one fitted value; there's not much to check for it. Model 1 and Model 2 appear okay, again apart from a few large outliers.

3. Describe the assumptions necessary to use an F test to test the differences between these models, and for the F statistic to be approximately F-distributed, and describe if these assumptions appear to be met.

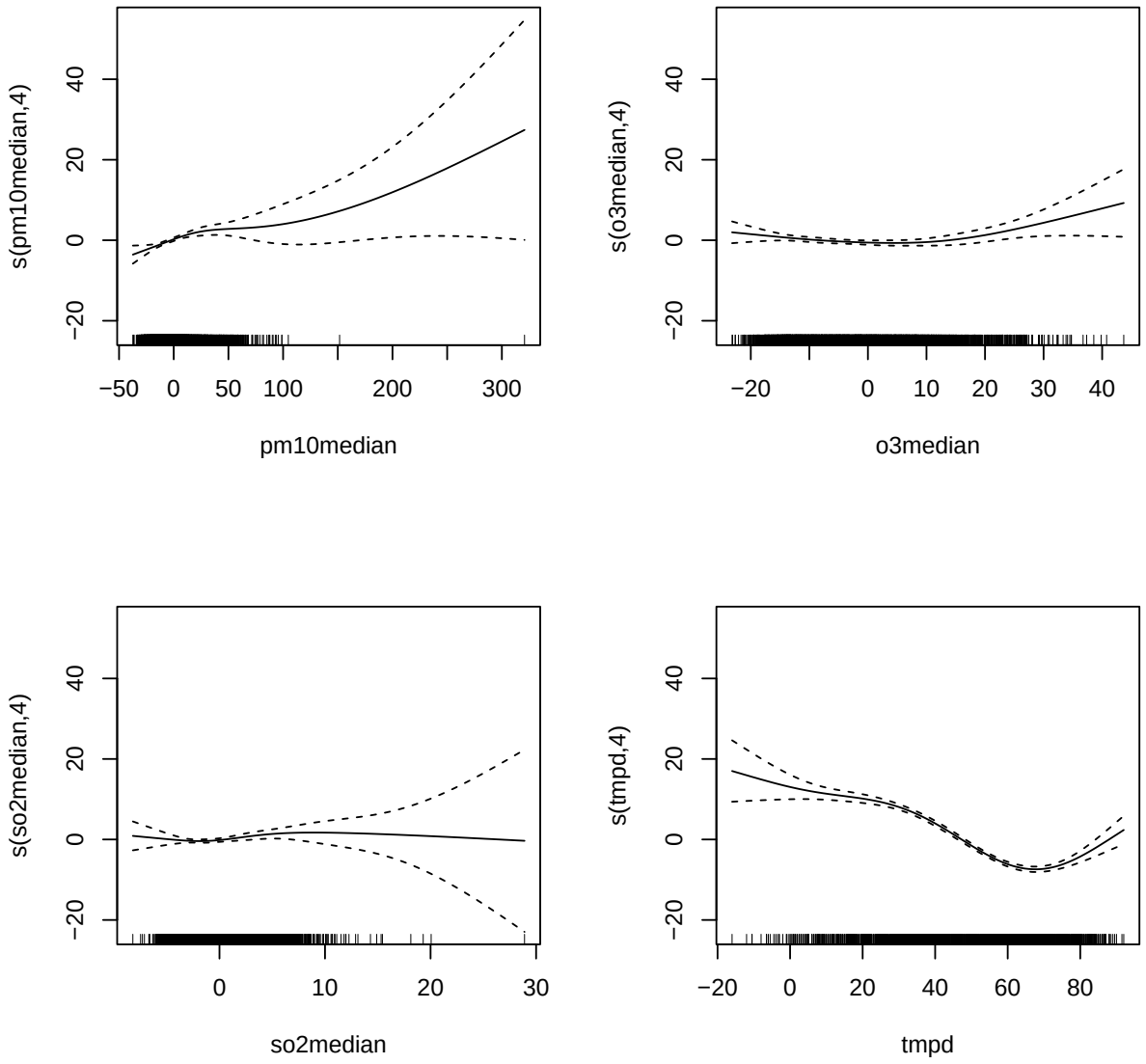
Solution: To conduct an F test comparing the models, we usually need one of the models to be nested within the other, and for the larger model to be approximately unbiased. See Lecture 9.

To have the test statistic be approximately F-distributed, we need the errors to be i.i.d. $N(0, \sigma^2)$, or to have a large sample size. Apart from the outliers we noted, these assumptions seem to hold.

4. Plot the partial response functions for Model 2. Comment on what you see. Visually, what is the relationship between each variable and death rate, and do the relationships appear strongly nonlinear?

