Examining model diagnostics

QERM 514 - Homework 3 Answer Key

17 April 2020

R Markdown file

You can find the R Markdown file used to create this answer key here.

Background

Section 7 of the U.S. Endangered Species Act (ESA) regulates situations in which a federal agency funds, permits, or otherwise has a "federal nexus" on any project that may influence a protected species. Federal agencies must seek a "consultation" on the project with either the U.S. Fish and Wildlife Service (USFWS) or the National Marine Fisheries Service (NMFS), depending on the species, and USFWS or NMFS must assure that any project does not cause "jeopardy" (a relatively high legal standard) for a protected species. A major conservation value of Section 7 consultation is the opportunity for USFWS and NMFS biologists to negotiate changes to projects that could minimize any negative impacts on species (or maximize any positive benefits).

The USFWS office in Lacey, Washington wanted to identify the characteristics of projects that would make them worthwhile for focused consultation time, with an emphasis on projects potentially impacting ESA-listed bull trout (*Salvelinus confluentus*). Experts developed assessments of the potential improvement(s) in a project that could be realized from negotiating changes to projects such as nearshore construction, culvert improvements, and riparian restoration. These assessment generated a unitless score of the potential value for 38 projects.

At this point the USFWS would like your assistance in evaluating a statistical model they hope to use for prioritizing project consultations. The accompanying data file `usfws_bull_trout.csv` contains 9 columns of information. They are

1. `score`: a project's potential value (numerical score on a scale of 0-15)
2. `stage`: 1 of 3 life history stage(s) occurring in the project area
    - adults (`A`)
    - juveniles/adults (`JA`)
    - eggs/juveniles (`EJ`)
3. `form`: 1 of 2 life history form(s) occurring in the project area
    - anadromous (`An`)
    - fluvial/anadromous (`FlAn`)
4. `cond`: 1 of 3 habitat conditions in the project area
    - pristine (`P`)
    - degraded (`D`)
    - highly degraded (`H`)

5. `risk`: 1 of 4 levels of extinction risk of the core population occurring in the project area
    - outside core area (`OC`)
    - low (`L`)
    - medium (`M`)
    - high (`H`)
6. `unit`: 1 of 4 habitat unit types in the project area
    - inside a core area (`IC`)
    - outside a core area in freshwater (`OF`)
    - marine (`M`)
    - other (`OT`)
7. `prog`: whether or not the set of detailed management guidelines for projects of that type have been established
    - `Yes`
    - `No`
8. `BMP`: whether or not established best management practices will be followed in the project
    - `Yes`
    - `No`
9. `degflex`: the degree of flexibility in project design, timing, and location
    - low (`L`)
    - medium (`M`)

As you work through the following problems, make sure to explain your thought process and show all of your **R** code, so Mark can give you partial credit, if necessary.

Problems

a) Fit a linear model to the dataset that includes all 8 predictor variables. What is the R^2 for the model? Does is seem like a promising model?

```
## get data
dat <- read.csv("usfws_bull_trout.csv")
## fit model wit all 8 predictors
fmod <- lm(score ~ stage + form + cond + risk + unit + prog + BMP + degflex, data = dat)
## summary
summary(fmod)
```

```
##
## Call:
## lm(formula = score ~ stage + form + cond + risk + unit + prog +
##     BMP + degflex, data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -4.8393 -0.4525  0.0465  0.6257  5.0838
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.27852    3.12978   1.048 0.305742
## stageEJ      6.54996    1.67547   3.909 0.000704 ***
## stageJA      5.66124    1.48850   3.803 0.000916 ***
## formFlAn    -0.63393    2.38959  -0.265 0.793151
## condH        0.62407    1.10026   0.567 0.576072
## condP       -0.96800    1.55225  -0.624 0.539018
## riskL        1.48544    2.70158   0.550 0.587728
## riskM        0.64192    1.79215   0.358 0.723471
## riskOC      -1.94297    4.11150  -0.473 0.640974
## unitM        2.29973    3.26551   0.704 0.488348
## unitOF       1.49315    2.84772   0.524 0.605065
## unitOT       0.36718    3.17158   0.116 0.908839
## progYes     -1.98726    1.05228  -1.889 0.071633 .
## BMPYes       0.06842    1.34844   0.051 0.959970
## degflexM     2.68433    0.91894   2.921 0.007685 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.152 on 23 degrees of freedom
## Multiple R-squared:  0.7549, Adjusted R-squared:  0.6058
## F-statistic: 5.061 on 14 and 23 DF,  p-value: 0.000314
```

