Lab 10: Solutions

John Poulsen

November 03, 2018

Problems

Your assignment is to conduct two different analyses (see descriptions below).

For each regression, write a 1-page description of your analysis, results, and inference. Each write-up should include the following information:

- Null and alternative hypotheses of your tests
- Results of your statistical test, interpreting your test in 2-3 sentences that include the appropriate reporting of the statistics
- A description of how you checked the assumptions of your statistical test, including an interpretation of diagnostic figures
- Create plots of your regression equations to illustrate the results.

Problem 1

In this problem, analyze the `TreePlot` data on Sakai. The database contains information on approximately 70 tree plots, including: (1) plot biomass (`AGBH.Mg.ha`), (2) mean tree diameter (`mDBH.cm`), (3) mean height (`mH.m`), (4) mean wood density (`mWD.g.m3`), mean basal area (`mBA.cm2`), and (5) presence of tree falls in the plot (`Tree.Fall`). Note that `Tree.Fall} is a factor with three levels:`majeur`(major tree fall),`mineur`(minor tree fall), and`rien‘ (no tree fall).

The goal of the analysis is to determine the variables that influence the plot biomass and the direction and magnitude of their effect on biomass.

Solutions

This is a messy dataset, and there are multiple ways to analyze the data. There are, however, a few things that must be noted in the analysis:

1. Basal area, tree height, and dbh are highly correlated. Two of them must be removed from the analysis. This is not surprising, as BA functionally depends on the DBH (see equation) and DBH and tree height are interdependent.

2. Several plots are seemingly outliers and have a large influence on the analysis. For this assignment, it doesn't really matter what is done with these plots (leaving or removing them), as students do not know enough about the dataset to make an informed decision. But students should discuss the fact that these datapoints can drive the significance of both wood density and tree fall. The best strategy is to run the analysis with and without these datapoints to evaluate whether their exclusion would change the overall conclusion.

3. The rows of the database with `NA`'s need to be removed. This could be done to the entire dataset using `na.omit()` on the whole data frame. Alternatively, the `NA`'s will be automatically removed by the `lm()` function.

4. The students should show and interpret the diagnostics plot, demonstrating that the residuals look fine.

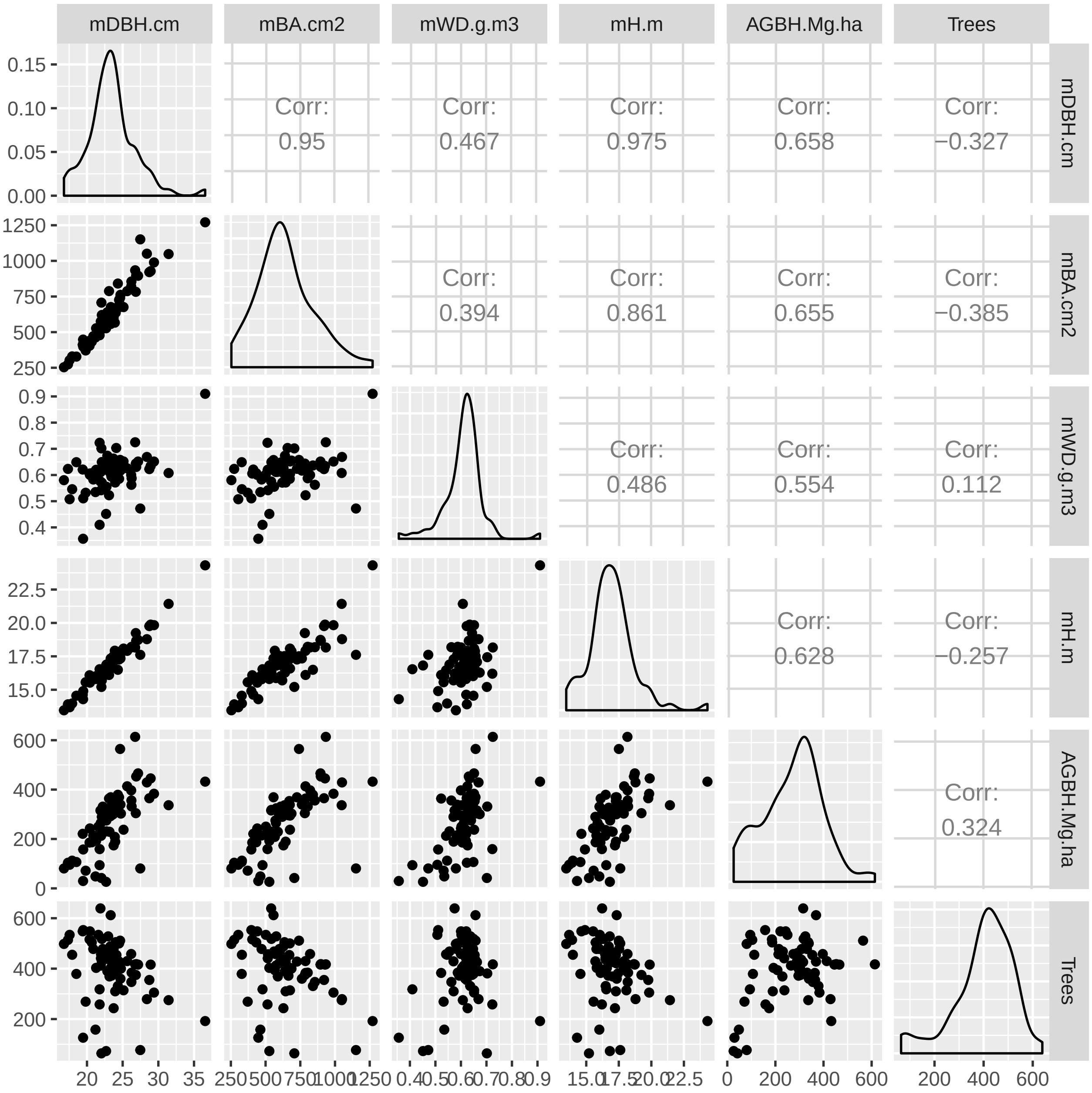
5. Students should interpret the results and provide a graph demonstrating the effects of the different IV's. For example,... "Trees, mean DBH, and disturbance all significantly affected the biomass of a forest plot, accounting for 78.2% of the variation ($F_{4,58} = 56.7, R^2 = 0.782, p < 0.001$)." Every additional tree increases aboveground biomass by 0.65 Mg, and every additional increase in mean tree diameter of the plot increases above ground biomass by 33.0 Mg. The presence of a treefall in the plot also influences aboveground biomass, with the presence of minor treefalls reducing aboveground biomass by 36.0 Mg compared to major treefalls ($t = 1.809, p = 0.076$). The effect of treefall seems counterintuitive, and needs further examination.

6. See R-code below for steps to the analysis, as well as the figures. I have shown a figure for Trees vs. AGB, and another figure could be done for mean DBH vs. AGB.

```
tdat  <-read.csv("TreePlots.csv", header=T, na.strings = ".")
 attach(tdat)

   require(HH)
    require(car)
   require(arm)
```

Examine the distribution of the variables and the relationships among variables.

```
require(ggplot2)
require(GGally)

ggpairs(tdat[,c(2:7)])
```



Evaluate whether to keep both basal area and DBH, or throw one out. First let's run a model with both variables, and then we will run a model with just DBH. The model could be made much more complex. For example, one could include an interaction between the number of trees and tree height or size. But for this assignment working with just the main effects is fine.

```
lm0 <- lm(AGBH.Mg.ha ~ Trees + mDBH.cm + mBA.cm2 + mH.m)
summary(lm0)

 lm1 <- lm(AGBH.Mg.ha ~ Trees + mDBH.cm + mBA.cm2)
 summary(lm1)

  lm2 <- lm(AGBH.Mg.ha ~ Trees + mDBH.cm + mH.m)
  summary(lm2)
 lm3 <- lm(AGBH.Mg.ha ~ Trees + mBA.cm2 + mH.m)
 summary(lm3)

lm4 <- lm(AGBH.Mg.ha ~ Trees + mDBH.cm)
summary(lm4)
```

