STAT452/652 Solution to HW02c

Due on Sept 25, 2020

1 Lecture 2C: Multiple Linear Regression

1.1 Question 1

```
### Activate data and extract the columns we want
data("airquality")
data = airquality[,1:4]

### Make a scatterplot matrix
pairs(data)
```

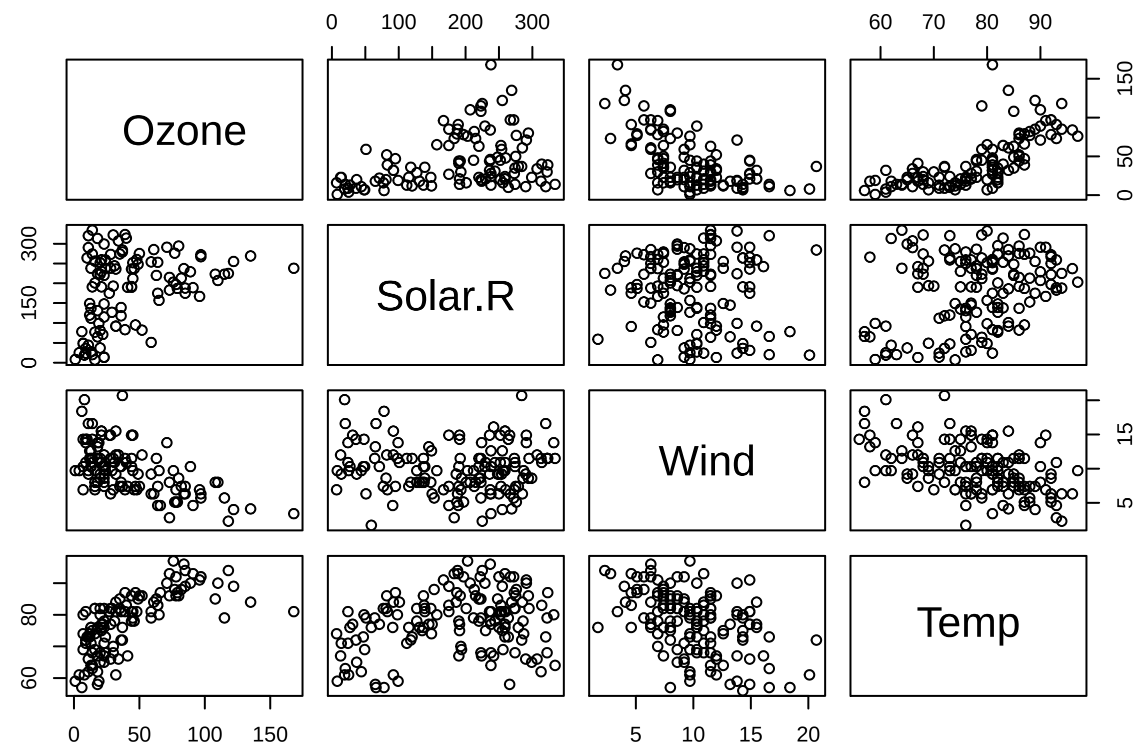

Figure 1: Scatterplot matrix for air quality data

(a) We see in Figure 1 that temperature has a pretty strong increasing relationsip with ozone with a few strong outliers, and is somewhat non-linear. Wind has a weaker decreasing relationship with ozone, and also looks somewhat non-linear. Solar radiation has an increasing relationship with ozone which appears linear, but is strongly heteroscedastic (has a funnel shape, or nonconstantr variance).

(b) Again from Figure 1, we see that wind and temperature have a weak-moderate decreasing linear relationship. The other pairs of explanatory variables have very weak relationships. The only other notable feature is that there appears to be some heteroscedasticity in the relationship between temperature and solar radiation.