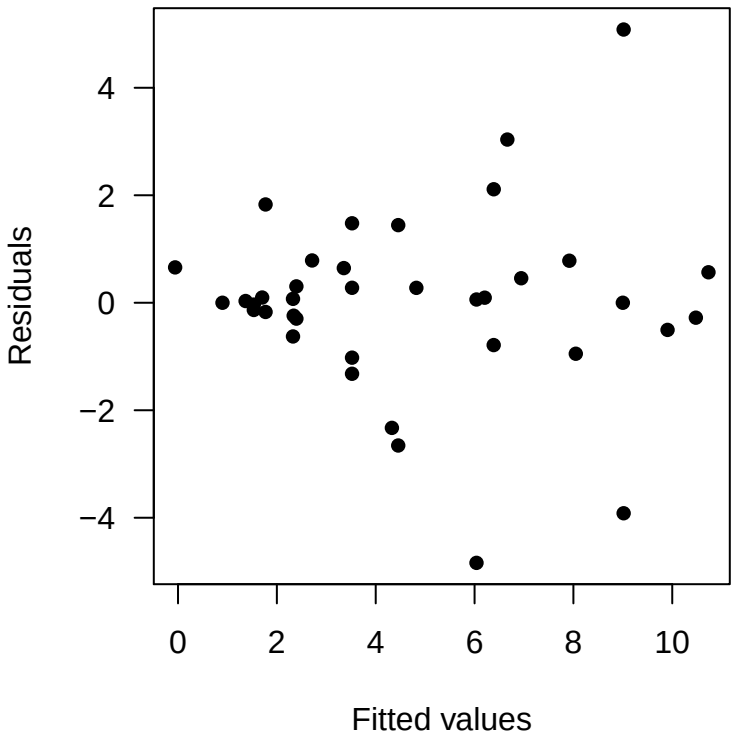
The R^2 for this model is 0.75 and the adjusted R^2 is 0.61. These R^2 values are pretty high, so this does indeed seem like a promising model.

---

b) Make a plot of the residuals against the model predictions. Name at least two things you should be looking for in a plot like this. What do you see?

Here is a plot of the residuals (e) versus the fitted values ($\hat{y}$).

```
## get residuals
res <- resid(fmod)
## get fitted values
yhat <- fitted(fmod)
## plot them
par(mai = c(0.9, 0.9, 0.6, 0.1),
    omi = c(0, 0, 0, 0))
plot(yhat, res, las = 1, pch = 16,
     xlab = "Fitted values", ylab = "Residuals")
```



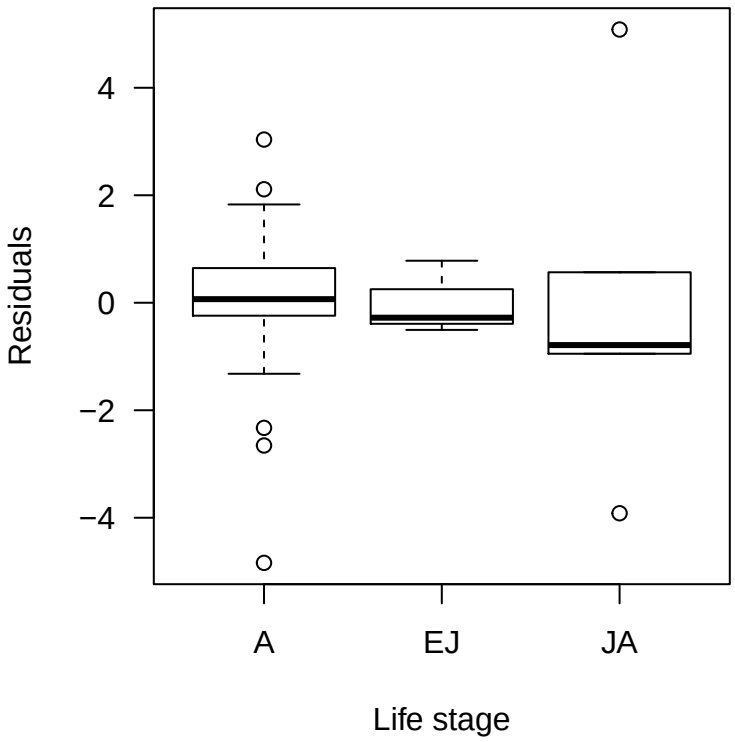
We should be looking for signs of non-constant variance in e (heteroscedasticity), possible nonlinear patterns in e (eg, skewness), and possible problems with the model design (outliers and leverage points). The sample size is somewhat small here, but there does appear to be some evidence of increasing variance in e with increasing $\hat{y}$.