Solution: We can use `plot` to get the partial response functions:

```
par(mfrow = c(2, 2))

plot(model2)
```



Temperature has the most obvious nonlinear relationship, with a dip around 60 degrees and sharp rise above 80 degrees. Ozone (o3median) also appears nonlinear, with a gentle increasing curve above levels of about 10. It's difficult to tell for PM_{10} and sulfur dioxide; for instance, the nonlinearity on the right side of PM_{10} appears to be driven entirely by two data points with very high values.

5. Conduct a hypothesis test to more formally determine if the nonlinear terms in Model 2 are necessary

when compared to Model 1. Use an F test and assume the approximate F distribution applies here, either because the assumptions are met or because of the large sample size.

Be sure to explicitly state the null hypothesis and the alternative hypothesis. When you report your result, state in words the test statistic, p value, and conclusion, as in this example:

> Using a [name of test], we [accept/reject] the null hypothesis and conclude that [substantive conclusion] ($F_{\text{df1,df2}} = X$, $p = Y$).

Formats like this are common in scientific writing, as they explicitly state the test statistic, its distribution, its degrees of freedom, and the p value, allowing readers to understand exactly what was done.

Solution: We can use `anova` to conduct the test:

```
ftest <- anova(model1, model2, test = "F")
ftest

## Analysis of Deviance Table
##
## Model 1: death ~ pm10median + o3median + so2median + tmpd
## Model 2: death ~ s(pm10median, k = 4 + 1, fx = TRUE) + s(o3median, k = 4 +
##     1, fx = TRUE) + s(so2median, k = 4 + 1, fx = TRUE) + s(tmpd,
##     k = 4 + 1, fx = TRUE)
##   Resid. Df Resid. Dev Df Deviance    F Pr(>F)
## 1      4836     962580
## 2      4824     929005 12    33575 14.5 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Our null hypothesis is that the linear model is approximately unbiased; the alternative hypothesis is that it is not and that at least some terms are nonlinear. (It is sufficient to state the null and alternative in words, provided they are clear and precise.) Or, in math, if $r(x) = \beta^T x + \delta(x)$, then

$$H_0 : \delta(x) = 0$$
$$H_1 : \delta(x) \neq 0.$$

Using an F test of nested models, we reject the null hypothesis and conclude that there are nonlinear relationships between some of the predictors and death ($F_{12,4824} = 14.529$, $p = 2.419 \times 10^{-30}$).

6. Suppose the Environmental Protection Agency is considering new rules to limit pollution from PM_{10}, ozone, and sulfur dioxide. Before the rules take effect, they must conduct a detailed analysis to determine the costs and benefits of the rules.

   Over the study period, the average observed values of the pollutants were roughly (`so2median`, `pm10median`, `o3median`) = (−0.64, −0.15, −2.18). The EPA's proposed rules aim to reduce the averages to (−1, −0.5, −2.5).

   Using Model 2, produce a 95% bootstrap confidence interval for the mean **difference** in deaths between a day with the old average and a day with the new average pollution values, if that day's average

temperature were 65 degrees. That is, we want a confidence interval for

$$\mathbb{E}\left[r(-0.64, -0.15, -2.18, 65) - r(-1, -0.5, -2.5, 65)\right].$$

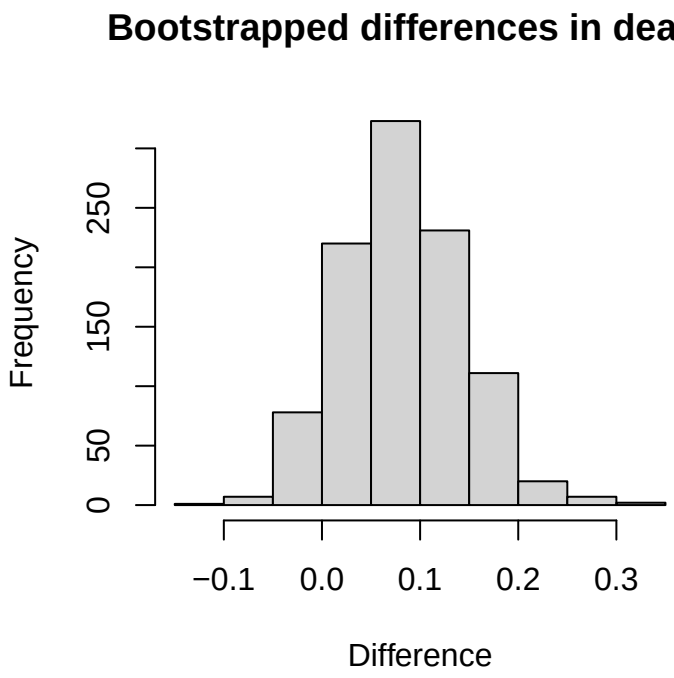
Clearly state any assumptions you make when constructing the confidence interval, and state how you decided which bootstrap method to use.

Solution: We will run 1,000 bootstrap iterations in which we make predictions and store the differences:

```
B <- 1000
diffs <- numeric(B)

newd1 <- data.frame(so2median = -0.64, pm10median = -0.15,
                    o3median = -2.18, tmpd = 65)
newd2 <- data.frame(so2median = -1, pm10median = -0.5,
                    o3median = -2.5, tmpd = 65)

for (i in 1:B) {
  rows <- sample(1:nrow(chicago), nrow(chicago), replace = TRUE)

  boot_data <- chicago[rows, ]

  model2_boot <- gam(death ~ s(pm10median, k = 4 + 1, fx = TRUE) +
                       s(o3median, k = 4 + 1, fx = TRUE) +
                       s(so2median, k = 4 + 1, fx = TRUE) +
                       s(tmpd, k = 4 + 1, fx = TRUE),
                     data = boot_data)

  diffs[i] <- predict(model2_boot, newdata = newd1) -
    predict(model2_boot, newdata = newd2)
}
```

Just heuristically, let's look at a histogram of those to see what to expect (this is not really necessary):

```
hist(diffs, xlab = "Difference", main = "Bootstrapped differences in deaths")
```



The pivotal confidence interval is:

```
realdiff <- as.vector(predict(model2, newdata = newd1) -
                        predict(model2, newdata = newd2))

2 * realdiff - quantile(diffs, probs = c(0.975, 0.025), names = FALSE)

## [1] -0.0606  0.1750
```

This confidence interval includes 0, suggesting there is not enough data available (or our model isn't good enough) to justify a claim that such a reduction in pollution is associated with a decrease in mean death rates.

In this case we chose to bootstrap by resampling cases, which requires few assumptions. Based on the results of the previous questions, we could have chosen to resample residuals as well, since apart from the few outliers, where was no sign the residuals have changing variance. Resampling residuals would have only required justifying that the residuals all have the same distribution.

Had we chosen a parametric bootstrap based on the normality of the residuals, we would have had to justify the assumption that the residuals are normally distributed. They look mostly okay apart from a number of high outliers, which may be influential on the results; to be safe, we'd probably use a different bootstrap, though it's possible the outliers would not matter too much.

7. Now let's consider prediction. Suppose that we want a model that predicts as accurately as possible, so instead of forcing the smoothing spline components of Model 2 to have 4 degrees of freedom, we choose the degrees of freedom to best fit the data.

   In `mgcv`, we can allow the `gam` function to choose the edf automatically. If we write `s(tmpd, k = 10 + 1)`, omitting `fx = TRUE`, the smoothing spline fit to `tmpd` will have **at most** 10 edf. Instead `gam` will

use generalized cross-validation to choose the exact value.

Fit a model where each term is a smoothing spline with at most 10 edf. Call this Model 3. Print the model summary, plot the partial response functions, and comment on the edf selected by `gam` for each component. Which variables are most linear and which are most nonlinear?

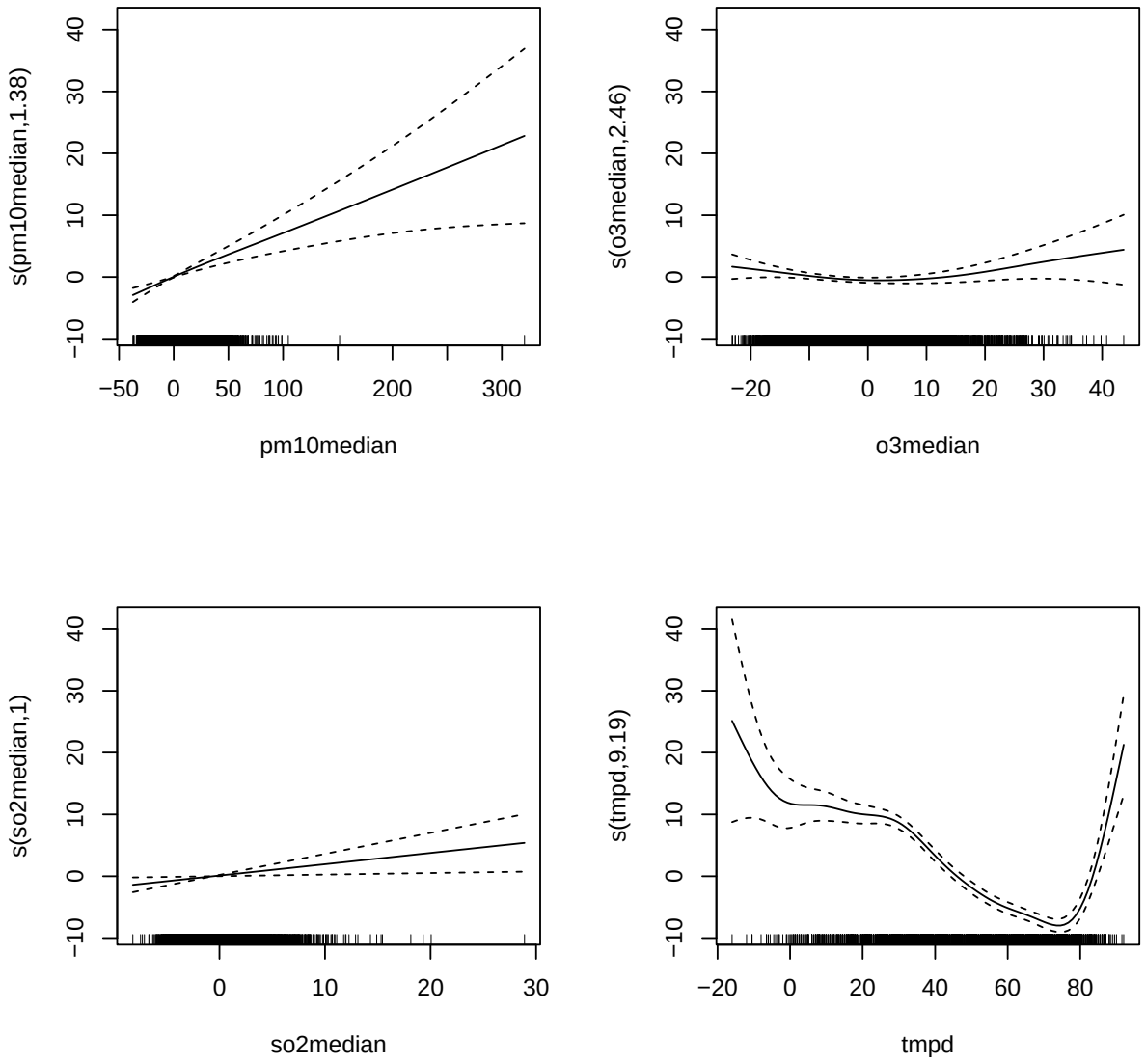
Solution: Fitting the model is easy enough:

```
model3 <- gam(death ~ s(pm10median, k = 10 + 1) +
                s(o3median, k = 10 + 1) +
                s(so2median, k = 10 + 1) +
                s(tmpd, k = 10 + 1),
              data = chicago)
summary(model3)

##
## Family: gaussian
## Link function: identity
##
## Formula:
## death ~ s(pm10median, k = 10 + 1) + s(o3median, k = 10 + 1) +
##     s(so2median, k = 10 + 1) + s(tmpd, k = 10 + 1)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  115.330      0.199     580   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##                 edf Ref.df     F p-value
## s(pm10median) 1.38   1.69 19.69 5.6e-07 ***
## s(o3median)   2.46   3.15  2.86   0.033 *
## s(so2median)  1.00   1.00  5.38   0.020 *
## s(tmpd)       9.19   9.80 81.84 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.184   Deviance explained = 18.6%
## GCV = 191.82  Scale est. = 191.22    n = 4841
```

Notice that `so2median` has an edf of 1.0, indicating that `gam` has decided to make that term linear. `tmpd` has a high edf over 9, suggesting the relationship with temperature is highly nonlinear.

We can see that in the partial response functions:

```
par(mfrow = c(2, 2))
plot(model3)
```



`pm10median` seems almost linear, `so2median` *is* linear, and `o3median` has a gentle curve. Only temperature shows a highly wiggly shape.

8. How accurately can one predict deaths using only temperature and the pollution variables? Use 10-fold cross-validation on Model 3 to estimate the generalization error. Report the square root of the estimate so it is on the same scale as deaths, and comment on the size of the error compared to the number of deaths on a typical day. Is this a practical way to predict exact death counts?

   Note: To avoid `NA` predictions for observations with missing predictors, subset the data to only rows with values for all your predictors.

Solution:

```
K <- 10
chicago_cv <- chicago[complete_rows, ]
folds <- sample(rep(1:K, length = nrow(chicago_cv)))

errors <- numeric(K)

for (k in 1:K) {
  train <- chicago_cv[folds != k, ]
  test <- chicago_cv[folds == k, ]

  model3_cv <- gam(death ~ s(pm10median, k = 10 + 1) +
                     s(o3median, k = 10 + 1) +
                     s(so2median, k = 10 + 1) +
                     s(tmpd, k = 10 + 1),
                   data = train)

  errors[k] <- sum((predict(model3_cv, newdata = test) - test$death)^2) / nrow(test)
}
sqrt(mean(errors))

## [1] 13.9
```

That error is perhaps lower than you might expect: about 14 deaths. But the average number of daily deaths is:

```
mean(chicago$death)

## [1] 115
```

And the standard deviation is:

```
sd(chicago$death)

## [1] 15.3
```

So predicting the mean would be only slightly worse than using Model 3, despite its fancy fit.