1.2 Question 2

```
### Fit regression models
fit.solar = lm(Ozone ~ Solar.R, data=data)
fit.wind = lm(Ozone ~ Wind, data=data)
fit.temp = lm(Ozone ~ Temp, data=data)

### Create a function to extract slopes and t-values
get.reg.summary = function(fit){
  fit.detail = summary(fit)
  info.table = fit.detail$coef
  output = info.table[2,c(1,3)]
  return(output)
}

### Create container for slopes and t-values
model.output = array(0, dim = c(3,2))
colnames(model.output) = c("Slope", "T-Value")
rownames(model.output) = c("Solar.R", "Wind", "Temp")

### Extract model output
model.output[1,] = round(get.reg.summary(fit.solar), 2)
model.output[2,] = round(get.reg.summary(fit.wind), 2)
model.output[3,] = round(get.reg.summary(fit.temp), 2)

cat("PART (a)")
```

```
## PART (a)
```

```
print(model.output)
```

```
##         Slope T-Value
## Solar.R  0.13    3.88
## Wind    -5.55   -8.04
## Temp     2.43   10.42
```

```
### Set all three plots side-by-side
### I.e. Create a plot grid with 1 row and 3 columns
par(mfrow = c(1,3))

### Plot bi-variate scatterplots and add regression lines
with(data, plot(Solar.R, Ozone, main = "Solar Radiation"))
abline(fit.solar)
with(data, plot(Wind, Ozone, main = "Wind Speed"))
abline(fit.wind)
```

```r
with(data, plot(Temp, Ozone, main = "Temperature"))
abline(fit.temp)
```

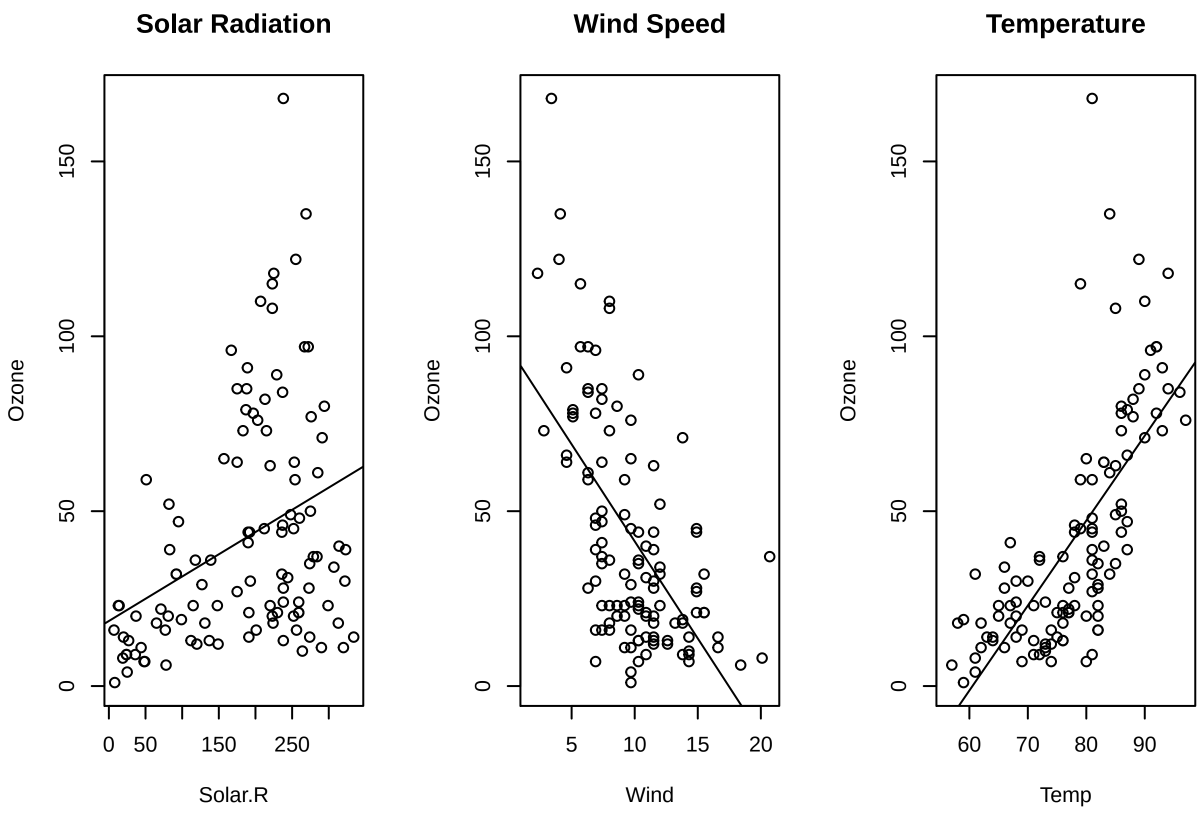

Figure 2: Univariate regression models

```r
### Reset normal plotting
### I.e. Return to a 1x1 plot grid
par(mfrow = c(1,1))
```