The significance of all three variables changes as one or the other is excluded, and the effect sizes are all over the place, signaling a problem of multicollinearity. The variation inflation factor (VIF) shows that the variances are very inflated and correlated, until only one of the three variables is left.

```
vif(lm0)
```

```
##      Trees    mDBH.cm    mBA.cm2       mH.m
##   1.233318 489.313986  91.042334 186.496450
```

```
 vif(lm1)
```

```
##     Trees   mDBH.cm   mBA.cm2
##  1.195781 10.422631 10.929458
```

```
  vif(lm2)
```

```
##     Trees   mDBH.cm      mH.m
##  1.225901 23.408028 22.388542
```

```
 vif(lm3)
```

```
##    Trees  mBA.cm2     mH.m
## 1.204363 4.355325 3.972467
```

```
vif(lm4)
```

```
##    Trees  mDBH.cm
## 1.119615 1.119615
```

Now, the model we are really after, which doesn't include high multicollinearity.

```
lm3 <- lm(AGBH.Mg.ha ~ mWD.g.m3 + Trees + mDBH.cm +
          factor(Tree.Fall), na.action = na.exclude)
summary(lm3)
```

```
##
## Call:
## lm(formula = AGBH.Mg.ha ~ mWD.g.m3 + Trees + mDBH.cm + factor(Tree.Fall),
##     na.action = na.exclude)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -138.28  -35.54  -10.30   26.45  193.25
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)             -766.29530   79.74083  -9.610 1.58e-13 ***
## mWD.g.m3                 135.40276  132.58321   1.021    0.311
## Trees                      0.62262    0.07215   8.630 6.24e-12 ***
## mDBH.cm                   31.02501    3.18518   9.740 9.77e-14 ***
## factor(Tree.Fall)mineur  -40.38613   20.35002  -1.985    0.052 .
## factor(Tree.Fall)rien     -1.16408   22.11767  -0.053    0.958
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 59.96 on 57 degrees of freedom
##   (7 observations deleted due to missingness)
## Multiple R-squared:    0.8,  Adjusted R-squared:  0.7824
## F-statistic:  45.6 on 5 and 57 DF,  p-value: < 2.2e-16
```

The `rownames()` function allows us to reset the row numbers so that we can take out rows later on.

```
tdat1<-na.omit(tdat)
 rownames(tdat1) = NULL
```

Now let’s check which model is the best, trying to find the most parsimonious model for biomass.

```
lm4 <- lm(AGBH.Mg.ha ~ mWD.g.m3 + Trees + mDBH.cm + factor(Tree.Fall), data=tdat1)

 lm5 <- lm(AGBH.Mg.ha ~ mWD.g.m3 + Trees + mDBH.cm, data=tdat1)

  lm6 <- lm(AGBH.Mg.ha ~ Trees + mDBH.cm + factor(Tree.Fall), data=tdat1)

   lm7 <- lm(AGBH.Mg.ha ~ mWD.g.m3 + mDBH.cm + factor(Tree.Fall), data=tdat1)

 anova(lm4, lm5)  # Can't take out Tree.Fall!
```

```
## Analysis of Variance Table
##
## Model 1: AGBH.Mg.ha ~ mWD.g.m3 + Trees + mDBH.cm + factor(Tree.Fall)
## Model 2: AGBH.Mg.ha ~ mWD.g.m3 + Trees + mDBH.cm
##   Res.Df    RSS Df Sum of Sq      F  Pr(>F)
## 1     57 204947
## 2     59 228658 -2    -23711 3.2972 0.04415 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
 anova(lm4, lm6)  # Taking out mWD.g.m3 doesn't significantly change the model
```

```
## Analysis of Variance Table
##
## Model 1: AGBH.Mg.ha ~ mWD.g.m3 + Trees + mDBH.cm + factor(Tree.Fall)
## Model 2: AGBH.Mg.ha ~ Trees + mDBH.cm + factor(Tree.Fall)
##   Res.Df    RSS Df Sum of Sq     F Pr(>F)
## 1     57 204947
## 2     58 208698 -1   -3750.1 1.043 0.3114
```

```
 anova(lm4, lm7)  # Expected that,... can’t take out Trees!
```

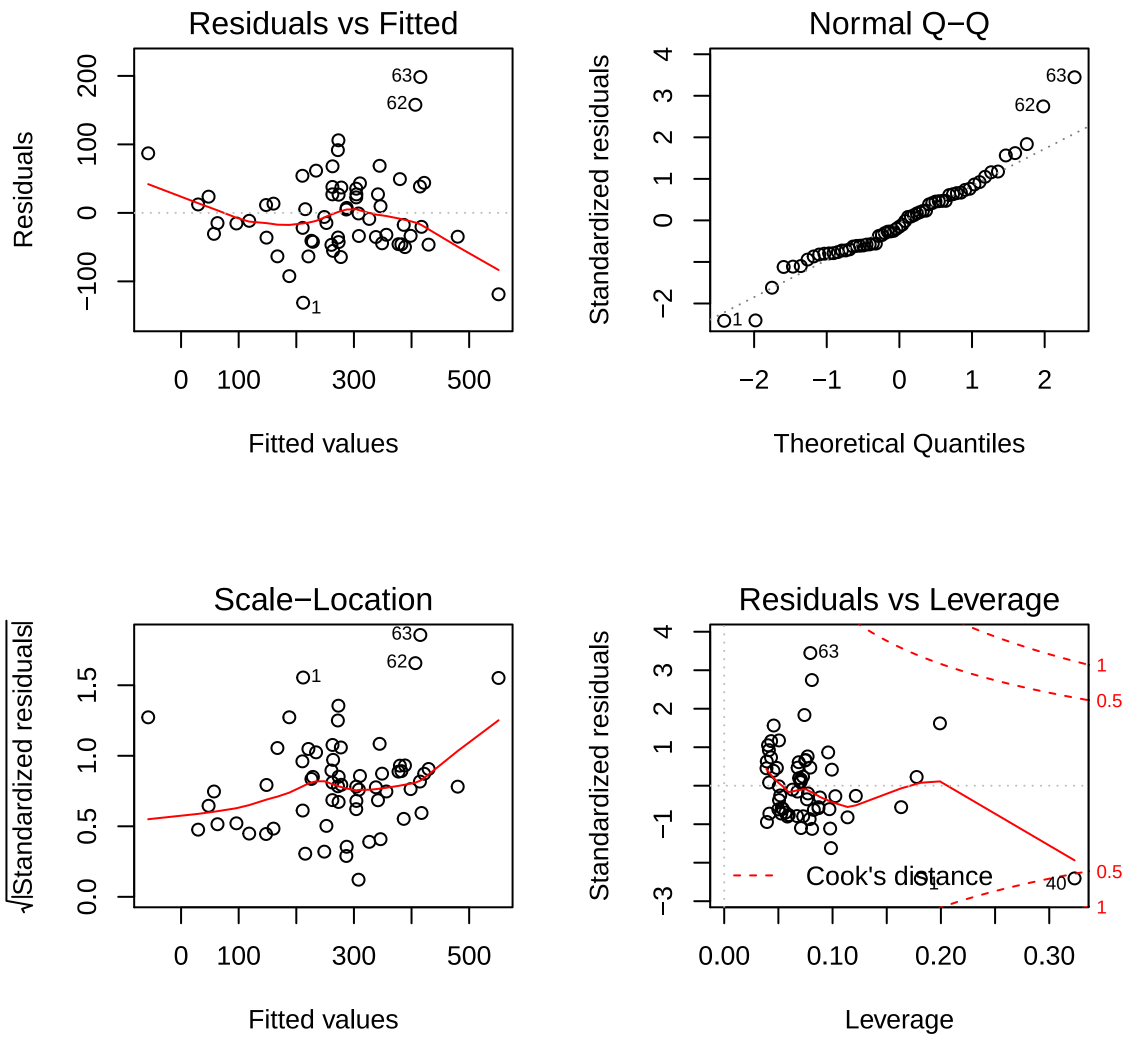
```
## Analysis of Variance Table
##
## Model 1: AGBH.Mg.ha ~ mWD.g.m3 + Trees + mDBH.cm + factor(Tree.Fall)
## Model 2: AGBH.Mg.ha ~ mWD.g.m3 + mDBH.cm + factor(Tree.Fall)
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1     57 204947
## 2     58 472725 -1   -267777 74.474 6.245e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We could also use the `StepAIC()` function to reduce the maximal model. (Students may not be familiar with this function, which is preferable as the idea is to learn how to reduce a model through other means and not have R just do it for you.)