---

c) Make a plot of the residuals against the predictor variable `stage`. Do you find this plot useful? Why or why not?

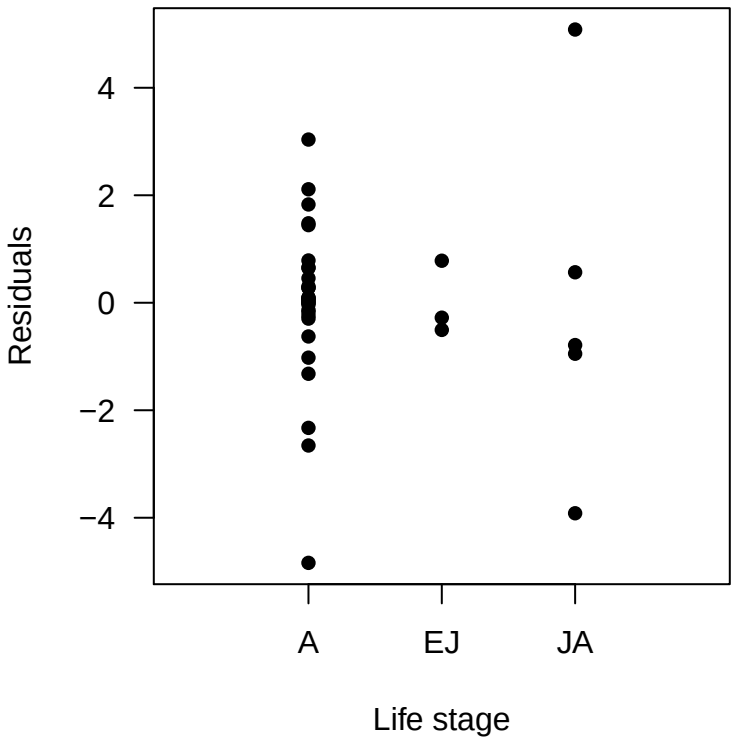
(**Note**: the default option with `plot()` generates a box-and-whisker plot, which is okay for our purposes here.)

```
## plot them
par(mai = c(0.9, 0.9, 0.6, 0.1),
    omi = c(0, 0, 0, 0))
## default box-and-whisker plot
plot(dat$stage, res, las = 1,
     xlab = "Life stage", ylab = "Residuals")
```



(**Note**: If you were really keen, you could replace this with `plot.default()`, which treats the categorical variable `stage` as a number.)

```
## plot them
par(mai = c(0.9, 0.9, 0.6, 0.1),
    omi = c(0, 0, 0, 0))
## plot with actual points
plot.default(dat$stage, res, las = 1, pch = 16, xlim = c(0,4),
     xaxt = "n", xlab = "Life stage", ylab = "Residuals")
axis(1, at = c(1, 2, 3), labels = c("A", "EJ", "JA"))
```

