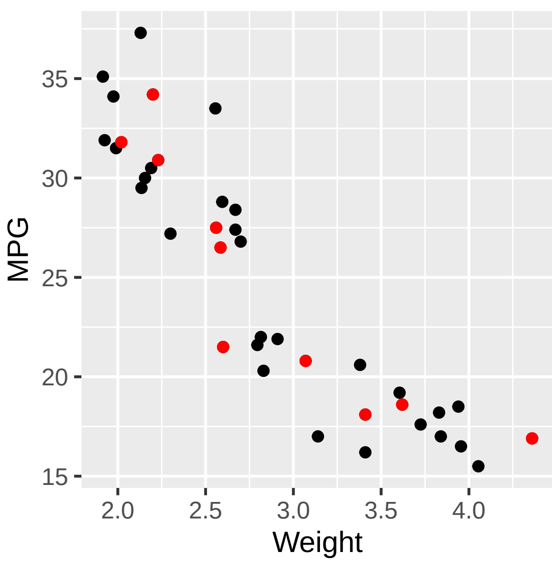
Model Comparison Example

We have a data set with information about 38 cars. Here we will look at two of the variables measured about these cars, their `Weight` (explanatory variable) and their fuel efficiency (response variable) as measured in miles per gallon (`MPG`, higher MPG is more fuel efficient). The full data set is loaded in and plotted here. In addition, I have highlighted in red a subset of these observations that I will use in fitting three candidate models below.

```
library(readr) # for read_csv, which can read csv files from the internet
library(dplyr) # for miscellaneous data manipulation functions (like slice)
library(ggplot2) # for making plots
library(gridExtra) # for grid.arrange, which arranges the plots next to each other
library(polynom) # for obtaining the third polynomial fit below

cars <- read_csv("http://www.evanlray.com/data/sdm4/Cars.csv")
train_inds <- c(1, 6, 8, 14, 15, 16, 21, 32, 33, 37)
train_cars <- cars %>% slice(train_inds) # 10 observations to use in getting fits below.

ggplot() +
  geom_point(data = cars, mapping = aes(x = Weight, y = MPG)) +
  geom_point(data = train_cars, mapping = aes(x = Weight, y = MPG), color = "red")
```



Below is R code for making plots displaying three separate polynomial regression fits to the 10 observations highlighted in red above: one with a degree 1 polynomial (i.e., a line), one with a degree 2 polynomial (a parabola), and one with a degree 9 polynomial.

```
lm1 <- lm(MPG ~ Weight, data = train_cars)
predict_1 <- function(x) {
  predict(lm1, data.frame(Weight = x))
}

p1 <- ggplot(data = train_cars, mapping = aes(x = Weight, y = MPG)) +
  geom_point(color = "red") +
  stat_function(fun = predict_1) +
  ggtitle("linear fit")
```

```r
lm2 <- lm(MPG ~ poly(Weight, degree = 2, raw = TRUE), data = train_cars)
predict_2 <- function(x) {
  predict(lm2, data.frame(Weight = x))
}

p2 <- ggplot(data = train_cars, mapping = aes(x = Weight, y = MPG)) +
  geom_point(color = "red") +
  stat_function(fun = predict_2) +
  ggtitle("quadratic fit")

# Our degree 9 polynomial fit is not obtained from lm (although you could do that too)
# You don't need to know how to use the poly.calc function.
fit9 <- poly.calc(train_cars$Weight, train_cars$MPG)
print(fit9)
```