```
par(mfrow=c(2,2))
 step.mod <- stepAIC(lm4)
```

```
## Start:  AIC=521.5
## AGBH.Mg.ha ~ mWD.g.m3 + Trees + mDBH.cm + factor(Tree.Fall)
##
##                     Df Sum of Sq    RSS    AIC
## - mWD.g.m3           1      3750 208698 520.65
## <none>                           204947 521.50
## - factor(Tree.Fall)  2     23711 228658 524.40
## - Trees              1    267777 472725 572.16
## - mDBH.cm            1    341132 546080 581.25
##
## Step:  AIC=520.65
## AGBH.Mg.ha ~ Trees + mDBH.cm + factor(Tree.Fall)
##
##                     Df Sum of Sq    RSS    AIC
## <none>                           208698 520.65
## - factor(Tree.Fall)  2     23967 232664 523.50
## - Trees              1    337476 546173 579.26
## - mDBH.cm            1    623913 832611 605.82
```

```
  plot(step.mod)
```

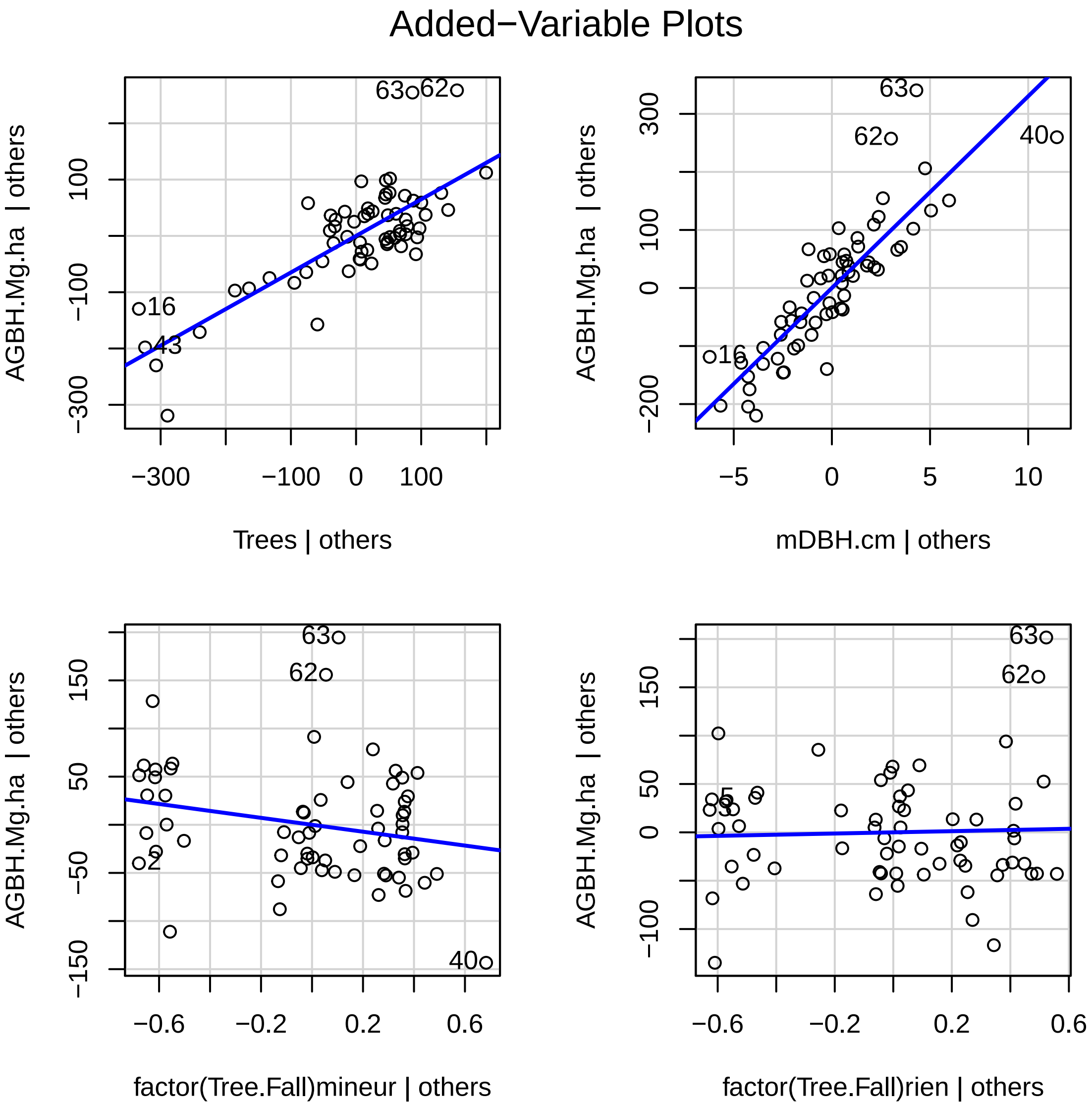


```
formula(step.mod)
```

```
## AGBH.Mg.ha ~ Trees + mDBH.cm + factor(Tree.Fall)
```

Using `stepAIC` indicates that the final model is $AGBH.Mg.ha \sim Trees + mDBH.cm + factor(Tree.Fall)$.

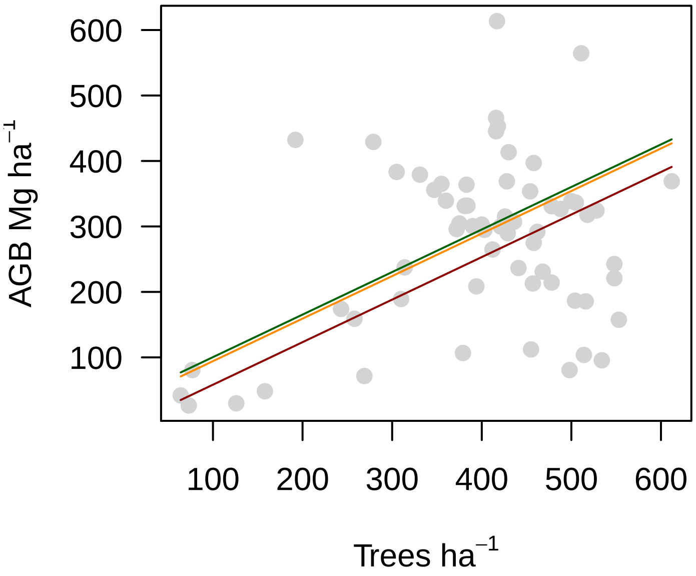
```
avPlots(step.mod)
```



The diagnostics plots and add-variable plots look fine, although there are a couple of datapoints with high leverage or relatively high residuals.

```
par(mfrow=c(1,1))
 coefs <- coefficients(step.mod)

 with(tdat1, plot(Trees, AGBH.Mg.ha, las = 1, xlab = expression(paste("Trees ha" ^-1)),
                 ylab = expression(paste("AGB Mg ha"^-1)),
                 pch = 19, col = "lightgrey"))
  x <- seq(min(tdat1$Trees), max(tdat1$Trees))

## For major treefalls
curve(coefs[1] + coefs[2]*x + coefs[3]*mean(tdat1$mDBH.cm) + coefs[4]*0 + coefs[5]*0,
      add=T, col="darkorange")

## For no treefalls
curve(coefs[1] + coefs[2]*x + coefs[3]*mean(tdat1$mDBH.cm) + coefs[4]*0 + coefs[5]*1,
      add=T, col="darkgreen")
```