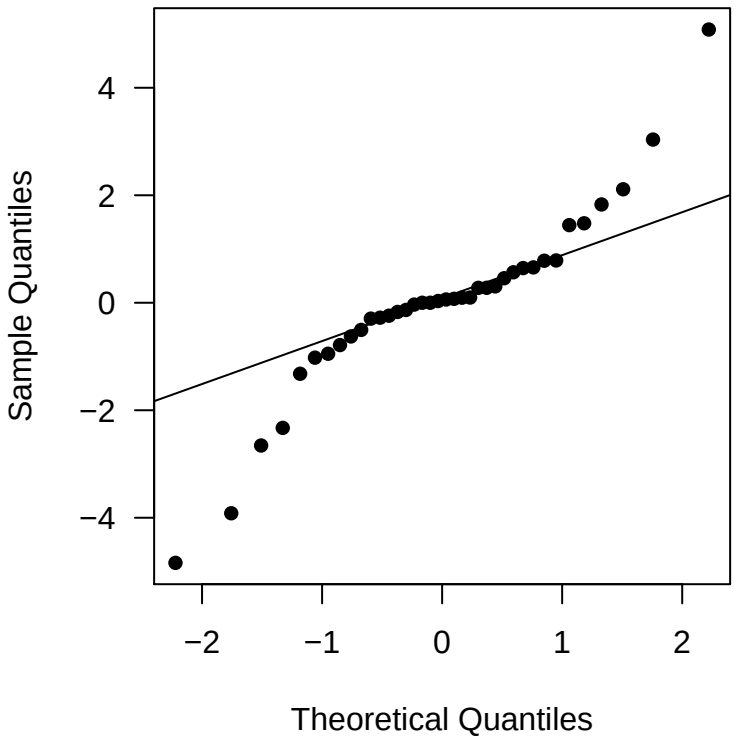


I do not find either of these plots particularly useful because of the limited range in `stage` against which to compare the residuals. We might try Levene's Test for homoscedasticity, but the unbalanced design leaves few residuals for the `EJ` and `JA` life stages.

---

d) Produce a Q-Q plot of the model residuals and include a Q-Q line. Describe what you would hope to see here. Do you?

```
par(mai = c(0.9, 0.9, 0.6, 0.1),
    omi = c(0, 0, 0, 0))
## Q-Q plot
qqnorm(res, las = 1, pch = 16, main = "")
## Q-Q line
qqline(res)
```



If the residuals are indeed normally distributed, then all of the points would fall along the Q-Q line here. In this case, however, there is some evidence of leptokurtosis (heavy-tailedness) because the sample quantiles are wider than expected at the tails of the distribution.

---

e) Would it make sense to plot e_t against e_{t+1} for this model? Explain why or why not.

We would typically use a plot of e_t against e_{t+1} to help diagnose whether the residuals were independently distributed (ie, temporally autocorrelated). In this case, however, the data were not collected sequentially in time, so we would have very little reason to believe that autocorrelation in the residuals is a potential problem.

---

f) Which projects have the 3 largest leverages? Briefly explain what this tells us.

We can use the `hatvalues()` function to get the leverages from our fitted model.

```
## get leverages
hv <- hatvalues(fmod)
## assign their names
names(hv) <- dat$ProjectName
## find the 3 largest leverages
rev(sort(hv))[1:3]
```

```
##    Vernon      Gail   Turner
## 1.0000000 1.0000000 0.7332097
```

There are two projects with $h = 1$ (Vernon , Gail) and two projects with $h = 0.73$ (Turner, Forgotten).

The leverage tells us how extreme a point is in predictor (X) space. We have seen many examples for regression models, but this concept is harder to think about with categorical predictors. Here these high leverages indicate that several predictors are rare in the dataset, or it may be that the particular combination of values rarely occur together across responses. If we look into this more carefully, we will find perfect collinearity in the dataset if either the `Vernon` or `Gail` projects are removed from the analysis. That is, without either of these points, one of the predictors is a combination of the other predictors, and hence one or more estimates would become unidentifiable.

---

g) What rule of thumb could you use to assess whether any leverages are particularly large? Under this rule of thumb, do you have any particularly large leverages? If yes, which projects?