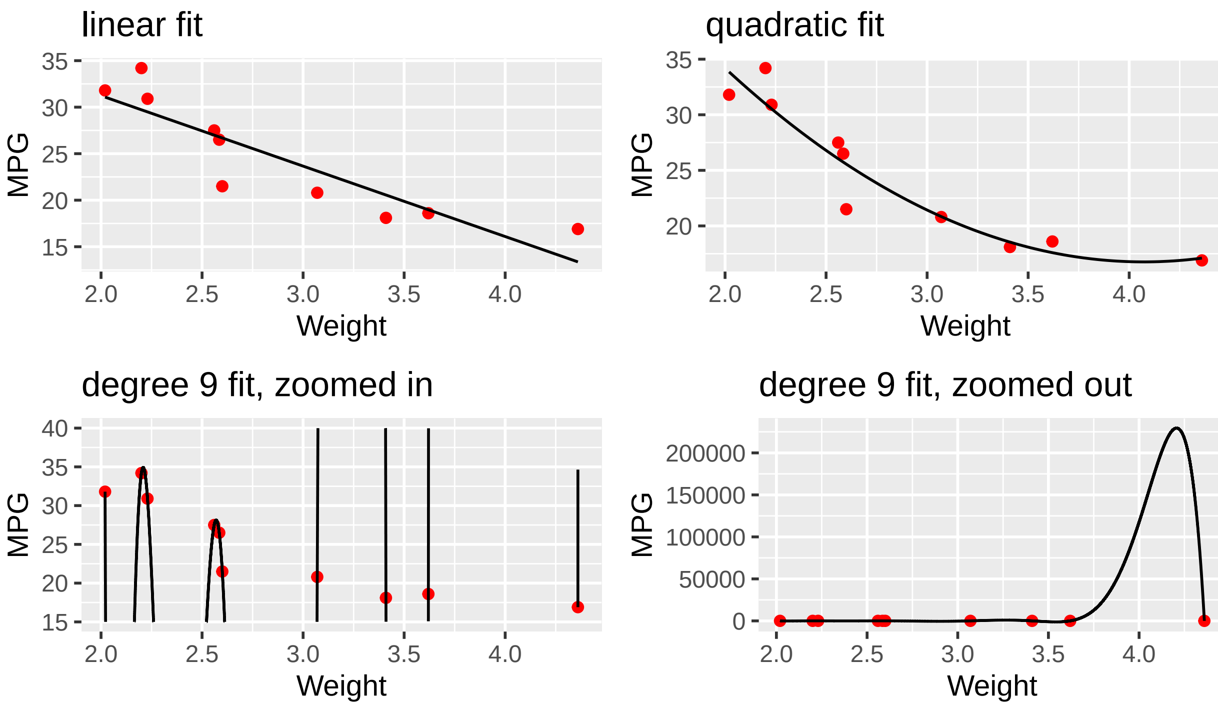
```
## 1299465000 - 4291996000*x + 6257315000*x^2 - 5284115000*x^3 +
## 2847945000*x^4 - 1015739000*x^5 + 239687900*x^6 - 36078670*x^7 +
## 3142816*x^8 - 120690*x^9
```

```r
predict_9 <- as.function(fit9)

p3 <- ggplot(data = train_cars, mapping = aes(x = Weight, y = MPG)) +
  geom_point(color = "red") +
  stat_function(fun = predict_9, n = 1000001) +
  ylim(c(15, 40)) +
  ggtitle("degree 9 fit, zoomed in")

p4 <- ggplot(data = train_cars, mapping = aes(x = Weight, y = MPG)) +
  geom_point(color = "red") +
  stat_function(fun = predict_9, n = 100001) +
  ggtitle("degree 9 fit, zoomed out")

grid.arrange(p1, p2, p3, p4, nrow = 2, ncol = 2)
```



With your neighbors, discuss which of these models you would prefer to use for predicting MPG and why. Then answer the questions below:

If you chose the model by picking the one with the smallest RSS, which model would you choose? Is that the most appropriate model?

Being as specific and concrete as possible, write down a rule for selecting your preferred model based only on *visual* characteristics of the plot. (That is, your rule should not involve any calculations of numeric quantities).

Being as specific and concrete as possible, write down a rule for selecting your preferred model based only on a *quantitative* summary of the data. You can describe how you would calculate your numeric summary of the data; if you'd like you can write down a formula.