```
## For minor treefalls
curve(coefs[1] + coefs[2]*x + coefs[3]*mean(tdat1$mDBH.cm) + coefs[4]*1 + coefs[5]*0,
      add=T, col="darkred")
```



Problem 2

The dataset `ozone.data.csv` consists of data on air quality (ozone concentration) and several weather variables, including temperature, solar radiation, and wind speed. Model the relationship of the weather variables on ozone concentration, making sure to consider interactions between the independent variables. (Do not worry about units in this example.) Once you have determined the best model to predict ozone concentration, write a one-page report that includes the following:

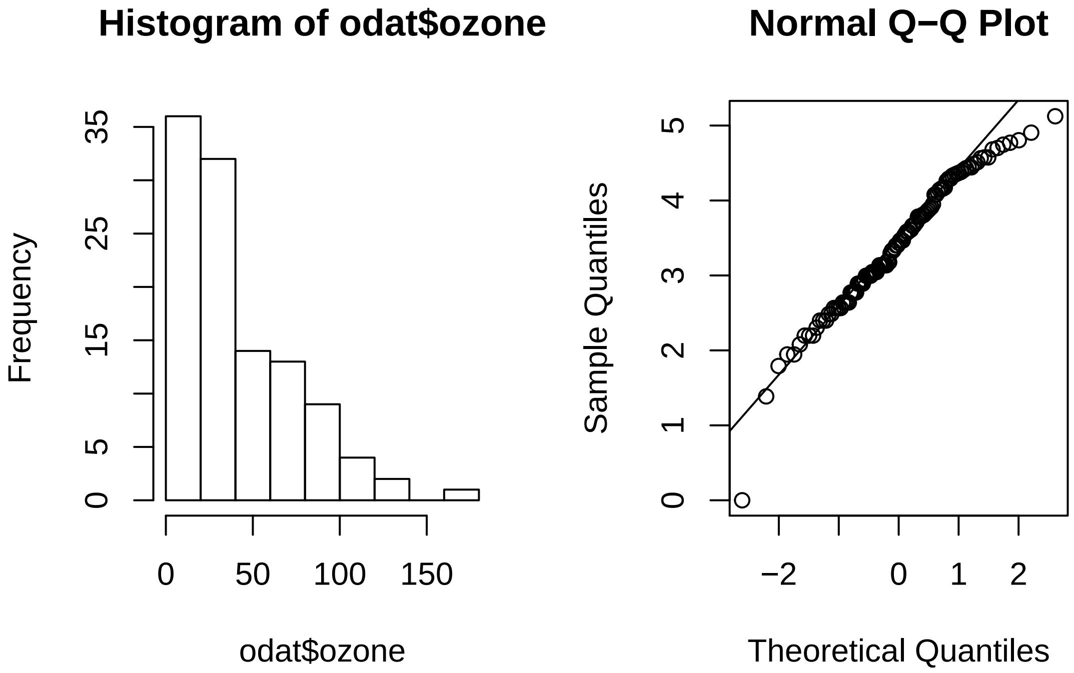
- Null and alternative hypotheses of your model
- Results of your statistical test, interpreting the fit of the model in 2-3 sentences that include the appropriate reporting of the statistics
- An interpretation of the regression model coefficients (i.e., what do each of the main effect(s) and interaction effect(s) mean
- A description of how you checked the assumptions of your statistical test
- Graphs that depict (i) the relationship between the IV and DV variable where a variable is only included as a main effect, (ii) the relationship between the interacting independent variables with the DV (i.e., plug in 2-3 interesting values for one of the independent variables and plot the other independent variable against the dependent variable).

Solutions

This problem is tricky because includes an interaction with two continuous variables. The students should, at a minimum, do the following:

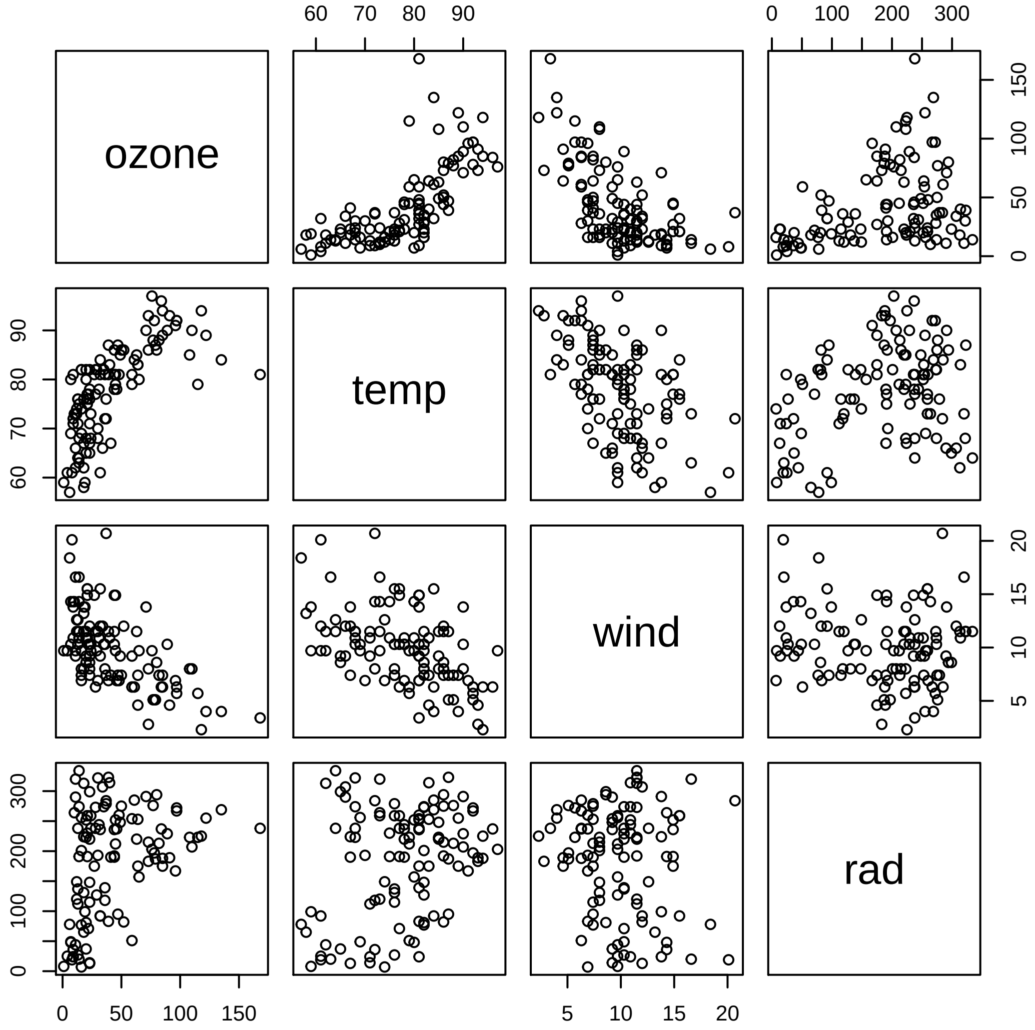
- Evalute the data, and transform *ozone*
- Find the best model, which should include the interaction between *temperature* and *radiation*
- Examine the diagnostics
- Explain what the interaction means, and hopefully make a figure of the interaction.

```
odat <- read.csv("ozone.data.csv", header=T, stringsAsFactors=F)

odat$rad<-as.numeric(odat$rad)