(b) See Figure 2 for simple linear regression fits.
- The line for solar radiation provides reasonable fit to the shape of the trend. With so much vatiability in the responses, it is hard to do much better, although I would rather that it did a better job for smaller Solar.R values where there is less variability.
- The line for wind speed provides shows a clear decreasing pattern, but does not capture the full trend. The line underestimates responses at low and high wind speeds and underestimates it in the middle.
- The line for temperature provides shows the strong increasing trend, but, like with wind, seems to underestimate ozone at extremes and underestimate it in the middle

1.3 Question 3

```
### Suppress output from this code chunk because we took a screenshot
library(rgl)
open3d()
with(data, plot3d(Ozone ~ Temp + Wind))
```

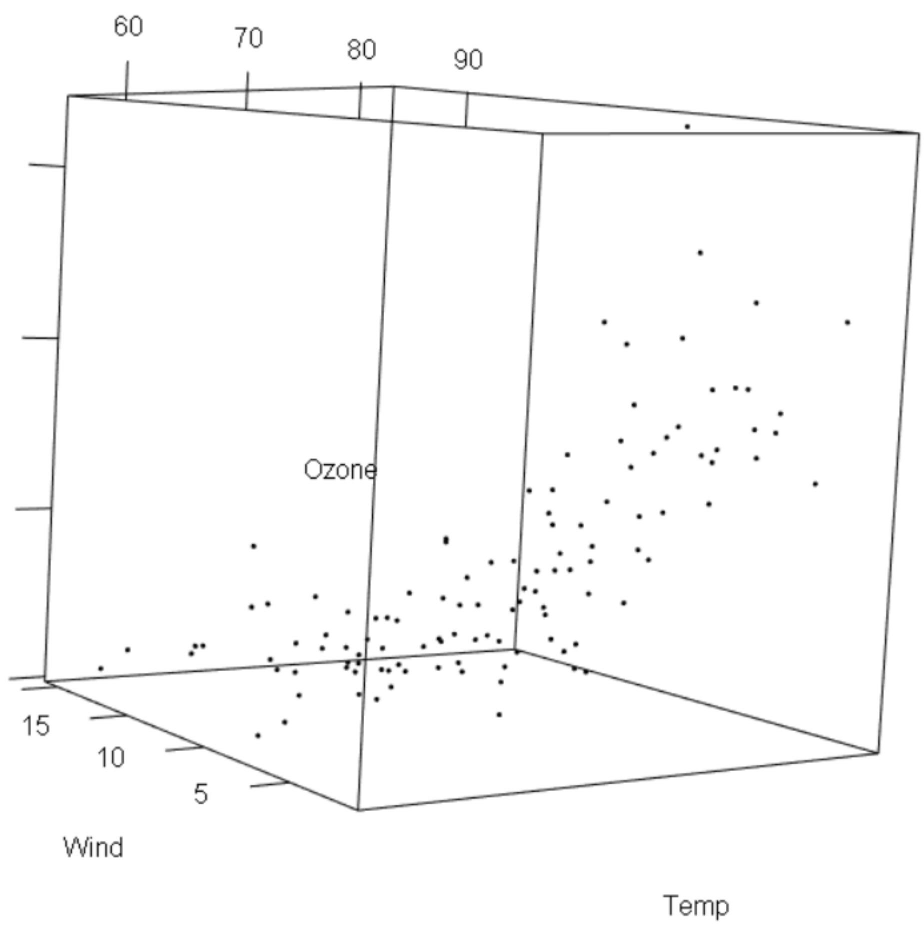

Figure 3: Scatterplot of ozone against temperature and wind

1.4 Question 4

```
### Fit multiple linear regression
fit2 = lm(Ozone ~ Temp + Wind, data=data)

### Extract and print slope/t-values
info2 = summary(fit2)
coef2 = info2$coef
```

```r
table2 = coef2[-1,c(1,3)]
cat("PART (a)")
```

```
## PART (a)
```

```r
print(round(table2,3))
```

```
##      Estimate t value
## Temp    1.840   7.362
## Wind   -3.055  -4.607
```

Comparing the above table to the one from Question 2, we see that the two variables' effects have the same direction, but the slopes have a bit smaller magnitude and the t-values are smaller (but still relatively large and significant).

```r
### Suppress output from this code chunk because we took a screenshot

### Construct a grid in Temp and Wind
vals.wind = seq(from = 0, to = 22, by = 1)
vals.temp = seq(from = 54, to = 98, by = 2)
pred.grid = expand.grid(Wind = vals.wind, Temp = vals.temp)

### Get fitted values on our grid
pred.ozone = predict(fit2, newdata = pred.grid)

### Plot fitted surface with scatterplot
open3d()
persp3d(x = vals.wind, y = vals.temp, z = pred.ozone, col = "orange")
with(data, points3d(Ozone ~ Wind + Temp))
```

From Figure 4, we see that, while our regression model is fairly close to most points, there does appear to be some curvature in the data that is not captured by the linear model.