In general, we look to see if $h > h_{\text{threshold}}$, where our threshold value for the leverages is given by

$$h_{\text{threshold}} = 2\frac{k}{n}$$

We can check this as follows.

```
## number of estimated params
k <- sum(hv) # = length(coef(fmod))
## sample size
n <- nrow(dat)
## threshold value for h
(ht <- 2 * k / n)
```

```
## [1] 0.7894737
```

```
## suspect leverages
hv[hv > ht]
```

```
##   Gail Vernon
##      1      1
```

It looks like 2 of the projects (Gail and Vernon) have leverages greater than our threshold value of ~0.79.

---

h) Calculate the studentized residuals to look for outliers. Remember to use a Bonferroni correction, and explain why you should use it. What did you find? Which project has the largest studentized residual?

Recall that we can use the leverages to scale the residuals so their variance is 1, such that

$$r_i = \frac{\hat{e}_i}{\hat{\sigma}\sqrt{1 - h_i}}$$

We can also make use of an alternative formulation given by

$$r_i = e_i \sqrt{\frac{n-k-1}{n-k-e_i^2}}$$

In **R**, we can use the `rstudent()` function to avoid doing these calculations by hand. We will use a Bonferroni-corrected α to because we are going to conduct 38 different null hypothesis tests, and we want to guard against rejecting an H_0 when it's true.

```
## get studentized residuals
(stud_e <- rstudent(fmod))
```

```
##           1           2           3           4           5           6
## -0.66292698  0.74389466 -0.51034964  0.13858004  0.65538526  0.24451238
##           7           8           9          10          11          12
##  0.33709078 -0.33133178 -2.44804907  3.52381466 -0.57663840 -1.38343234
##          13          14          15          16          17          18
##  0.05719023 -0.08694509  0.94769022 -0.34236729  0.03988007 -0.15667583
##          19          20          21          22          23          24
##  0.16175687 -0.01761444 -0.06953573         NaN  0.45193697 -0.65538526
##          25          26          27          28          29          30
##  0.01874359  0.49037416 -0.14579219         NaN  0.05085715 -2.94901708
##          31          32          33          34          35          36
##  0.03133582  1.93772708  0.34291078  1.30854014  0.29585331 -1.47454229
##          37          38
##  0.77559154 -0.24451238
```

```
## get sample size
n <- nrow(dat)
## Bonferroni correction: alpha / n
alpha <- 0.05 / n
## critical t value
df <- n - length(coef(fmod)) - 1
t_crit <- qt(1 - alpha/2, df)
## compare t_stud to t_crit
sum(stud_e > t_crit, na.rm = TRUE)
```

```
## [1] 0
```

This analysis indicates that none of the studentized residuals were greater than the critical value of 3.679, but 2 of the projects had studentized residuals equal to `NaN` (`Gail` and `Vernon`). Those are the same 2 projects with leverages equal to 1, which means dividing by 0 in the r_i calculations (ie, $r_i = \infty$).

---

i) Calculate Cook's Distances and produce a halfnormal plot of them. Label the 3 largest D_i in the plot with the project names. Are these the same sites as the top 3 projects you identified in (g)? Briefly explain why or why not.