## Check out the data
par(mfrow=c(1,2))
hist(odat$ozone)

qqnorm(log(odat$ozone))
qqline(log(odat$ozone))
```

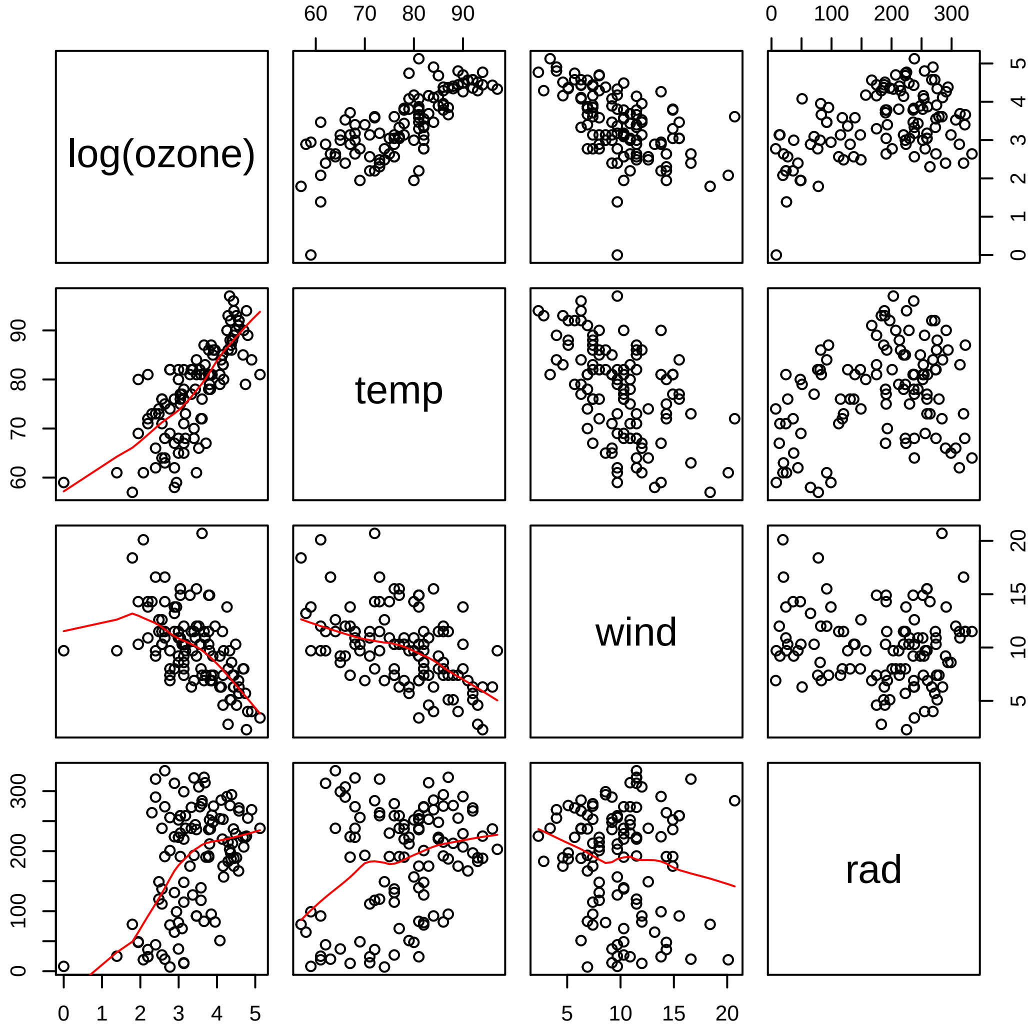


These plots demonstrate that `ozone` is not normally distributed, and there is curvature (lack of linear relationship) between `ozone` and the other variables.

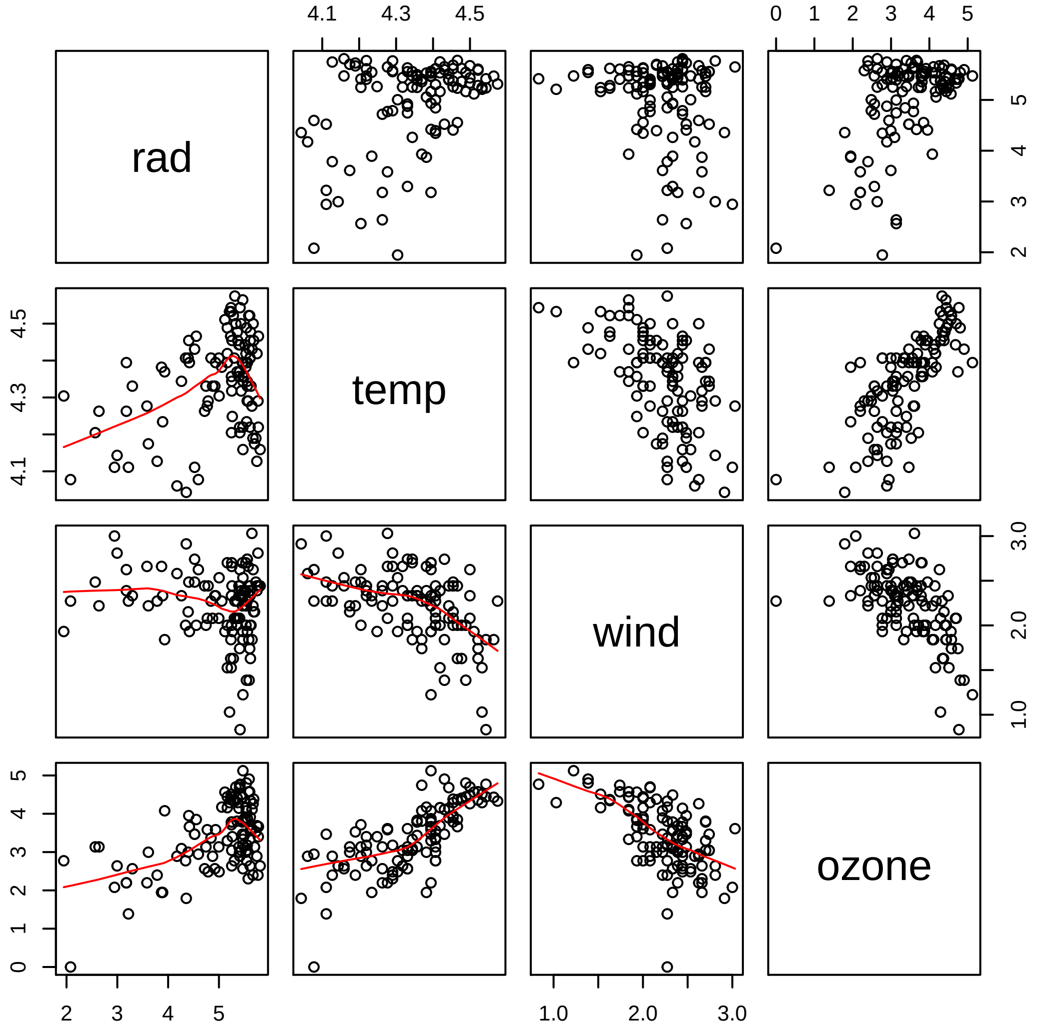
```
with(odat, pairs(ozone~temp+wind+rad))
```



```
with(odat, pairs(log(ozone)~temp+wind+rad, lower.panel=panel.smooth))
```



```r
pairs(log(odat), lower.panel=panel.smooth)
```



```r
cor(log(odat))
```

```
##              rad       temp       wind      ozone
## rad    1.0000000  0.3867591 -0.1760234  0.5383271
## temp   0.3867591  1.0000000 -0.5077183  0.7378570
## wind  -0.1760234 -0.5077183  1.0000000 -0.5929688
## ozone  0.5383271  0.7378570 -0.5929688  1.0000000
```

Reduce the model to a point where all remaining predictors are significant, eliminating one predictor at a time. Here I used `update()`, which students could have used above.

The syntax of update is a little tricky. Inside the new formula, a period means “same”. Update the model using the same response variables and predictor variables minus the specified variable or combination of variables: `temp:wind:rad`.