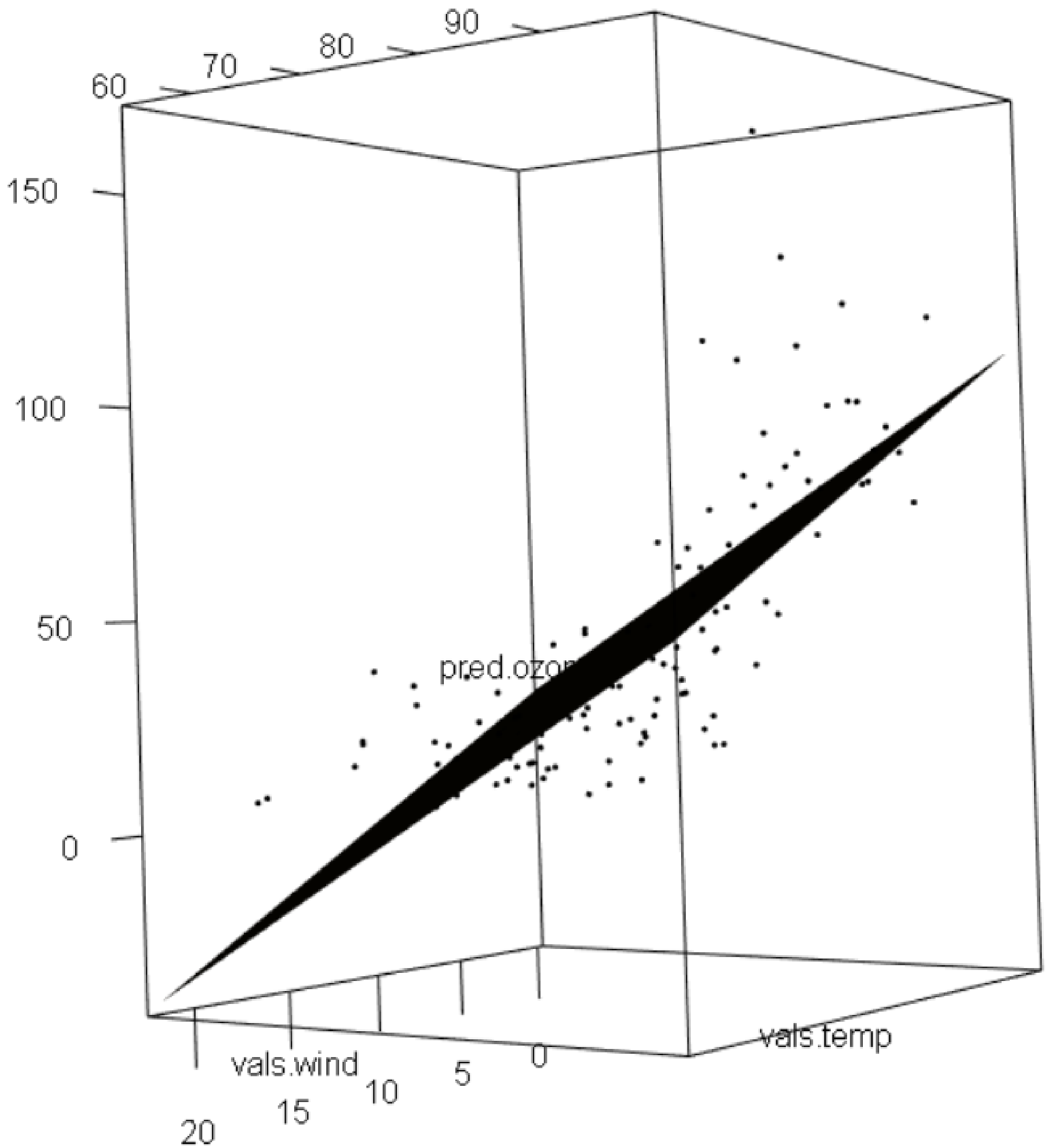

Figure 4: Regression surface of ozone on temperature and wind