Recall that Cook's Distance (D) is given by

$$D_i = e_i^2 \frac{1}{k} \left(\frac{h_i}{1 - h_i} \right).$$

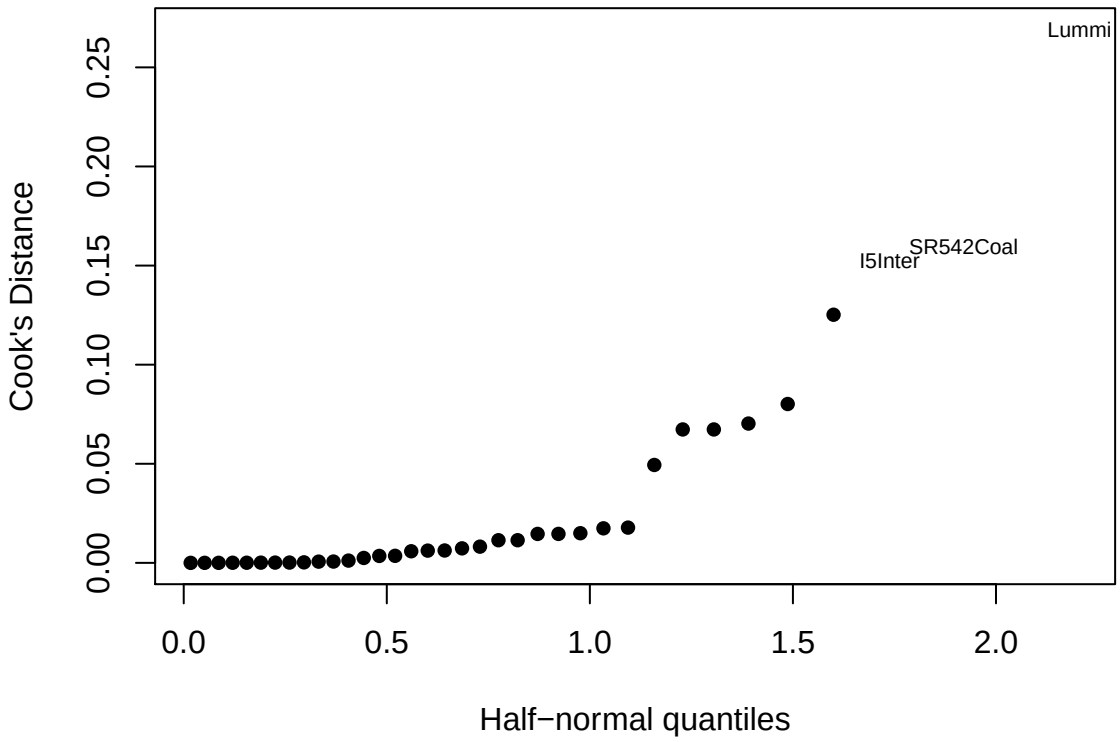
Note that we will encounter the same problem as part (h) with respect to the 2 projects with leverages equal to 1, in that we will be dividing by 0 and hence $D_i = \infty$ for those 2 projects.

We can calculate the D_i in **R** with `cooks.distance()` and use a slightly modified `halfnorm()` function from the **Faraway** package to plot them.

```
## modifed `halfnorm()` function from faraway
halfnorm <- function (x, nlab = 2, labs = NULL, ylab = "Sorted Data", ...) {
  x <- abs(x)
  labord <- order(x)
  x <- sort(x)
  i <- order(x)
  n <- length(x)
  ui <- qnorm((n + 1:n)/(2 * n + 1))
  plot(ui, x[i], xpd = NA,
       xlab = "Half-normal quantiles", ylab = ylab,
       ylim = c(0, max(x)), type = "n", ...)
  if(nlab < n) {
    points(ui[1:(n - nlab)], x[i][1:(n - nlab)], pch = 16)
  }
  if(is.null(labs)) {
    labs <- as.character(1:length(x))
  }
  text(ui[(n - nlab + 1):n], x[i][(n - nlab + 1):n],
       labs[labord][(n - nlab + 1):n], cex = 0.7)
}
## Cook's D
cook <- cooks.distance(fmod)
names(cook) <- dat$ProjectName
## 3 largest Cook's D
round(rev(sort(cook)), 3)[1:3]
```

```
##     Lummi SR542Coal   I5Inter
##     0.269     0.160     0.153
```

```
## half-normal plot
par(mai = c(0.9,0.9,0.6,0.1),
    omi = c(0, 0, 0, 0),
    cex = 1)
halfnorm(cook, nlab = 3,
         labs = dat$ProjectName, ylab = "Cook's Distance")
```



---

j) In a few sentences, summarize what you have learned about this analysis in terms of heteroscedasticity, normality, and influential observations.

There is some evidence of heteroscedasticity, but it is not very strong (ie, the variance of the residuals seems to increase with $\hat{y}$. There is good evidence that the errors are long-tailed, meaning we should probably not rely on distributional assumptions for null hypothesis tests or confidence intervals (but we could use bootstrap methods). There are two observations with (the maximum possible) leverage equal to 1 (`Gail` and `Vernon`), and there is one observation (`Lummi`) that is particularly influential. Thus, we may want to check whether there were any errors when entering those data points, or perhaps they might tell us something about the system that we didn't know.