```r
omod1 <- with(odat, lm(log(ozone)~temp*wind*rad))
 omod2 <- update(omod1, ~.-temp:wind:rad)
  omod3 <- update(omod2, ~.-wind:rad)
   omod4 <- update(omod3, ~.-temp:rad)
  omod5 <- with(odat, lm(log(ozone)~temp+rad+wind))
 omod6 <- with(odat, lm(log(ozone)~temp+rad))
anova(omod1, omod2)
 anova(omod2, omod3)
  anova(omod3, omod4)
 anova(omod4, omod5)
anova(omod5, omod6)
```

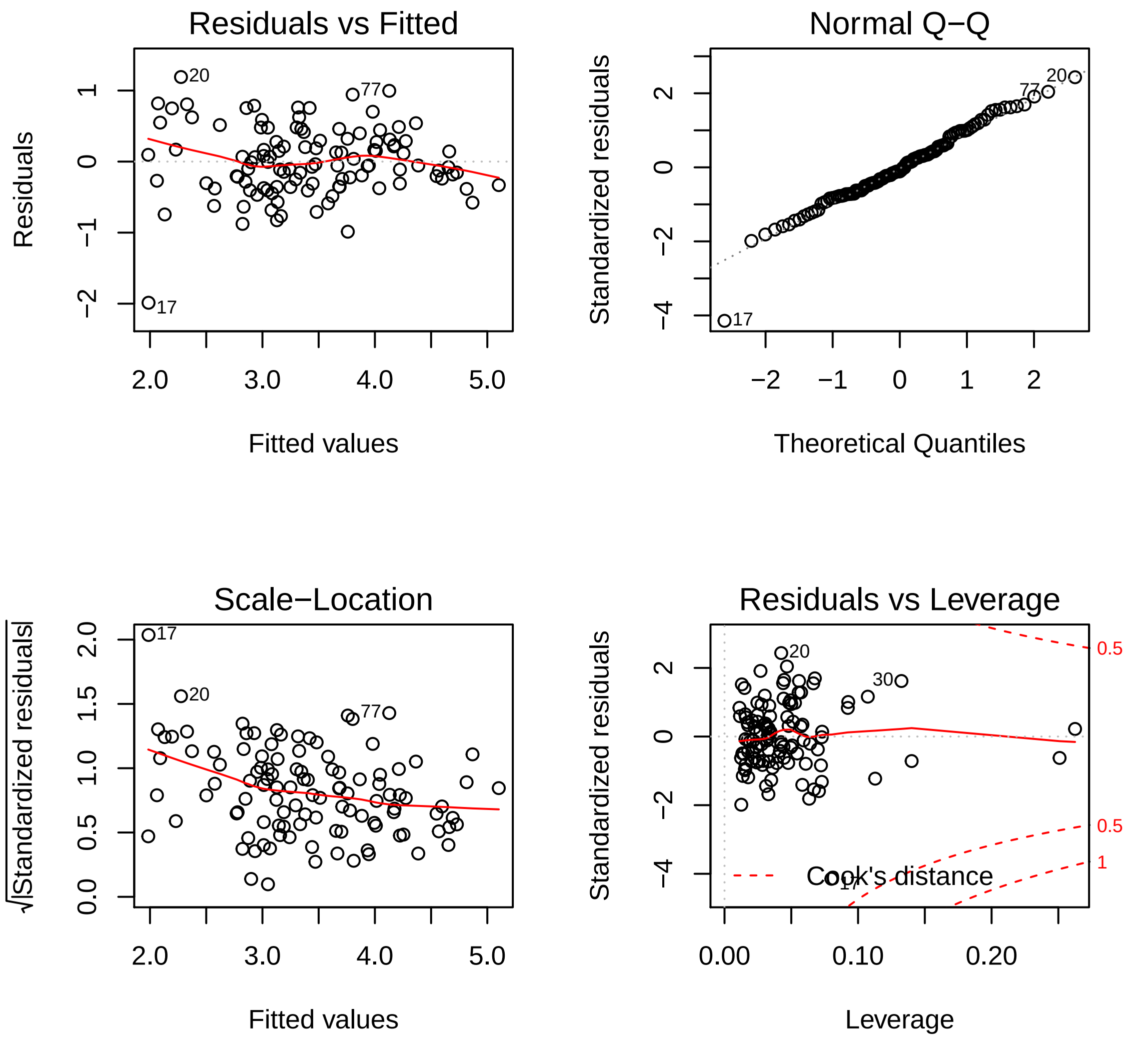
We can automate this using `step()` – but probably shouldn't! Don't just let R pick the model. Losing the `temp:wind` interaction made the model worse. We need to keep the main effects for the interaction, but let's see if `rad` is necessary.

```
omod7 <- with(odat, lm(log(ozone)~temp+wind + temp:wind))
 anova(omod4, omod7)
```

```
## Analysis of Variance Table
##
## Model 1: log(ozone) ~ temp + wind + rad + temp:wind
## Model 2: log(ozone) ~ temp + wind + temp:wind
##   Res.Df    RSS Df Sum of Sq      F    Pr(>F)
## 1    106 26.473
## 2    107 32.063 -1   -5.5902 22.383 6.926e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Nope, can't lose `rad`, so `omod4` is the final model.

```
par(mfrow=c(2,2))
 plot(omod4)
```

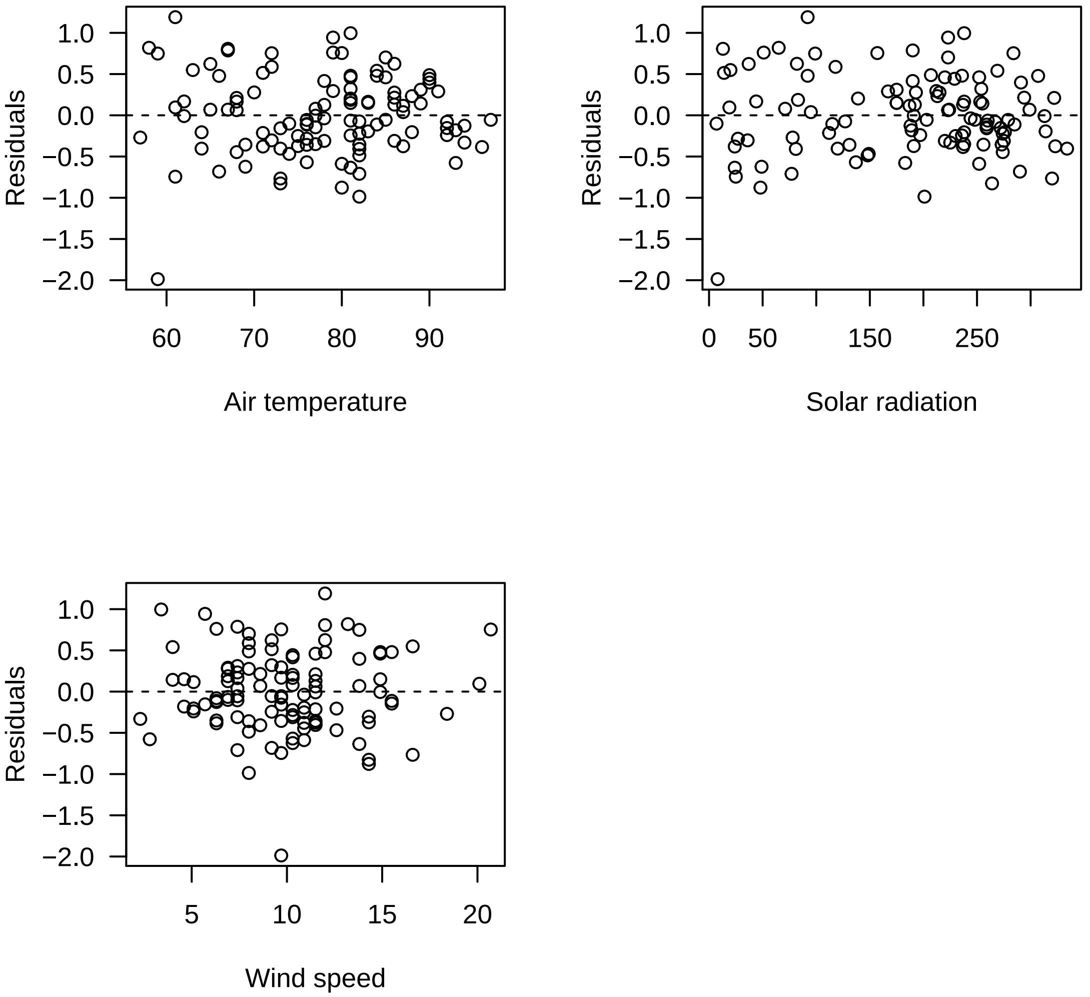


Check the fit by examining the residuals plotted against the independent variables.

```
par(mfrow=c(2,2))
 plot(odat$temp, resid(omod4), xlab="Air temperature", ylab="Residuals",las=1)
  abline(h=0, lty=2)

plot(odat$rad, resid(omod4), xlab="Solar radiation", ylab="Residuals", las=1)
 abline(h=0, lty=2)

plot(odat$wind, resid(omod4), xlab="Wind speed", ylab="Residuals", las=1)
 abline(h=0, lty=2)

summary(omod4)
```

```
##
## Call:
## lm(formula = log(ozone) ~ temp + wind + rad + temp:wind)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.98696 -0.32076 -0.05428  0.30238  1.19016
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.5845520  1.1936395  -2.165   0.0326 *
## temp         0.0782341  0.0145784   5.366 4.76e-07 ***
## wind         0.1661571  0.1052917   1.578   0.1175
## rad          0.0025939  0.0005483   4.731 6.93e-06 ***
## temp:wind   -0.0029299  0.0013399  -2.187   0.0310 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4997 on 106 degrees of freedom
## Multiple R-squared:  0.679,  Adjusted R-squared:  0.6669
## F-statistic: 56.05 on 4 and 106 DF,  p-value: < 2.2e-16
```



What do each of these coefficients mean?

- The intercept represents the predicted `log(ozone)` for the situation where solar radiation, temperature and wind are all 0 – not very likely or helpful.
- The coefficient of `rad` is the change in `log(ozone)` with each addition unit of radiation.
- The coefficients for the main effects of `wind` and `temp` reflect conditional relationships. For example,

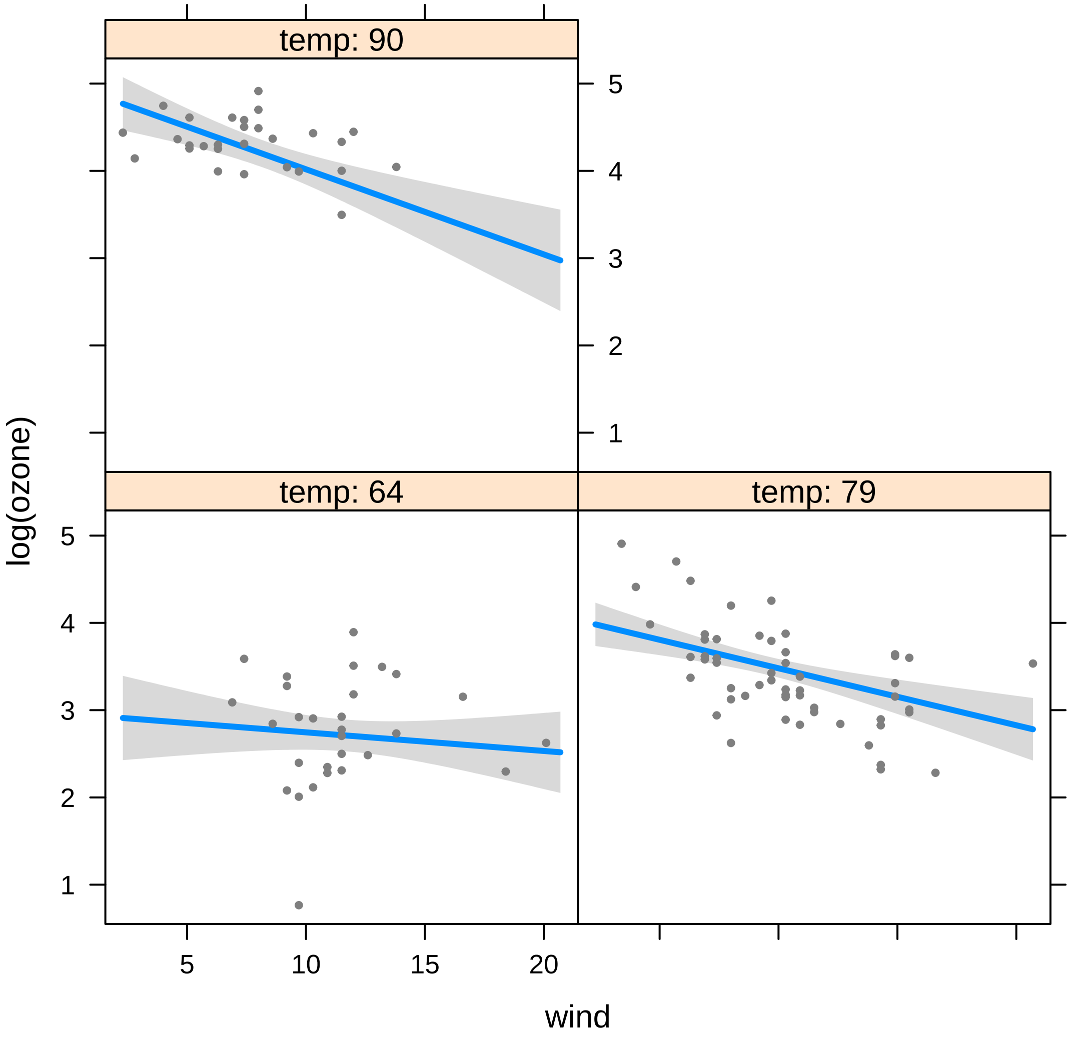
`temp` is the effect of a 1-unit change in temperature (X_1) on `log(ozone)` when (X_2) is 0. Similarly, wind is the effect of X_2 (wind) on Y (log(ozone)) when X_1 (temperature) = 0.

$$Y = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 (X_1 \times X_2) + \varepsilon$$

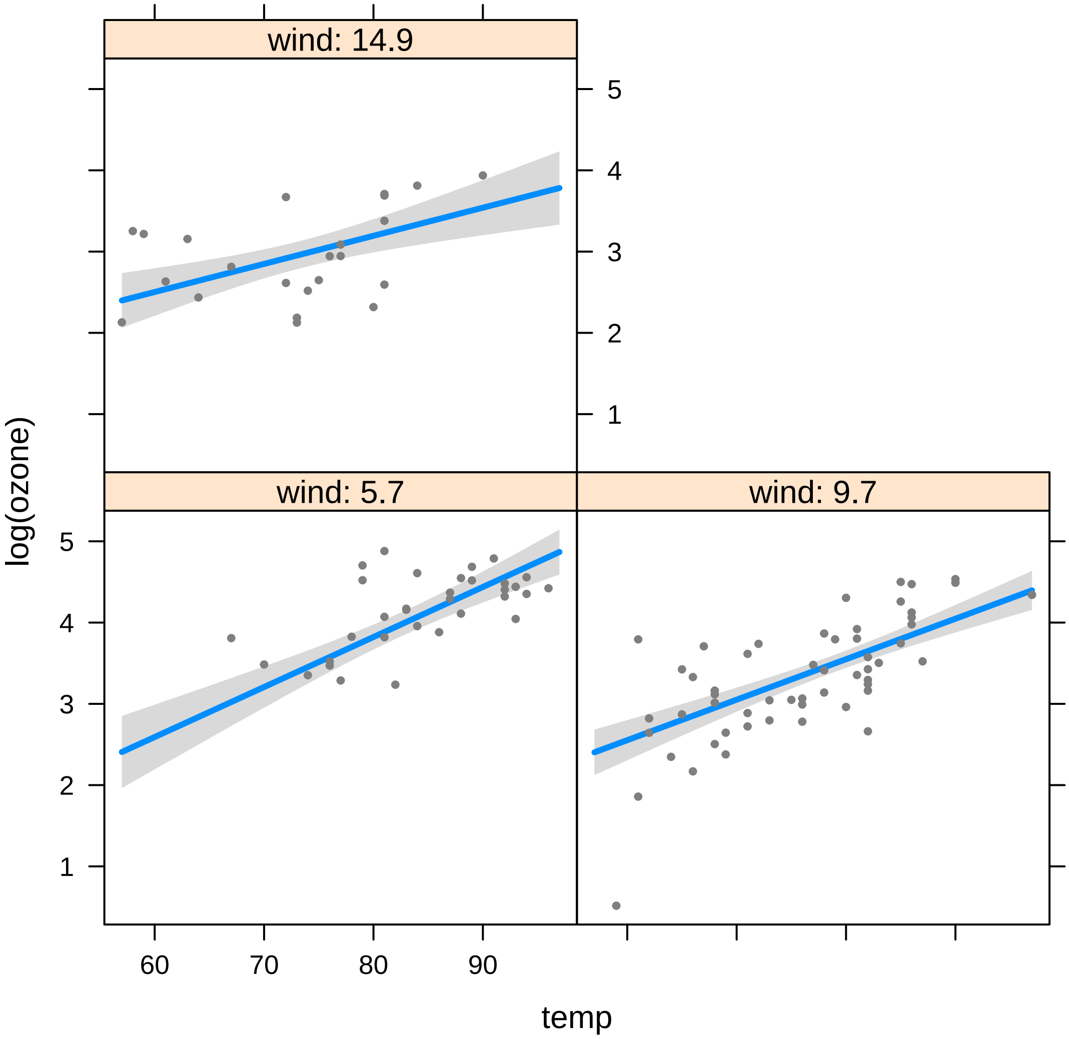
The coefficient on the interaction term represents the difference in the slope for wind at different levels of temperature. When $X_2 = 3$, a one unit increase in X_1 will produce a $(\beta_1 + 3 \times \beta_2)$ unit increase in Y. So the effect of X_1 on Y depends on X_2.

We can look at what the interaction means using `visreg`.

```r
require(visreg)

visreg(omod4, "wind", by = "temp")
```

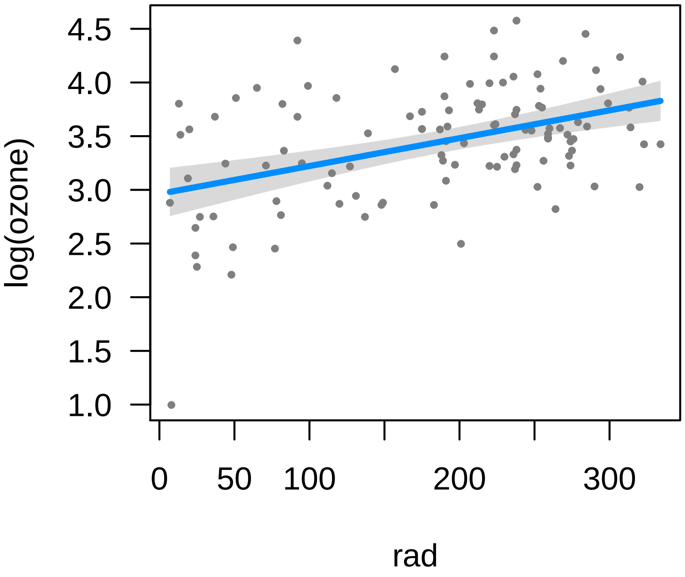


```r
visreg(omod4, "temp", by = "wind")
```



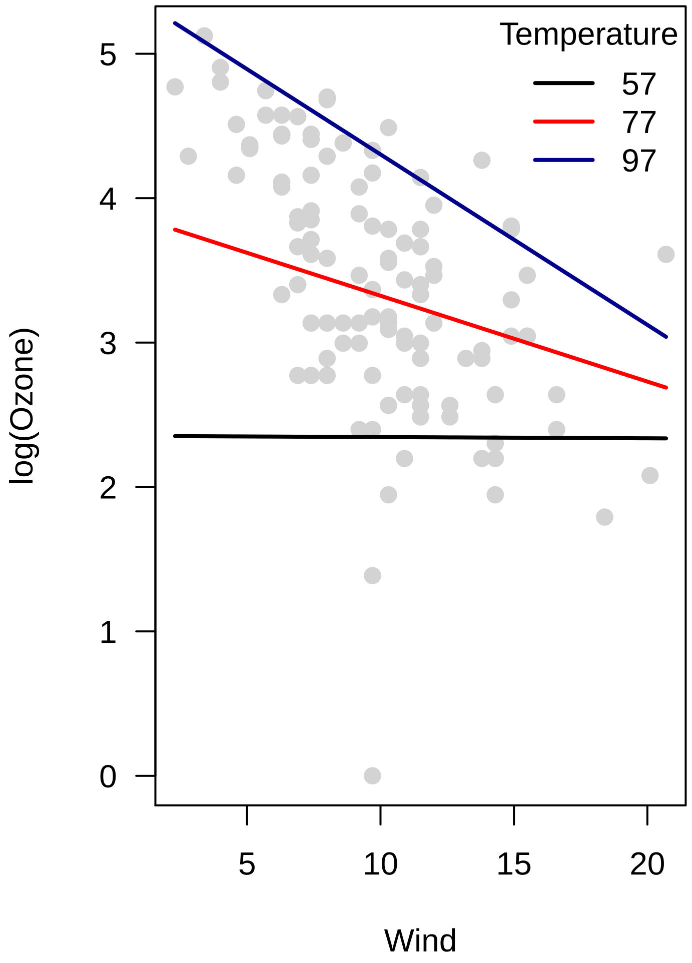
```r
visreg(omod4, "rad")
```

```
## Conditions used in construction of plot
## temp: 79
## wind: 9.7
```

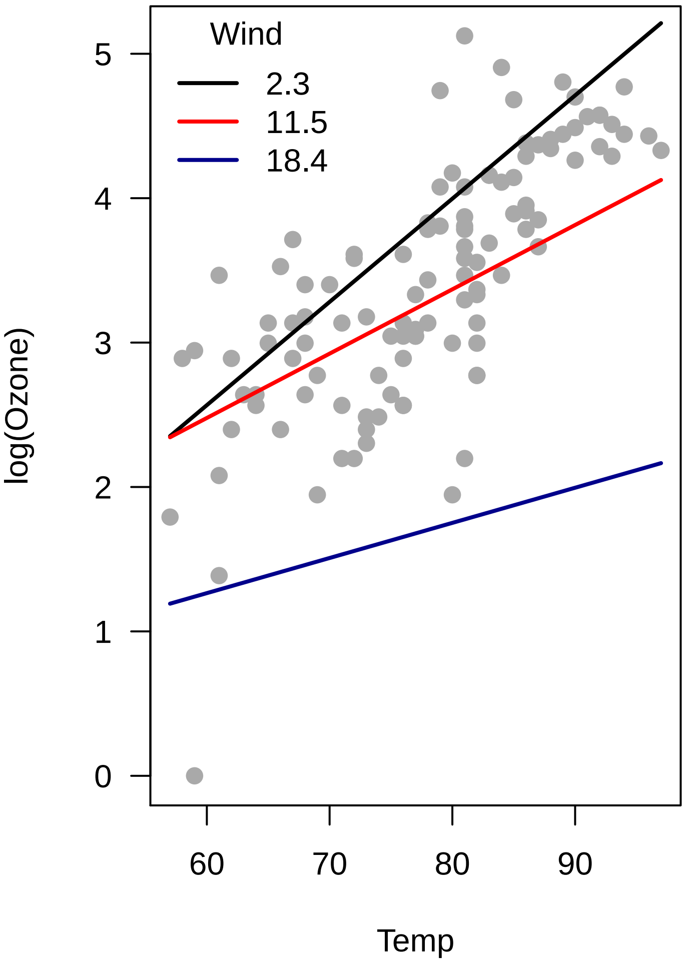


Or, better yet, we can plot it ourselves. Although there are several ways to examine the effect of interactions, one of the most intuitive is to plug in interesting values of one of the interacting IV's and then plot the other IV against the DV. This will give us an idea of how the effect of wind differs by temperature (or vice versa, below).

```
at.wind <- with(odat, seq(min(wind), max(wind), 0.5))
at.temp <- with(odat, seq(min(temp), max(temp), 0.5))

with(odat, plot(wind, log(ozone), las = 1, pch = 16, cex = 1.2,
                col = "lightgrey", ylab = "log(Ozone)", xlab = "Wind"))

t1 <- 57
t2 <- 77
t3 <- 97

x <- at.wind

curve(coef(omod4)[1] + coef(omod4)[2]*t1 + coef(omod4)[3]*x +
        coef(omod4)[4]*mean(odat$rad) + coef(omod4)[5]*(t1*x),
      add=T, lwd = 2)

curve(coef(omod4)[1] + coef(omod4)[2]*t2 + coef(omod4)[3]*x +
        coef(omod4)[4]*mean(odat$rad) + coef(omod4)[5]*(t2*x),
      col = "red", lwd = 2, add=T)

curve(coef(omod4)[1] + coef(omod4)[2]*t3 + coef(omod4)[3]*x +
        coef(omod4)[4]*mean(odat$rad) + coef(omod4)[5]*(t3*x),
      col = "darkblue", lwd = 2, add=T)

legend("topright", c("57", "77", "97"), col =
         c("black","red", "darkblue"), lty = 1, lwd = 2,
       title = "Temperature", bty="n")
```



```r
with(odat, plot(temp, log(ozone), las = 1, pch = 16, cex = 1.2,
                col = "darkgrey", ylab = "log(Ozone)", xlab = "Temp"))

w1 <- 2.3
w2 <- 11.5
w3 <- 18.4
x <- at.temp

curve(coef(omod4)[1] + coef(omod4)[2]*x + coef(omod4)[3]*w1 +
        coef(omod4)[4]*mean(odat$rad) + coef(omod4)[5]*(w1*x),
      add=T, lwd = 2)

curve(coef(omod4)[1] + coef(omod4)[2]*x + coef(omod4)[3]*w2 +
        coef(omod4)[4]*mean(odat$rad) + coef(omod4)[5]*(w2*x),
      col = "red", lwd = 2, add=T)

curve(coef(omod4)[1] + coef(omod4)[2]*x + coef(omod4)[3]*w2 +
        coef(omod4)[4]*mean(odat$rad) + coef(omod4)[5]*(w3*x),
      col = "darkblue", lwd = 2, add=T)

legend("topleft", c("2.3", "11.5", "18.4"), col =
         c("black","red", "darkblue"), lty = 1, lwd = 2, title = "Wind",
       bty="n")
```


Wind
2.3
11.5
18.4
log(Ozone)
0
1
2
3
4
5
Temp
60
70
80
90